API Reference

AWS CodeCommit

API Version 2015-04-13

AWS CodeCommit: API Reference

Table of Contents

Welcome

This is the *AWS CodeCommit API Reference*. This reference provides descriptions of the operations and data types for AWS CodeCommit API along with usage examples.

You can use the AWS CodeCommit API to work with the following objects:

Repositories, by calling the following:

- BatchGetRepositories, which returns information about one or more repositories associated with your AWS account.
- CreateRepository, which creates an AWS CodeCommit repository.
- DeleteRepository, which deletes an AWS CodeCommit repository.
- GetRepository, which returns information about a specified repository.
- ListRepositories, which lists all AWS CodeCommit repositories associated with your AWS account.
- UpdateRepositoryDescription, which sets or updates the description of the repository.
- UpdateRepositoryEncryptionKey, which updates the AWS Key Management Service encryption key used to encrypt and decrypt a repository.
- UpdateRepositoryName, which changes the name of the repository. If you change the name of a repository, no other users of that repository can access it until you send them the new HTTPS or SSH URL to use.

Branches, by calling the following:

- CreateBranch, which creates a branch in a specified repository.
- DeleteBranch, which deletes the specified branch in a repository unless it is the default branch.
- GetBranch, which returns information about a specified branch.
- ListBranches, which lists all branches for a specified repository.
- UpdateDefaultBranch, which changes the default branch for a repository.

Files, by calling the following:

- DeleteFile, which deletes the content of a specified file from a specified branch.
- GetBlob, which returns the base-64 encoded content of an individual Git blob object in a repository.

- GetFile, which returns the base-64 encoded content of a specified file.
- GetFolder, which returns the contents of a specified folder or directory.
- ListFileCommitHistory, which retrieves a list of commits and changes to a specified file.
- PutFile, which adds or modifies a single file in a specified repository and branch.

Commits, by calling the following:

- BatchGetCommits, which returns information about one or more commits in a repository.
- CreateCommit, which creates a commit for changes to a repository.
- GetCommit, which returns information about a commit, including commit messages and author and committer information.
- GetDifferences, which returns information about the differences in a valid commit specifier (such as a branch, tag, HEAD, commit ID, or other fully qualified reference).

Merges, by calling the following:

- BatchDescribeMergeConflicts, which returns information about conflicts in a merge between commits in a repository.
- CreateUnreferencedMergeCommit, which creates an unreferenced commit between two branches or commits for the purpose of comparing them and identifying any potential conflicts.
- DescribeMergeConflicts, which returns information about merge conflicts between the base, source, and destination versions of a file in a potential merge.
- GetMergeCommit, which returns information about the merge between a source and destination commit.
- GetMergeConflicts, which returns information about merge conflicts between the source and destination branch in a pull request.
- GetMergeOptions, which returns information about the available merge options between two branches or commit specifiers.
- MergeBranchesByFastForward, which merges two branches using the fast-forward merge option.
- MergeBranchesBySquash, which merges two branches using the squash merge option.
- MergeBranchesByThreeWay, which merges two branches using the three-way merge option.

Pull requests, by calling the following:

- CreatePullRequest, which creates a pull request in a specified repository.
- CreatePullRequestApprovalRule, which creates an approval rule for a specified pull request.
- DeletePullRequestApprovalRule, which deletes an approval rule for a specified pull request.
- DescribePullRequestEvents, which returns information about one or more pull request events.
- EvaluatePullRequestApprovalRules, which evaluates whether a pull request has met all the conditions specified in its associated approval rules.
- GetCommentsForPullRequest, which returns information about comments on a specified pull request.
- GetPullRequest, which returns information about a specified pull request.
- GetPullRequestApprovalStates, which returns information about the approval states for a specified pull request.
- GetPullRequestOverrideState, which returns information about whether approval rules have been set aside (overriden) for a pull request, and if so, the Amazon Resource Name (ARN) of the user or identity that overrode the rules and their requirements for the pull request.
- ListPullRequests, which lists all pull requests for a repository.
- MergePullRequestByFastForward, which merges the source destination branch of a pull request into the specified destination branch for that pull request using the fast-forward merge option.
- MergePullRequestBySquash, which merges the source destination branch of a pull request into the specified destination branch for that pull request using the squash merge option.
- MergePullRequestByThreeWay, which merges the source destination branch of a pull request into the specified destination branch for that pull request using the three-way merge option.
- OverridePullRequestApprovalRules, which sets aside all approval rule requirements for a pull request.
- PostCommentForPullRequest, which posts a comment to a pull request at the specified line, file, or request.
- UpdatePullRequestApprovalRuleContent, which updates the structure of an approval rule for a pull request.
- UpdatePullRequestApprovalState, which updates the state of an approval on a pull request.
- UpdatePullRequestDescription, which updates the description of a pull request.
- UpdatePullRequestStatus, which updates the status of a pull request.
- UpdatePullRequestTitle, which updates the title of a pull request.

Approval rule templates, by calling the following:

- AssociateApprovalRuleTemplateWithRepository, which associates a template with a specified repository. After the template is associated with a repository, AWS CodeCommit creates approval rules that match the template conditions on every pull request created in the specified repository.
- BatchAssociateApprovalRuleTemplateWithRepositories, which associates a template with one or more specified repositories. After the template is associated with a repository, AWS CodeCommit creates approval rules that match the template conditions on every pull request created in the specified repositories.
- BatchDisassociateApprovalRuleTemplateFromRepositories, which removes the association between a template and specified repositories so that approval rules based on the template are not automatically created when pull requests are created in those repositories.
- CreateApprovalRuleTemplate, which creates a template for approval rules that can then be associated with one or more repositories in your AWS account.
- DeleteApprovalRuleTemplate, which deletes the specified template. It does not remove approval rules on pull requests already created with the template.
- DisassociateApprovalRuleTemplateFromRepository, which removes the association between a template and a repository so that approval rules based on the template are not automatically created when pull requests are created in the specified repository.
- GetApprovalRuleTemplate, which returns information about an approval rule template.
- ListApprovalRuleTemplates, which lists all approval rule templates in the AWS Region in your AWS account.
- ListAssociatedApprovalRuleTemplatesForRepository, which lists all approval rule templates that are associated with a specified repository.
- ListRepositoriesForApprovalRuleTemplate, which lists all repositories associated with the specified approval rule template.
- UpdateApprovalRuleTemplateDescription, which updates the description of an approval rule template.
- UpdateApprovalRuleTemplateName, which updates the name of an approval rule template.
- UpdateApprovalRuleTemplateContent, which updates the content of an approval rule template.

Comments in a repository, by calling the following:

- DeleteCommentContent, which deletes the content of a comment on a commit in a repository.
- GetComment, which returns information about a comment on a commit.
- GetCommentReactions, which returns information about emoji reactions to comments.
- GetCommentsForComparedCommit, which returns information about comments on the comparison between two commit specifiers in a repository.
- PostCommentForComparedCommit, which creates a comment on the comparison between two commit specifiers in a repository.
- PostCommentReply, which creates a reply to a comment.
- PutCommentReaction, which creates or updates an emoji reaction to a comment.
- UpdateComment, which updates the content of a comment on a commit in a repository.

Tags used to tag resources in AWS CodeCommit (not Git tags), by calling the following:

- ListTagsForResource, which gets information about AWStags for a specified Amazon Resource Name (ARN) in AWS CodeCommit.
- TagResource, which adds or updates tags for a resource in AWS CodeCommit.
- UntagResource, which removes tags for a resource in AWS CodeCommit.

Triggers, by calling the following:

- GetRepositoryTriggers, which returns information about triggers configured for a repository.
- PutRepositoryTriggers, which replaces all triggers for a repository and can be used to create or delete triggers.
- TestRepositoryTriggers, which tests the functionality of a repository trigger by sending data to the trigger target.

For information about how to use AWS CodeCommit, see the AWS CodeCommit User Guide.

This document was last published on March 13, 2026.

Actions

The following actions are supported:

- AssociateApprovalRuleTemplateWithRepository
- BatchAssociateApprovalRuleTemplateWithRepositories
- BatchDescribeMergeConflicts
- BatchDisassociateApprovalRuleTemplateFromRepositories
- BatchGetCommits
- BatchGetRepositories
- CreateApprovalRuleTemplate
- CreateBranch
- CreateCommit
- CreatePullRequest
- CreatePullRequestApprovalRule
- CreateRepository
- CreateUnreferencedMergeCommit
- DeleteApprovalRuleTemplate
- DeleteBranch
- DeleteCommentContent
- DeleteFile
- DeletePullRequestApprovalRule
- DeleteRepository
- DescribeMergeConflicts
- DescribePullRequestEvents
- DisassociateApprovalRuleTemplateFromRepository
- EvaluatePullRequestApprovalRules
- GetApprovalRuleTemplate
- GetBlob
- GetBranch
- GetComment

- GetCommentReactions
- GetCommentsForComparedCommit
- GetCommentsForPullRequest
- GetCommit
- GetDifferences
- GetFile
- GetFolder
- GetMergeCommit
- GetMergeConflicts
- GetMergeOptions
- GetPullRequest
- GetPullRequestApprovalStates
- GetPullRequestOverrideState
- GetRepository
- GetRepositoryTriggers
- ListApprovalRuleTemplates
- ListAssociatedApprovalRuleTemplatesForRepository
- ListBranches
- ListFileCommitHistory
- ListPullRequests
- ListRepositories
- ListRepositoriesForApprovalRuleTemplate
- ListTagsForResource
- MergeBranchesByFastForward
- MergeBranchesBySquash
- MergeBranchesByThreeWay
- MergePullRequestByFastForward
- MergePullRequestBySquash
- MergePullRequestByThreeWay
- OverridePullRequestApprovalRules

- PostCommentForComparedCommit
- PostCommentForPullRequest
- PostCommentReply
- PutCommentReaction
- PutFile
- PutRepositoryTriggers
- TagResource
- TestRepositoryTriggers
- UntagResource
- UpdateApprovalRuleTemplateContent
- UpdateApprovalRuleTemplateDescription
- UpdateApprovalRuleTemplateName
- UpdateComment
- UpdateDefaultBranch
- UpdatePullRequestApprovalRuleContent
- UpdatePullRequestApprovalState
- UpdatePullRequestDescription
- UpdatePullRequestStatus
- UpdatePullRequestTitle
- UpdateRepositoryDescription
- UpdateRepositoryEncryptionKey
- UpdateRepositoryName

AssociateApprovalRuleTemplateWithRepository

Creates an association between an approval rule template and a specified repository. Then, the next time a pull request is created in the repository where the destination reference (if specified) matches the destination reference (branch) for the pull request, an approval rule that matches the template conditions is automatically created for that pull request. If no destination references are specified in the template, an approval rule that matches the template contents is created for all pull requests in that repository.

Request Syntax

```
{
   "approvalRuleTemplateName": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

approvalRuleTemplateName

The name for the approval rule template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: Yes

repositoryName

The name of the repository that you want to associate with the template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors.

ApprovalRuleTemplateDoesNotExistException

The specified approval rule template does not exist. Verify that the name is correct and that you are signed in to the AWS Region where the template was created, and then try again.

HTTP Status Code: 400

ApprovalRuleTemplateNameRequiredException

An approval rule template name is required, but was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidApprovalRuleTemplateNameException

The name of the approval rule template is not valid. Template names must be between 1 and 100 valid characters in length. For more information about limits in AWS CodeCommit, see Quotas in the *AWS CodeCommit User Guide*.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

MaximumRuleTemplatesAssociatedWithRepositoryException

The maximum number of approval rule templates for a repository has been exceeded. You cannot associate more than 25 approval rule templates with a repository.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of AssociateApprovalRuleTemplateWithRepository.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 226
X-Amz-Target: CodeCommit_20150413.AssociateApprovalRuleTemplateWithRepository
X-Amz-Date: 20191021T213222Z
User-Agent: aws-cli/1.16.137 Python/3.6.0 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "approvalRuleTemplateName": "2-approver-rule-for-main"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 1681
Date: Mon, 21 Oxr 2019 22:43:13 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2

- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

BatchAssociateApprovalRuleTemplateWithRepositories

Creates an association between an approval rule template and one or more specified repositories.

Request Syntax

```
{
   "approvalRuleTemplateName": "string",
   "repositoryNames": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

approvalRuleTemplateName

The name of the template you want to associate with one or more repositories.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: Yes

repositoryNames

The names of the repositories you want to associate with the template.

> **Note**
>
> The length constraint limit is for each string in the array. The array itself can be empty.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "associatedRepositoryNames": [ "string" ],
   "errors": [
      {
         "errorCode": "string",
         "errorMessage": "string",
         "repositoryName": "string"
      }
   ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

associatedRepositoryNames

A list of names of the repositories that have been associated with the template.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

errors

A list of any errors that might have occurred while attempting to create the association between the template and the repositories.

Type: Array of BatchAssociateApprovalRuleTemplateWithRepositoriesError objects

Errors

For information about the errors that are common to all actions, see Common Errors.

ApprovalRuleTemplateDoesNotExistException

The specified approval rule template does not exist. Verify that the name is correct and that you are signed in to the AWS Region where the template was created, and then try again.

HTTP Status Code: 400

ApprovalRuleTemplateNameRequiredException

An approval rule template name is required, but was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidApprovalRuleTemplateNameException

The name of the approval rule template is not valid. Template names must be between 1 and 100 valid characters in length. For more information about limits in AWS CodeCommit, see Quotas in the *AWS CodeCommit User Guide*.

HTTP Status Code: 400

MaximumRepositoryNamesExceededException

The maximum number of allowed repository names was exceeded. Currently, this number is 100.

HTTP Status Code: 400

RepositoryNamesRequiredException

At least one repository name object is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of BatchAssociateApprovalRuleTemplateWithRepositories.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 226
X-Amz-Target: CodeCommit_20150413.BatchAssociateApprovalRuleTemplateWithRepositories
X-Amz-Date: 20191021T213222Z
User-Agent: aws-cli/1.16.137 Python/3.6.0 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
    "approvalRuleTemplateName": "2-approver-rule-for-main",
    "repositoryNames": [
        "MyDemoRepo",
        "MyOtherDemoRepo"
    ]
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 1681
Date: Mon, 21 Oct 2019 22:43:13 GMT

{
    "associatedRepositoryNames": [
        "MyDemoRepo",
        "MyOtherDemoRepo"
    ],
    "errors": []
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

BatchDescribeMergeConflicts

Returns information about one or more merge conflicts in the attempted merge of two commit specifiers using the squash or three-way merge strategy.

Request Syntax

```
{
   "conflictDetailLevel": "string",
   "conflictResolutionStrategy": "string",
   "destinationCommitSpecifier": "string",
   "filePaths": [ "string" ],
   "maxConflictFiles": number,
   "maxMergeHunks": number,
   "mergeOption": "string",
   "nextToken": "string",
   "repositoryName": "string",
   "sourceCommitSpecifier": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

conflictDetailLevel

The level of conflict detail to use. If unspecified, the default FILE_LEVEL is used, which returns a not-mergeable result if the same file has differences in both branches. If LINE_LEVEL is specified, a conflict is considered not mergeable if the same file in both branches has differences on the same line.

Type: String

Valid Values: `FILE_LEVEL | LINE_LEVEL`

Required: No

conflictResolutionStrategy

Specifies which branch to use when resolving conflicts, or whether to attempt automatically merging two versions of a file. The default is NONE, which requires any conflicts to be resolved manually before the merge operation is successful.

Type: String

Valid Values: `NONE | ACCEPT_SOURCE | ACCEPT_DESTINATION | AUTOMERGE`

Required: No

destinationCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

filePaths

The path of the target files used to describe the conflicts. If not specified, the default is all conflict files.

Type: Array of strings

Required: No

maxConflictFiles

The maximum number of files to include in the output.

Type: Integer

Required: No

maxMergeHunks

The maximum number of merge hunks to include in the output.

Type: Integer

Required: No

mergeOption

The merge option or strategy you want to use to merge the code.

Type: String

Valid Values: `FAST_FORWARD_MERGE | SQUASH_MERGE | THREE_WAY_MERGE`

Required: Yes

nextToken

An enumeration token that, when provided in a request, returns the next batch of the results.

Type: String

Required: No

repositoryName

The name of the repository that contains the merge conflicts you want to review.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

sourceCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

Response Syntax

```
{
   "baseCommitId": "string",
   "conflicts": [
      {
         "conflictMetadata": {
            "contentConflict": boolean,
            "fileModeConflict": boolean,
            "fileModes": {
               "base": "string",
               "destination": "string",
               "source": "string"
            },
            "filePath": "string",
            "fileSizes": {
               "base": number,
               "destination": number,
               "source": number
            },
            "isBinaryFile": {
               "base": boolean,
               "destination": boolean,
               "source": boolean
            },
            "mergeOperations": {
               "destination": "string",
               "source": "string"
            },
            "numberOfConflicts": number,
            "objectTypeConflict": boolean,
            "objectTypes": {
               "base": "string",
               "destination": "string",
               "source": "string"
            }
         },
         "mergeHunks": [
            {
               "base": {
                  "endLine": number,
                  "hunkContent": "string",
                  "startLine": number
               },
               "destination": {
                  "endLine": number,
                  "hunkContent": "string",
                  "startLine": number
               },
               "isConflict": boolean,
               "source": {
                  "endLine": number,
                  "hunkContent": "string",
                  "startLine": number
               }
            }
         ]
      }
   ],
   "destinationCommitId": "string",
   "errors": [
      {
         "exceptionName": "string",
         "filePath": "string",
         "message": "string"
      }
   ],
   "nextToken": "string",
   "sourceCommitId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

baseCommitId

The commit ID of the merge base.

Type: String

conflicts

A list of conflicts for each file, including the conflict metadata and the hunks of the differences between the files.

Type: Array of Conflict objects

destinationCommitId

The commit ID of the destination commit specifier that was used in the merge evaluation.

Type: String

errors

A list of any errors returned while describing the merge conflicts for each file.

Type: Array of BatchDescribeMergeConflictsError objects

nextToken

An enumeration token that can be used in a request to return the next batch of the results.

Type: String

sourceCommitId

The commit ID of the source commit specifier that was used in the merge evaluation.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

CommitRequiredException

A commit was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidCommitException

The specified commit is not valid.

HTTP Status Code: 400

InvalidConflictDetailLevelException

The specified conflict detail level is not valid.

HTTP Status Code: 400

InvalidConflictResolutionStrategyException

The specified conflict resolution strategy is not valid.

HTTP Status Code: 400

InvalidContinuationTokenException

The specified continuation token is not valid.

HTTP Status Code: 400

InvalidMaxConflictFilesException

The specified value for the number of conflict files to return is not valid.

HTTP Status Code: 400

InvalidMaxMergeHunksException

The specified value for the number of merge hunks to return is not valid.

HTTP Status Code: 400

InvalidMergeOptionException

The specified merge option is not valid for this operation. Not all merge strategies are supported for all operations.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

MaximumFileContentToLoadExceededException

The number of files to load exceeds the allowed limit.

HTTP Status Code: 400

MaximumItemsToCompareExceededException

The number of items to compare between the source or destination branches and the merge base has exceeded the maximum allowed.

HTTP Status Code: 400

MergeOptionRequiredException

A merge option or stategy is required, and none was provided.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

TipsDivergenceExceededException

The divergence between the tips of the provided commit specifiers is too great to determine whether there might be any merge conflicts. Locally compare the specifiers using `git diff` or a diff tool.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of BatchDescribeMergeConflicts.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 226
X-Amz-Target: CodeCommit_20150413.BatchDescribeMergeConflicts
X-Amz-Date: 20190428T213222Z
User-Agent: aws-cli/1.16.137 Python/3.6.0 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "destinationCommitSpecifier": "bugfix-bug1234",
  "sourceCommitSpecifier": "main",
  "mergeOption": "THREE_WAY_MERGE",
  "conflictDetailLevel" "LINE_LEVEL",
  "conflictResolutionStrategy": "NONE",
  "nextToken": "exampleToken",
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 1681
Date: Sun, 28 Apr 2019 22:43:13 GMT

{
       "conflicts":[
        {
            "conflictMetadata": {
               "filePath": "file1.py",
               "fileSizes": {
                 "source": 123,
                 "destination": 125,
                 "base": 124
            },
            "fileModes": {
                "source": "EXECUTABLE",
                "destination": "EXECUTABLE",
                "base": "EXECUTABLE"
                },
            "numberOfConflicts", 4,
            "isBinaryFile": {
                "source": false,
                "destination": false,
                "base": false
            },
            "contentConflict": true,
            "fileModeConflict": false,
            "mergeOperations": {
                "source": "M",
                "destination": "M"
            }
        }
    "mergeHunks":[
        {
            "mergeHunk": {
                "isConflict": true
                "source": {
                    "startLine": 123,
                    "endLine": 123,
                    "hunkContent" "JzCQbIVyEXAMPLE="
                }
                "destination": {
                    "startLine": 125,
                    "endLine": 125,
                    "hunkContent" "BytPbuMiEXAMPLE="
                }
                "base": {
                    "startLine": 124,
                    "endLine": 124,
                    "hunkContent" "MnKCdITaEXAMPLE="
                }
            }
        }
    ]
    "errors":
    "sourceCommitId": "c5709475EXAMPLE",
    "destinationCommitId": "317f8570EXAMPLE",
    "baseCommitId": "fb12a539EXAMPLE",
    "nextToken": "exampleToken"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

BatchDisassociateApprovalRuleTemplateFromRepositories

Removes the association between an approval rule template and one or more specified repositories.

Request Syntax

```
{
   "approvalRuleTemplateName": "string",
   "repositoryNames": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

approvalRuleTemplateName

The name of the template that you want to disassociate from one or more repositories.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: Yes

repositoryNames

The repository names that you want to disassociate from the approval rule template.

> **Note**
>
> The length constraint limit is for each string in the array. The array itself can be empty.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "disassociatedRepositoryNames": [ "string" ],
   "errors": [
      {
         "errorCode": "string",
         "errorMessage": "string",
         "repositoryName": "string"
      }
   ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

disassociatedRepositoryNames

A list of repository names that have had their association with the template removed.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

errors

A list of any errors that might have occurred while attempting to remove the association between the template and the repositories.

Type: Array of BatchDisassociateApprovalRuleTemplateFromRepositoriesError objects

Errors

For information about the errors that are common to all actions, see Common Errors.

ApprovalRuleTemplateDoesNotExistException

The specified approval rule template does not exist. Verify that the name is correct and that you are signed in to the AWS Region where the template was created, and then try again.

HTTP Status Code: 400

ApprovalRuleTemplateNameRequiredException

An approval rule template name is required, but was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidApprovalRuleTemplateNameException

The name of the approval rule template is not valid. Template names must be between 1 and 100 valid characters in length. For more information about limits in AWS CodeCommit, see Quotas in the *AWS CodeCommit User Guide*.

HTTP Status Code: 400

MaximumRepositoryNamesExceededException

The maximum number of allowed repository names was exceeded. Currently, this number is 100.

HTTP Status Code: 400

RepositoryNamesRequiredException

At least one repository name object is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of BatchDisassociateApprovalRuleTemplateFromRepositories.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 226
X-Amz-Target: CodeCommit_20150413.BatchDisassociateApprovalRuleTemplateFromRepositories
X-Amz-Date: 20191021T213222Z
User-Agent: aws-cli/1.16.137 Python/3.6.0 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
    "approvalRuleTemplateName": "2-approver-rule-for-main",
    "repositoryNames": [
        "MyDemoRepo",
        "MyOtherDemoRepo"
    ]
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 1681
Date: Mon, 21 Oxr 2019 22:43:13 GMT

    "disassociatedRepositoryNames": [
        "MyDemoRepo",
        "MyOtherDemoRepo"
    ],
    "errors": []
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

BatchGetCommits

Returns information about the contents of one or more commits in a repository.

Request Syntax

```
{
   "commitIds": [ "string" ],
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

commitIds

The full commit IDs of the commits to get information about.

> **Note**
>
> You must supply the full SHA IDs of each commit. You cannot use shortened SHA IDs.

Type: Array of strings

Required: Yes

repositoryName

The name of the repository that contains the commits.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "commits": [
      {
         "additionalData": "string",
         "author": {
            "date": "string",
            "email": "string",
            "name": "string"
         },
         "commitId": "string",
         "committer": {
            "date": "string",
            "email": "string",
            "name": "string"
         },
         "message": "string",
         "parents": [ "string" ],
         "treeId": "string"
      }
   ],
   "errors": [
      {
         "commitId": "string",
         "errorCode": "string",
         "errorMessage": "string"
      }
   ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

commits

An array of commit data type objects, each of which contains information about a specified commit.

Type: Array of Commit objects

errors

Returns any commit IDs for which information could not be found. For example, if one of the commit IDs was a shortened SHA ID or that commit was not found in the specified repository, the ID returns an error object with more information.

Type: Array of BatchGetCommitsError objects

Errors

For information about the errors that are common to all actions, see Common Errors.

CommitIdsLimitExceededException

The maximum number of allowed commit IDs in a batch request is 100. Verify that your batch requests contains no more than 100 commit IDs, and then try again.

HTTP Status Code: 400

CommitIdsListRequiredException

A list of commit IDs is required, but was either not specified or the list was empty.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of BatchGetCommits.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 105
X-Amz-Target: CodeCommit_20150413.BatchGetCommits
X-Amz-Date: 20190724T201222Z
User-Agent: aws-cli/1.11.187 Python/3.4.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "commitIds": [
    317f8570EXAMPLE,
    4c925148EXAMPLE
  ]
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 1218
Date: Wed, 24 Jul 2019 20:12:13 GMT

{
    "commits": [
      {
        "additionalData": "",
        "committer": {
            "date": "1508280564 -0800",
            "name": "Mary Major",
            "email": "mary_major@example.com"
        },
        "author": {
            "date": "1508280564 -0800",
            "name": "Mary Major",
            "email": "mary_major@example.com"
        },
        "commitId": "317f8570EXAMPLE",
        "treeId": "1f330709EXAMPLE",
        "parents": [
            "6e147360EXAMPLE"
        ],
        "message": "Change variable name and add new response element"
    },
    {
        "additionalData": "",
        "committer": {
            "date": "1508280542 -0800",
            "name": "Li Juan",
            "email": "li_juan@example.com"
        },
        "author": {
            "date": "1508280542 -0800",
            "name": "Li Juan",
            "email": "li_juan@example.com"
        },
        "commitId": "4c925148EXAMPLE",
        "treeId": "1f330709EXAMPLE",
        "parents": [
            "317f8570EXAMPLE"
        ],
        "message": "Added new class"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3

- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

BatchGetRepositories

Returns information about one or more repositories.

> **Note**
>
> The description field for a repository accepts all HTML characters and all valid Unicode characters. Applications that do not HTML-encode the description and display it in a webpage can expose users to potentially malicious code. Make sure that you HTML-encode the description field in any application that uses this API to display the repository description on a webpage.

Request Syntax

```
{
   "repositoryNames": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

repositoryNames

The names of the repositories to get information about.

> **Note**
>
> The length constraint limit is for each string in the array. The array itself can be empty.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "errors": [
      {
         "errorCode": "string",
         "errorMessage": "string",
         "repositoryId": "string",
         "repositoryName": "string"
      }
   ],
   "repositories": [
      {
         "accountId": "string",
         "Arn": "string",
         "cloneUrlHttp": "string",
         "cloneUrlSsh": "string",
         "creationDate": number,
         "defaultBranch": "string",
         "kmsKeyId": "string",
         "lastModifiedDate": number,
         "repositoryDescription": "string",
         "repositoryId": "string",
         "repositoryName": "string"
      }
   ],
   "repositoriesNotFound": [ "string" ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

errors

Returns information about any errors returned when attempting to retrieve information about the repositories.

Type: Array of BatchGetRepositoriesError objects

repositories

A list of repositories returned by the batch get repositories operation.

Type: Array of RepositoryMetadata objects

repositoriesNotFound

Returns a list of repository names for which information could not be found.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

MaximumRepositoryNamesExceededException

The maximum number of allowed repository names was exceeded. Currently, this number is 100.

HTTP Status Code: 400

RepositoryNamesRequiredException

At least one repository name object is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of BatchGetRepositories.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
```

```
Content-Length: 50
X-Amz-Target: CodeCommit_20150413.BatchGetRepositories
X-Amz-Date: 20151028T213222Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryNames": [
    "MyDemoRepo",
 "MyOtherDemoRepo"
  ]
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 550
Date: Wed, 28 Oct 2015 21:32:24 GMT

{
    "repositories": [
        {
            "repositoryName": "MyDemoRepo",
   "defaultBranch": "main",
            "cloneUrlSsh": "ssh://git-codecommit.us-east-1.amazonaws.com/v1/repos/
MyDemoRepo",
            "lastModifiedDate": 1441725693.583,
            "repositoryDescription": "My demonstration repository",
            "cloneUrlHttp": "https://git-codecommit.us-east-1.amazonaws.com/v1/repos/
MyDemoRepo",
            "defaultBranch": "main",
            "kmsKeyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "creationDate": 1441725693.583,
            "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
            "Arn": "arn:aws:codecommit:us-east-1:123456789012EXAMPLE:MyDemoRepo",
            "accountId": "123456789012"
        },
        {
            "creationDate": 1431980067.669,
            "defaultBranch": "main",
            "repositoryName": "MyOtherDemoRepo",
            "cloneUrlSsh": "ssh://git-codecommit.us-east-1.amazonaws.com/v1/repos/
MyOtherDemoRepo",
            "lastModifiedDate": 1444776815.847,
            "repositoryDescription": "My other demonstration repository",
            "cloneUrlHttp": "https://git-codecommit.us-east-1.amazonaws.com/v1/repos/
MyOtherDemoRepo",
            "defaultBranch": "release",
            "kmsKeyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "repositoryId": "cfc29ac4-b0cb-44dc-9990-f6f51EXAMPLE",
            "Arn": "arn:aws:codecommit:us-east-1:123456789012EXAMPLE:MyOtherDemoRepo",
            "accountId": "123456789012"
        }
    ],
    "repositoriesNotFound": []
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateApprovalRuleTemplate

Creates a template for approval rules that can then be associated with one or more repositories in your AWS account. When you associate a template with a repository, AWS CodeCommit creates an approval rule that matches the conditions of the template for all pull requests that meet the conditions of the template. For more information, see AssociateApprovalRuleTemplateWithRepository.

Request Syntax

```
{
   "approvalRuleTemplateContent": "string",
   "approvalRuleTemplateDescription": "string",
   "approvalRuleTemplateName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

approvalRuleTemplateContent

The content of the approval rule that is created on pull requests in associated repositories. If you specify one or more destination references (branches), approval rules are created in an associated repository only if their destination references (branches) match those specified in the template.

> **Note**
>
> When you create the content of the approval rule template, you can specify approvers in an approval pool in one of two ways:
>
> - **CodeCommitApprovers**: This option only requires an AWS account and a resource. It can be used for both IAM users and federated access users whose name matches the provided resource name. This is a very powerful option that offers a great deal of flexibility. For example, if you specify the AWS account *123456789012* and *Mary_Major*, all of the following are counted as approvals coming from that user:

  - An IAM user in the account (arn:aws:iam::*123456789012*:user/*Mary_Major*)
  - A federated user identified in IAM as Mary_Major (arn:aws:sts::*123456789012*:federated-user/*Mary_Major*)

  This option does not recognize an active session of someone assuming the role of CodeCommitReview with a role session name of *Mary_Major* (arn:aws:sts::*123456789012*:assumed-role/CodeCommitReview/*Mary_Major*) unless you include a wildcard (*Mary_Major).
- **Fully qualified ARN**: This option allows you to specify the fully qualified Amazon Resource Name (ARN) of the IAM user or role.

For more information about IAM ARNs, wildcards, and formats, see IAM Identifiers in the *IAM User Guide*.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 3000.

Required: Yes

approvalRuleTemplateDescription

The description of the approval rule template. Consider providing a description that explains what this template does and when it might be appropriate to associate it with repositories.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1000.

Required: No

approvalRuleTemplateName

The name of the approval rule template. Provide descriptive names, because this name is applied to the approval rules created automatically in associated repositories.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: Yes

Response Syntax

```
{
   "approvalRuleTemplate": {
      "approvalRuleTemplateContent": "string",
      "approvalRuleTemplateDescription": "string",
      "approvalRuleTemplateId": "string",
      "approvalRuleTemplateName": "string",
      "creationDate": number,
      "lastModifiedDate": number,
      "lastModifiedUser": "string",
      "ruleContentSha256": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

approvalRuleTemplate

The content and structure of the created approval rule template.

Type: ApprovalRuleTemplate object

Errors

For information about the errors that are common to all actions, see Common Errors.

ApprovalRuleTemplateContentRequiredException

The content for the approval rule template is empty. You must provide some content for an approval rule template. The content cannot be null.

HTTP Status Code: 400

ApprovalRuleTemplateNameAlreadyExistsException

You cannot create an approval rule template with that name because a template with that name already exists in this AWS Region for your AWS account. Approval rule template names must be unique.

HTTP Status Code: 400

ApprovalRuleTemplateNameRequiredException

An approval rule template name is required, but was not specified.

HTTP Status Code: 400

InvalidApprovalRuleTemplateContentException

The content of the approval rule template is not valid.

HTTP Status Code: 400

InvalidApprovalRuleTemplateDescriptionException

The description for the approval rule template is not valid because it exceeds the maximum characters allowed for a description. For more information about limits in AWS CodeCommit, see Quotas in the *AWS CodeCommit User Guide*.

HTTP Status Code: 400

InvalidApprovalRuleTemplateNameException

The name of the approval rule template is not valid. Template names must be between 1 and 100 valid characters in length. For more information about limits in AWS CodeCommit, see Quotas in the *AWS CodeCommit User Guide*.

HTTP Status Code: 400

NumberOfRuleTemplatesExceededException

The maximum number of approval rule templates has been exceeded for this AWS Region.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of CreateApprovalRuleTemplate.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 113
X-Amz-Target: CodeCommit_20150413.CreateApprovalRuleTemplate
X-Amz-Date: 20191021T221237Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\",
\"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
    "approvalRuleTemplateDescription": "Requires two developers from the team to
 approve the pull request if the destination branch is main",
    "approvalRuleTemplateName": "2-approver-rule-for-main"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 0
Date: Mon, 21 Oct 2019 22:12:40 GMT

{
    "approvalRuleTemplate": {
        "approvalRuleTemplateName": "2-approver-rule-for-main",
        "creationDate": 1571356106.936,
        "approvalRuleTemplateId": "dd8b17fe-EXAMPLE",
        "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\",
\"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "approvalRuleTemplateDescription": "Requires two developers from the team to
 approve the pull request if the destination branch is main",
        "lastModifiedDate": 1571356106.936,
        "ruleContentSha256": "4711b576EXAMPLE"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateBranch

Creates a branch in a repository and points the branch to a commit.

> **Note**
>
> Calling the create branch operation does not set a repository's default branch. To do this, call the update default branch operation.

Request Syntax

```
{
   "branchName": "string",
   "commitId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

branchName

The name of the new branch to create.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: Yes

commitId

The ID of the commit to point the new branch to.

Type: String

Required: Yes

repositoryName

The name of the repository in which you want to create the new branch.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors.

BranchNameExistsException

Cannot create the branch with the specified name because the commit conflicts with an existing branch with the same name. Branch names must be unique.

HTTP Status Code: 400

BranchNameRequiredException

A branch name is required, but was not specified.

HTTP Status Code: 400

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

CommitIdRequiredException

A commit ID was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidBranchNameException

The specified reference name is not valid.

HTTP Status Code: 400

InvalidCommitIdException

The specified commit ID is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of CreateBranch.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 113
X-Amz-Target: CodeCommit_20150413.CreateBranch
X-Amz-Date: 20151028T221237Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "commitId": "317f8570EXAMPLE",
  "repositoryName": "MyDemoRepo",
  "branchName": "MyNewBranch"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 0
Date: Wed, 28 Oct 2015 22:12:40 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateCommit

Creates a commit for a repository on the tip of a specified branch.

Request Syntax

```
{
   "authorName": "string",
   "branchName": "string",
   "commitMessage": "string",
   "deleteFiles": [
      {
         "filePath": "string"
      }
   ],
   "email": "string",
   "keepEmptyFolders": boolean,
   "parentCommitId": "string",
   "putFiles": [
      {
         "fileContent": blob,
         "fileMode": "string",
         "filePath": "string",
         "sourceFile": {
            "filePath": "string",
            "isMove": boolean
         }
      }
   ],
   "repositoryName": "string",
   "setFileModes": [
      {
         "fileMode": "string",
         "filePath": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

authorName

The name of the author who created the commit. This information is used as both the author and committer for the commit.

Type: String

Required: No

branchName

The name of the branch where you create the commit.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: Yes

commitMessage

The commit message you want to include in the commit. Commit messages are limited to 256 KB. If no message is specified, a default message is used.

Type: String

Required: No

deleteFiles

The files to delete in this commit. These files still exist in earlier commits.

Type: Array of DeleteFileEntry objects

Required: No

email

The email address of the person who created the commit.

Type: String

Required: No

keepEmptyFolders

If the commit contains deletions, whether to keep a folder or folder structure if the changes leave the folders empty. If true, a ..gitkeep file is created for empty folders. The default is false.

Type: Boolean

Required: No

parentCommitId

The ID of the commit that is the parent of the commit you create. Not required if this is an empty repository.

Type: String

Required: No

putFiles

The files to add or update in this commit.

Type: Array of PutFileEntry objects

Required: No

repositoryName

The name of the repository where you create the commit.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

setFileModes

The file modes to update for files in this commit.

Type: Array of SetFileModeEntry objects

Required: No

Response Syntax

```
{
   "commitId": "string",
   "filesAdded": [
      {
         "absolutePath": "string",
         "blobId": "string",
         "fileMode": "string"
      }
   ],
   "filesDeleted": [
      {
         "absolutePath": "string",
         "blobId": "string",
         "fileMode": "string"
      }
   ],
   "filesUpdated": [
      {
         "absolutePath": "string",
         "blobId": "string",
         "fileMode": "string"
      }
   ],
   "treeId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

commitId

The full commit ID of the commit that contains your committed file changes.

Type: String

filesAdded

The files added as part of the committed file changes.

Type: Array of FileMetadata objects

filesDeleted

The files deleted as part of the committed file changes.

Type: Array of FileMetadata objects

filesUpdated

The files updated as part of the commited file changes.

Type: Array of FileMetadata objects

treeId

The full SHA-1 pointer of the tree information for the commit that contains the commited file changes.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

BranchDoesNotExistException

The specified branch does not exist.

HTTP Status Code: 400

BranchNameIsTagNameException

The specified branch name is not valid because it is a tag name. Enter the name of a branch in the repository. For a list of valid branch names, use ListBranches.

HTTP Status Code: 400

BranchNameRequiredException

A branch name is required, but was not specified.

HTTP Status Code: 400

CommitMessageLengthExceededException

The commit message is too long. Provide a shorter string.

HTTP Status Code: 400

DirectoryNameConflictsWithFileNameException

A file cannot be added to the repository because the specified path name has the same name as a file that already exists in this repository. Either provide a different name for the file, or specify a different path for the file.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

FileContentAndSourceFileSpecifiedException

The commit cannot be created because both a source file and file content have been specified for the same file. You cannot provide both. Either specify a source file or provide the file content directly.

HTTP Status Code: 400

FileContentSizeLimitExceededException

The file cannot be added because it is too large. The maximum file size is 6 MB, and the combined file content change size is 7 MB. Consider making these changes using a Git client.

HTTP Status Code: 400

FileDoesNotExistException

The specified file does not exist. Verify that you have used the correct file name, full path, and extension.

HTTP Status Code: 400

FileEntryRequiredException

The commit cannot be created because no files have been specified as added, updated, or changed (PutFile or DeleteFile) for the commit.

HTTP Status Code: 400

FileModeRequiredException

The commit cannot be created because no file mode has been specified. A file mode is required to update mode permissions for a file.

HTTP Status Code: 400

FileNameConflictsWithDirectoryNameException

A file cannot be added to the repository because the specified file name has the same name as a directory in this repository. Either provide another name for the file, or add the file in a directory that does not match the file name.

HTTP Status Code: 400

FilePathConflictsWithSubmodulePathException

The commit cannot be created because a specified file path points to a submodule. Verify that the destination files have valid file paths that do not point to a submodule.

HTTP Status Code: 400

FolderContentSizeLimitExceededException

The commit cannot be created because at least one of the overall changes in the commit results in a folder whose contents exceed the limit of 6 MB. Either reduce the number and size of your changes, or split the changes across multiple folders.

HTTP Status Code: 400

InvalidBranchNameException

The specified reference name is not valid.

HTTP Status Code: 400

InvalidDeletionParameterException

The specified deletion parameter is not valid.

HTTP Status Code: 400

InvalidEmailException

The specified email address either contains one or more characters that are not allowed, or it exceeds the maximum number of characters allowed for an email address.

HTTP Status Code: 400

InvalidFileModeException

The specified file mode permission is not valid. For a list of valid file mode permissions, see PutFile.

HTTP Status Code: 400

InvalidParentCommitIdException

The parent commit ID is not valid. The commit ID cannot be empty, and must match the head commit ID for the branch of the repository where you want to add or update a file.

HTTP Status Code: 400

InvalidPathException

The specified path is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

MaximumFileEntriesExceededException

The number of specified files to change as part of this commit exceeds the maximum number of files that can be changed in a single commit. Consider using a Git client for these changes.

HTTP Status Code: 400

NameLengthExceededException

The user name is not valid because it has exceeded the character limit for author names.

HTTP Status Code: 400

NoChangeException

The commit cannot be created because no changes will be made to the repository as a result of this commit. A commit must contain at least one change.

HTTP Status Code: 400

ParentCommitDoesNotExistException

The parent commit ID is not valid because it does not exist. The specified parent commit ID does not exist in the specified branch of the repository.

HTTP Status Code: 400

ParentCommitIdOutdatedException

The file could not be added because the provided parent commit ID is not the current tip of the specified branch. To view the full commit ID of the current head of the branch, use GetBranch.

HTTP Status Code: 400

ParentCommitIdRequiredException

A parent commit ID is required. To view the full commit ID of a branch in a repository, use GetBranch or a Git command (for example, git pull or git log).

HTTP Status Code: 400

PathRequiredException

The folderPath for a location cannot be null.

HTTP Status Code: 400

PutFileEntryConflictException

The commit cannot be created because one or more files specified in the commit reference both a file and a folder.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

RestrictedSourceFileException

The commit cannot be created because one of the changes specifies copying or moving a .gitkeep file.

HTTP Status Code: 400

SamePathRequestException

The commit cannot be created because one or more changes in this commit duplicate actions in the same file path. For example, you cannot make the same delete request to the same file in the same file path twice, or make a delete request and a move request to the same file as part of the same commit.

HTTP Status Code: 400

SourceFileOrContentRequiredException

The commit cannot be created because no source files or file content have been specified for the commit.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of CreateCommit.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.CreateCommit
X-Amz-Date: 201801219T224019Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "repositoryName": "MyDemoRepo",
   "branchName": "MyFeatureBranch",
   "parentCommitId": "4c925148EXAMPLE",
   "commitMessage": "I'm creating this commit to update a variable name in a number of
 files.",
   "authorName": "Saanvi Sarkar"
   "email": "saanvi_sarkar@example.com",
   "keepEmptyFolders": false,
   "putFiles": [
     {
       "filePath": "file1.py",
       "fileMode": "EXECUTABLE",
       "fileContent": "bucket_name = sys.argv[1] region = sys.argv[2]"
     },
     {
       "filePath": "file2.txt",
       "fileMode": "NORMAL",
       "fileContent": "//Adding a comment to explain the variable changes in file1.py"
     },
     {
       "filePath": "images/image1.png",
       "fileMode": "NORMAL",
       "sourceFile": {
         "filePath": "pictures/picture.png",
         "isMove": true
       }
     }
   ],
   "deleteFiles": [
     {
       "filePath": "ExampleSolution.py"
     }
   ]
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Fri, 19 Dec 2018 22:40:19 GMT

{
   "commitId": "317f8570EXAMPLE",
   "treeId": "347a3408EXAMPLE",
   "filesAdded": [
        {
        "absolutePath": "images/image1.png",
        "blobId": "d68ba6ccEXAMPLE",
        "fileMode": "NORMAL"
        }
    ],
    "filesUpdated": [
        {
        "absolutePath": "file1.py",
        "blobId": "0a4d55a8EXAMPLE",
        "fileMode": "EXECUTABLE"
        },
        {
        "absolutePath": "file2.txt",
        "blobId": "915766bbEXAMPLE",
        "fileMode": "NORMAL"
        }
    ],
    "filesDeleted": [
        {
        "absolutePath": "ExampleSolution.py",
        "blobId": "4f9cebe6aEXAMPLE",
        "fileMode": "EXECUTABLE"
        },
        {
        "absolutePath": "pictures/picture.png",
        "blobId": "fb12a539EXAMPLE",
        "fileMode": "NORMAL"
        }
    ]
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreatePullRequest

Creates a pull request in the specified repository.

Request Syntax

```
{
   "clientRequestToken": "string",
   "description": "string",
   "targets": [
      {
         "destinationReference": "string",
         "repositoryName": "string",
         "sourceReference": "string"
      }
   ],
   "title": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

clientRequestToken

A unique, client-generated idempotency token that, when provided in a request, ensures the request cannot be repeated with a changed parameter. If a request is received with the same parameters and a token is included, the request returns information about the initial request that used that token.

> **Note**
>
> The AWSSDKs prepopulate client request tokens. If you are using an AWSSDK, an idempotency token is created for you.

Type: String

Required: No

description

A description of the pull request.

Type: String

Length Constraints: Maximum length of 10240.

Required: No

targets

The targets for the pull request, including the source of the code to be reviewed (the source branch) and the destination where the creator of the pull request intends the code to be merged after the pull request is closed (the destination branch).

Type: Array of Target objects

Required: Yes

title

The title of the pull request. This title is used to identify the pull request to other users in the repository.

Type: String

Length Constraints: Maximum length of 150.

Required: Yes

Response Syntax

```
{
   "pullRequest": {
      "approvalRules": [
         {
            "approvalRuleContent": "string",
            "approvalRuleId": "string",
            "approvalRuleName": "string",
            "creationDate": number,
            "lastModifiedDate": number,
            "lastModifiedUser": "string",
            "originApprovalRuleTemplate": {
               "approvalRuleTemplateId": "string",
               "approvalRuleTemplateName": "string"
            },
            "ruleContentSha256": "string"
         }
      ],
      "authorArn": "string",
      "clientRequestToken": "string",
      "creationDate": number,
      "description": "string",
      "lastActivityDate": number,
      "pullRequestId": "string",
      "pullRequestStatus": "string",
      "pullRequestTargets": [
         {
            "destinationCommit": "string",
            "destinationReference": "string",
            "mergeBase": "string",
            "mergeMetadata": {
               "isMerged": boolean,
               "mergeCommitId": "string",
               "mergedBy": "string",
               "mergeOption": "string"
            },
            "repositoryName": "string",
            "sourceCommit": "string",
            "sourceReference": "string"
         }
      ],
      "revisionId": "string",
      "title": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

pullRequest

Information about the newly created pull request.

Type: PullRequest object

Errors

For information about the errors that are common to all actions, see Common Errors.

ClientRequestTokenRequiredException

A client request token is required. A client request token is an unique, client-generated idempotency token that, when provided in a request, ensures the request cannot be repeated with a changed parameter. If a request is received with the same parameters and a token is included, the request returns information about the initial request that used that token.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

IdempotencyParameterMismatchException

The client request token is not valid. Either the token is not in a valid format, or the token has been used in a previous request and cannot be reused.

HTTP Status Code: 400

InvalidClientRequestTokenException

The client request token is not valid.

HTTP Status Code: 400

InvalidDescriptionException

The pull request description is not valid. Descriptions cannot be more than 1,000 characters.

HTTP Status Code: 400

InvalidReferenceNameException

The specified reference name format is not valid. Reference names must conform to the Git references format (for example, refs/heads/main). For more information, see Git Internals - Git References or consult your Git documentation.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

InvalidTargetException

The target for the pull request is not valid. A target must contain the full values for the repository name, source branch, and destination branch for the pull request.

HTTP Status Code: 400

InvalidTargetsException

The targets for the pull request is not valid or not in a valid format. Targets are a list of target objects. Each target object must contain the full values for the repository name, source branch, and destination branch for a pull request.

HTTP Status Code: 400

InvalidTitleException

The title of the pull request is not valid. Pull request titles cannot exceed 100 characters in length.

HTTP Status Code: 400

MaximumOpenPullRequestsExceededException

You cannot create the pull request because the repository has too many open pull requests. The maximum number of open pull requests for a repository is 1,000. Close one or more open pull requests, and then try again.

HTTP Status Code: 400

MultipleRepositoriesInPullRequestException

You cannot include more than one repository in a pull request. Make sure you have specified only one repository name in your request, and then try again.

HTTP Status Code: 400

ReferenceDoesNotExistException

The specified reference does not exist. You must provide a full commit ID.

HTTP Status Code: 400

ReferenceNameRequiredException

A reference name is required, but none was provided.

HTTP Status Code: 400

ReferenceTypeNotSupportedException

The specified reference is not a supported type.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

SourceAndDestinationAreSameException

The source branch and destination branch for the pull request are the same. You must specify different branches for the source and destination.

HTTP Status Code: 400

TargetRequiredException

A pull request target is required. It cannot be empty or null. A pull request target must contain the full values for the repository name, source branch, and destination branch for the pull request.

HTTP Status Code: 400

TargetsRequiredException

An array of target objects is required. It cannot be empty or null.

HTTP Status Code: 400

TitleRequiredException

A pull request title is required. It cannot be empty or null.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of CreatePullRequest.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.CreatePullRequest
X-Amz-Date: 20171025T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "clientRequestToken": "123Example",
   "description": "A code review of the new feature I just added to the service.",
   "targets": [
      {
         "destinationReference": "refs/heads/main",
         "repositoryName": "MyDemoRepo",
         "sourceReference": "refs/heads/jane-branch"
      }
   ],
   "title": "Pronunciation difficulty analyzer"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Wed, 25 Oct 2017 20:20:13 GMT

{
    "pullRequest": {
        "approvalRules": [
            {
                "approvalRuleContent": "{\"Version\": \"2018-11-08\",
\"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
                "approvalRuleId": "dd8b17fe-EXAMPLE",
                "approvalRuleName": "2-approver-rule-for-main",
                "creationDate": 1571356106.936,
                "lastModifiedDate": 571356106.936,
                "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
                "originApprovalRuleTemplate": {
                    "approvalRuleTemplateId": "dd8b17fe-EXAMPLE",
                    "approvalRuleTemplateName": "2-approver-rule-for-main"
                },
                "ruleContentSha256": "4711b576EXAMPLE"
            }
        ],
        "authorArn": "arn:aws:iam::123456789012:user/Jane_Doe",
        "description": "A code review of the new feature I just added to the service.",
        "title": "Pronunciation difficulty analyzer",
        "pullRequestTargets": [
            {
                "destinationCommit": "5d036259EXAMPLE",
                "destinationReference": "refs/heads/main",
                "repositoryName": "MyDemoRepo",
                "sourceCommit": "317f8570EXAMPLE",
                "sourceReference": "refs/heads/jane-branch",
                "mergeMetadata": {
                    "isMerged": false
                }
            }
        ],
        "lastActivityDate": 1508962823.285,
        "pullRequestId": "42",
        "clientRequestToken": "123Example",
        "pullRequestStatus": "OPEN",
        "creationDate": 1508962823.285
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4

- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreatePullRequestApprovalRule

Creates an approval rule for a pull request.

Request Syntax

```
{
   "approvalRuleContent": "string",
   "approvalRuleName": "string",
   "pullRequestId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

approvalRuleContent

The content of the approval rule, including the number of approvals needed and the structure of an approval pool defined for approvals, if any. For more information about approval pools, see the AWS CodeCommit User Guide.

> **Note**
>
> When you create the content of the approval rule, you can specify approvers in an approval pool in one of two ways:
>
> - **CodeCommitApprovers**: This option only requires an AWS account and a resource. It can be used for both IAM users and federated access users whose name matches the provided resource name. This is a very powerful option that offers a great deal of flexibility. For example, if you specify the AWS account *123456789012* and *Mary_Major*, all of the following would be counted as approvals coming from that user:
>   - An IAM user in the account (arn:aws:iam::*123456789012*:user/*Mary_Major*)
>   - A federated user identified in IAM as Mary_Major (arn:aws:sts::*123456789012*:federated-user/*Mary_Major*)

> This option does not recognize an active session of someone assuming the role of CodeCommitReview with a role session name of *Mary_Major* (arn:aws:sts::*123456789012*:assumed-role/CodeCommitReview/*Mary_Major*) unless you include a wildcard (*Mary_Major).
>
> - **Fully qualified ARN**: This option allows you to specify the fully qualified Amazon Resource Name (ARN) of the IAM user or role.
>
> For more information about IAM ARNs, wildcards, and formats, see IAM Identifiers in the *IAM User Guide*.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 3000.

Required: Yes

approvalRuleName

The name for the approval rule.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: Yes

pullRequestId

The system-generated ID of the pull request for which you want to create the approval rule.

Type: String

Required: Yes

Response Syntax

```
{
   "approvalRule": {
      "approvalRuleContent": "string",
      "approvalRuleId": "string",
      "approvalRuleName": "string",
      "creationDate": number,
      "lastModifiedDate": number,
      "lastModifiedUser": "string",
      "originApprovalRuleTemplate": {
         "approvalRuleTemplateId": "string",
         "approvalRuleTemplateName": "string"
      },
      "ruleContentSha256": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

approvalRule

Information about the created approval rule.

Type: ApprovalRule object

Errors

For information about the errors that are common to all actions, see Common Errors.

ApprovalRuleContentRequiredException

The content for the approval rule is empty. You must provide some content for an approval rule. The content cannot be null.

HTTP Status Code: 400

ApprovalRuleNameAlreadyExistsException

An approval rule with that name already exists. Approval rule names must be unique within the scope of a pull request.

HTTP Status Code: 400

ApprovalRuleNameRequiredException

An approval rule name is required, but was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidApprovalRuleContentException

The content for the approval rule is not valid.

HTTP Status Code: 400

InvalidApprovalRuleNameException

The name for the approval rule is not valid.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

NumberOfRulesExceededException

The approval rule cannot be added. The pull request has the maximum number of approval rules associated with it.

HTTP Status Code: 400

PullRequestAlreadyClosedException

The pull request status cannot be updated because it is already closed.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of CreatePullRequestApprovalRule.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.CreatePullRequestApprovalRule
X-Amz-Date: 20191025T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
```

```
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
    "pullRequestId": "27",
    "approvalRuleName": "Require two approved approvers",
    "approvalRuleContent": "{\"Version\": \"2018-11-08\",\"Statements\": [{\"Type
\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Fri, 25 Oct 2019 20:20:13 GMT

{
    "approvalRule": {
        "approvalRuleName": "Require two approved approvers",
        "lastModifiedDate": 1570752871.932,
        "ruleContentSha256": "7c44e6ebEXAMPLE",
        "creationDate": 1570752871.932,
        "approvalRuleId": "aac33506-EXAMPLE",
        "approvalRuleContent": "{\"Version\": \"2018-11-08\",\"Statements\":
 [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4

- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateRepository

Creates a new, empty repository.

Request Syntax

```
{
   "kmsKeyId": "string",
   "repositoryDescription": "string",
   "repositoryName": "string",
   "tags": {
      "string" : "string"
   }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

kmsKeyId

The ID of the encryption key. You can view the ID of an encryption key in the AWS KMS console, or use the AWS KMS APIs to programmatically retrieve a key ID. For more information about acceptable values for kmsKeyID, see KeyId in the Decrypt API description in the *AWS Key Management Service API Reference*.

If no key is specified, the default `aws/codecommit` AWS managed key is used.

Type: String

Pattern: `^[a-zA-Z0-9:/_-]+$`

Required: No

repositoryDescription

A comment or description about the new repository.

> **Note**
>
> The description field for a repository accepts all HTML characters and all valid Unicode characters. Applications that do not HTML-encode the description and display it in a webpage can expose users to potentially malicious code. Make sure that you HTML-encode the description field in any application that uses this API to display the repository description on a webpage.

Type: String

Length Constraints: Maximum length of 1000.

Required: No

repositoryName

The name of the new repository to be created.

> **Note**
>
> The repository name must be unique across the calling AWS account. Repository names are limited to 100 alphanumeric, dash, and underscore characters, and cannot include certain characters. For more information about the limits on repository names, see Quotas in the *AWS CodeCommit User Guide*. The suffix .git is prohibited.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

tags

One or more tag key-value pairs to use when tagging this repository.

Type: String to string map

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Required: No

Response Syntax

```
{
   "repositoryMetadata": {
      "accountId": "string",
      "Arn": "string",
      "cloneUrlHttp": "string",
      "cloneUrlSsh": "string",
      "creationDate": number,
      "defaultBranch": "string",
      "kmsKeyId": "string",
      "lastModifiedDate": number,
      "repositoryDescription": "string",
      "repositoryId": "string",
      "repositoryName": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

repositoryMetadata

Information about the newly created repository.

Type: RepositoryMetadata object

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyInvalidIdException

The AWS Key Management Service encryption key is not valid.

HTTP Status Code: 400

EncryptionKeyInvalidUsageException

A AWS KMS encryption key was used to try and encrypt or decrypt a repository, but either the repository or the key was not in a valid state to support the operation.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidRepositoryDescriptionException

The specified repository description is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

InvalidSystemTagUsageException

The specified tag is not valid. Key names cannot be prefixed with aws:.

HTTP Status Code: 400

InvalidTagsMapException

The map of tags is not valid.

HTTP Status Code: 400

OperationNotAllowedException

Thrown when request is denied due to other reason than lack of permission.

HTTP Status Code: 400

RepositoryLimitExceededException

A repository resource limit was exceeded.

HTTP Status Code: 400

RepositoryNameExistsException

The specified repository name already exists.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

TagPolicyException

The tag policy is not valid.

HTTP Status Code: 400

TooManyTagsException

The maximum number of tags for an AWS CodeCommit resource has been exceeded.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of CreateRepository.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 88
X-Amz-Target: CodeCommit_20150413.CreateRepository
X-Amz-Date: 20151028T223339Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "repositoryDescription": "My demonstration repository",
  "tags": {
    "Team":"Saanvi"
  }
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 483
```

```
Date: Wed, 28 Oct 2015 22:33:42 GMT

{
    "repositoryMetadata": {
        "repositoryName": "MyDemoRepo",
        "cloneUrlSsh": "ssh://git-codecommit.us-east-1.amazonaws.com/v1/repos/
MyDemoRepo",
        "lastModifiedDate": 1446071622.494,
        "repositoryDescription": "My demonstration repository",
        "cloneUrlHttp": "https://git-codecommit.us-east-1.amazonaws.com/v1/repos/
MyDemoRepo",
        "defaultBranch": main,
        "kmsKeyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "creationDate": 1446071622.494,
        "repositoryId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbb",
        "Arn": "arn:aws:codecommit:us-east-1:123456789012EXAMPLE:MyDemoRepo",
        "accountId": "123456789012"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateUnreferencedMergeCommit

Creates an unreferenced commit that represents the result of merging two branches using a specified merge strategy. This can help you determine the outcome of a potential merge. This API cannot be used with the fast-forward merge strategy because that strategy does not create a merge commit.

> **Note**
>
> This unreferenced merge commit can only be accessed using the GetCommit API or through git commands such as git fetch. To retrieve this commit, you must specify its commit ID or otherwise reference it.

Request Syntax

```
{
   "authorName": "string",
   "commitMessage": "string",
   "conflictDetailLevel": "string",
   "conflictResolution": {
      "deleteFiles": [
         {
            "filePath": "string"
         }
      ],
      "replaceContents": [
         {
            "content": blob,
            "fileMode": "string",
            "filePath": "string",
            "replacementType": "string"
         }
      ],
      "setFileModes": [
         {
            "fileMode": "string",
            "filePath": "string"
         }
      ]
   },
   "conflictResolutionStrategy": "string",
   "destinationCommitSpecifier": "string",
   "email": "string",
   "keepEmptyFolders": boolean,
   "mergeOption": "string",
   "repositoryName": "string",
   "sourceCommitSpecifier": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

authorName

The name of the author who created the unreferenced commit. This information is used as both the author and committer for the commit.

Type: String

Required: No

commitMessage

The commit message for the unreferenced commit.

Type: String

Required: No

conflictDetailLevel

The level of conflict detail to use. If unspecified, the default FILE_LEVEL is used, which returns a not-mergeable result if the same file has differences in both branches. If LINE_LEVEL is specified, a conflict is considered not mergeable if the same file in both branches has differences on the same line.

Type: String

Valid Values: `FILE_LEVEL | LINE_LEVEL`

Required: No

conflictResolution

If AUTOMERGE is the conflict resolution strategy, a list of inputs to use when resolving conflicts during a merge.

Type: ConflictResolution object

Required: No

conflictResolutionStrategy

Specifies which branch to use when resolving conflicts, or whether to attempt automatically merging two versions of a file. The default is NONE, which requires any conflicts to be resolved manually before the merge operation is successful.

Type: String

Valid Values: `NONE | ACCEPT_SOURCE | ACCEPT_DESTINATION | AUTOMERGE`

Required: No

destinationCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

email

The email address for the person who created the unreferenced commit.

Type: String

Required: No

keepEmptyFolders

If the commit contains deletions, whether to keep a folder or folder structure if the changes leave the folders empty. If this is specified as true, a .gitkeep file is created for empty folders. The default is false.

Type: Boolean

Required: No

mergeOption

The merge option or strategy you want to use to merge the code.

Type: String

Valid Values: `FAST_FORWARD_MERGE | SQUASH_MERGE | THREE_WAY_MERGE`

Required: Yes

repositoryName

The name of the repository where you want to create the unreferenced merge commit.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

sourceCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

Response Syntax

```
{
   "commitId": "string",
   "treeId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

commitId

The full commit ID of the commit that contains your merge results.

Type: String

treeId

The full SHA-1 pointer of the tree information for the commit that contains the merge results.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

CommitMessageLengthExceededException

The commit message is too long. Provide a shorter string.

HTTP Status Code: 400

CommitRequiredException

A commit was not specified.

HTTP Status Code: 400

ConcurrentReferenceUpdateException

The merge cannot be completed because the target branch has been modified. Another user might have modified the target branch while the merge was in progress. Wait a few minutes, and then try again.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

FileContentSizeLimitExceededException

The file cannot be added because it is too large. The maximum file size is 6 MB, and the combined file content change size is 7 MB. Consider making these changes using a Git client.

HTTP Status Code: 400

FileModeRequiredException

The commit cannot be created because no file mode has been specified. A file mode is required to update mode permissions for a file.

HTTP Status Code: 400

FolderContentSizeLimitExceededException

The commit cannot be created because at least one of the overall changes in the commit results in a folder whose contents exceed the limit of 6 MB. Either reduce the number and size of your changes, or split the changes across multiple folders.

HTTP Status Code: 400

InvalidCommitException

The specified commit is not valid.

HTTP Status Code: 400

InvalidConflictDetailLevelException

The specified conflict detail level is not valid.

HTTP Status Code: 400

InvalidConflictResolutionException

The specified conflict resolution list is not valid.

HTTP Status Code: 400

InvalidConflictResolutionStrategyException

The specified conflict resolution strategy is not valid.

HTTP Status Code: 400

InvalidEmailException

The specified email address either contains one or more characters that are not allowed, or it exceeds the maximum number of characters allowed for an email address.

HTTP Status Code: 400

InvalidFileModeException

The specified file mode permission is not valid. For a list of valid file mode permissions, see PutFile.

HTTP Status Code: 400

InvalidMergeOptionException

The specified merge option is not valid for this operation. Not all merge strategies are supported for all operations.

HTTP Status Code: 400

InvalidPathException

The specified path is not valid.

HTTP Status Code: 400

InvalidReplacementContentException

Automerge was specified for resolving the conflict, but the replacement type is not valid or content is missing.

HTTP Status Code: 400

InvalidReplacementTypeException

Automerge was specified for resolving the conflict, but the specified replacement type is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

ManualMergeRequiredException

The pull request cannot be merged automatically into the destination branch. You must manually merge the branches and resolve any conflicts.

HTTP Status Code: 400

MaximumConflictResolutionEntriesExceededException

The number of allowed conflict resolution entries was exceeded.

HTTP Status Code: 400

MaximumFileContentToLoadExceededException

The number of files to load exceeds the allowed limit.

HTTP Status Code: 400

MaximumItemsToCompareExceededException

The number of items to compare between the source or destination branches and the merge base has exceeded the maximum allowed.

HTTP Status Code: 400

MergeOptionRequiredException

A merge option or stategy is required, and none was provided.

HTTP Status Code: 400

MultipleConflictResolutionEntriesException

More than one conflict resolution entries exists for the conflict. A conflict can have only one conflict resolution entry.

HTTP Status Code: 400

NameLengthExceededException

The user name is not valid because it has exceeded the character limit for author names.

HTTP Status Code: 400

PathRequiredException

The folderPath for a location cannot be null.

HTTP Status Code: 400

ReplacementContentRequiredException

USE_NEW_CONTENT was specified, but no replacement content has been provided.

HTTP Status Code: 400

ReplacementTypeRequiredException

A replacement type is required.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

TipsDivergenceExceededException

The divergence between the tips of the provided commit specifiers is too great to determine whether there might be any merge conflicts. Locally compare the specifiers using `git diff` or a diff tool.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of CreateUnreferencedMergeCommit.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 312
X-Amz-Target: CodeCommit_20150413.CreateUnReferencedMergeCommit
X-Amz-Date: 20190428T213222Z
User-Agent: aws-cli/1.16.137 Python/3.6.0 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "sourceCommitSpecifier": "bugfix-bug1234",
  "destinationCommitSpecifier": "main",
  "mergeOption": "THREE_WAY_MERGE",
  "authorName": "Maria Garcia",
  "email": "maria_garcia@example.com",
  "commitMessage": "Testing the merge of this bugfix branch to main."
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 220
Date: Sun, 28 Apr 2019 22:43:13 GMT

{
    "commitId": "4f178133EXAMPLE",
    "treeId": "389765daEXAMPLE"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DeleteApprovalRuleTemplate

Deletes a specified approval rule template. Deleting a template does not remove approval rules on pull requests already created with the template.

Request Syntax

```
{
   "approvalRuleTemplateName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

approvalRuleTemplateName

The name of the approval rule template to delete.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: Yes

Response Syntax

```
{
   "approvalRuleTemplateId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

approvalRuleTemplateId

The system-generated ID of the deleted approval rule template. If the template has been previously deleted, the only response is a 200 OK.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

ApprovalRuleTemplateInUseException

The approval rule template is associated with one or more repositories. You cannot delete a template that is associated with a repository. Remove all associations, and then try again.

HTTP Status Code: 400

ApprovalRuleTemplateNameRequiredException

An approval rule template name is required, but was not specified.

HTTP Status Code: 400

InvalidApprovalRuleTemplateNameException

The name of the approval rule template is not valid. Template names must be between 1 and 100 valid characters in length. For more information about limits in AWS CodeCommit, see Quotas in the *AWS CodeCommit User Guide*.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of DeleteApprovalRuleTemplate.

Sample Request

```
HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
```

```
Accept-Encoding: identity
Content-Length: 57
X-Amz-Target: CodeCommit_20150413.DeleteApprovalRuleTemplate
X-Amz-Date: 20191021T224659Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "approvalRuleTemplateName": "1-approver-for-all-pull-requests"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 48
Date: Mon, 21 Oct 2019 22:47:03 GMT

{
    "approvalRuleTemplateId": "41de97b7-EXAMPLE"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3

- AWS SDK for Python
- AWS SDK for Ruby V3

DeleteBranch

Deletes a branch from a repository, unless that branch is the default branch for the repository.

Request Syntax

```
{
   "branchName": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

branchName

The name of the branch to delete.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: Yes

repositoryName

The name of the repository that contains the branch to be deleted.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "deletedBranch": {
      "branchName": "string",
      "commitId": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

deletedBranch

Information about the branch deleted by the operation, including the branch name and the commit ID that was the tip of the branch.

Type: BranchInfo object

Errors

For information about the errors that are common to all actions, see Common Errors.

BranchNameRequiredException

A branch name is required, but was not specified.

HTTP Status Code: 400

DefaultBranchCannotBeDeletedException

The specified branch is the default branch for the repository, and cannot be deleted. To delete this branch, you must first set another branch as the default branch.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidBranchNameException

The specified reference name is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of DeleteBranch.

Sample Request

```
HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 57
X-Amz-Target: CodeCommit_20150413.DeleteBranch
X-Amz-Date: 20151028T224659Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "branchName": "MyNewBranch"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 88
Date: Wed, 28 Oct 2015 22:47:03 GMT

{
  "deletedBranch": {
    "branchName": "MyNewBranch",
    "commitId": "317f8570EXAMPLE"
  }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DeleteCommentContent

Deletes the content of a comment made on a change, file, or commit in a repository.

Request Syntax

```
{
   "commentId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

commentId

The unique, system-generated ID of the comment. To get this ID, use GetCommentsForComparedCommit or GetCommentsForPullRequest.

Type: String

Required: Yes

Response Syntax

```
{
   "comment": {
      "authorArn": "string",
      "callerReactions": [ "string" ],
      "clientRequestToken": "string",
      "commentId": "string",
      "content": "string",
      "creationDate": number,
      "deleted": boolean,
      "inReplyTo": "string",
      "lastModifiedDate": number,
      "reactionCounts": {
         "string" : number
```

```
        }
    }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

comment

Information about the comment you just deleted.

Type: Comment object

Errors

For information about the errors that are common to all actions, see Common Errors.

CommentDeletedException

This comment has already been deleted. You cannot edit or delete a deleted comment.

HTTP Status Code: 400

CommentDoesNotExistException

No comment exists with the provided ID. Verify that you have used the correct ID, and then try again.

HTTP Status Code: 400

CommentIdRequiredException

The comment ID is missing or null. A comment ID is required.

HTTP Status Code: 400

InvalidCommentIdException

The comment ID is not in a valid format. Make sure that you have provided the full comment ID.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of DeleteCommentContent.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.DeleteCommentContent
X-Amz-Date: 20171025T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "commentId": "ff30b348EXAMPLEb9aa670f"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Wed, 25 Oct 2017 20:20:13 GMT

{
   "comment": {
      "creationDate": 1508369768.142,
      "deleted": true,
      "lastModifiedDate": 1508369842.278,
      "clientRequestToken": "123Example",
      "commentId": "ff30b348EXAMPLEb9aa670f",
      "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
      "callerReactions": [],
      "reactionCounts":
         {
           "CLAP" : 1
         }
   }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DeleteFile

Deletes a specified file from a specified branch. A commit is created on the branch that contains the revision. The file still exists in the commits earlier to the commit that contains the deletion.

Request Syntax

```
{
   "branchName": "string",
   "commitMessage": "string",
   "email": "string",
   "filePath": "string",
   "keepEmptyFolders": boolean,
   "name": "string",
   "parentCommitId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

branchName

The name of the branch where the commit that deletes the file is made.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: Yes

commitMessage

The commit message you want to include as part of deleting the file. Commit messages are limited to 256 KB. If no message is specified, a default message is used.

Type: String

Required: No

email

The email address for the commit that deletes the file. If no email address is specified, the email address is left blank.

Type: String

Required: No

filePath

The fully qualified path to the file that to be deleted, including the full name and extension of that file. For example, /examples/file.md is a fully qualified path to a file named file.md in a folder named examples.

Type: String

Required: Yes

keepEmptyFolders

If a file is the only object in the folder or directory, specifies whether to delete the folder or directory that contains the file. By default, empty folders are deleted. This includes empty folders that are part of the directory structure. For example, if the path to a file is dir1/dir2/dir3/dir4, and dir2 and dir3 are empty, deleting the last file in dir4 also deletes the empty folders dir4, dir3, and dir2.

Type: Boolean

Required: No

name

The name of the author of the commit that deletes the file. If no name is specified, the user's ARN is used as the author name and committer name.

Type: String

Required: No

parentCommitId

The ID of the commit that is the tip of the branch where you want to create the commit that deletes the file. This must be the HEAD commit for the branch. The commit that deletes the file is created from this commit ID.

Type: String

Required: Yes

repositoryName

The name of the repository that contains the file to delete.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "blobId": "string",
   "commitId": "string",
   "filePath": "string",
   "treeId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

blobId

The blob ID removed from the tree as part of deleting the file.

Type: String

commitId

The full commit ID of the commit that contains the change that deletes the file.

Type: String

filePath

The fully qualified path to the file to be deleted, including the full name and extension of that file.

Type: String

treeId

The full SHA-1 pointer of the tree information for the commit that contains the delete file change.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

BranchDoesNotExistException

The specified branch does not exist.

HTTP Status Code: 400

BranchNameIsTagNameException

The specified branch name is not valid because it is a tag name. Enter the name of a branch in the repository. For a list of valid branch names, use ListBranches.

HTTP Status Code: 400

BranchNameRequiredException

A branch name is required, but was not specified.

HTTP Status Code: 400

CommitMessageLengthExceededException

The commit message is too long. Provide a shorter string.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

FileDoesNotExistException

The specified file does not exist. Verify that you have used the correct file name, full path, and extension.

HTTP Status Code: 400

InvalidBranchNameException

The specified reference name is not valid.

HTTP Status Code: 400

InvalidEmailException

The specified email address either contains one or more characters that are not allowed, or it exceeds the maximum number of characters allowed for an email address.

HTTP Status Code: 400

InvalidParentCommitIdException

The parent commit ID is not valid. The commit ID cannot be empty, and must match the head commit ID for the branch of the repository where you want to add or update a file.

HTTP Status Code: 400

InvalidPathException

The specified path is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

NameLengthExceededException

The user name is not valid because it has exceeded the character limit for author names.

HTTP Status Code: 400

ParentCommitDoesNotExistException

The parent commit ID is not valid because it does not exist. The specified parent commit ID does not exist in the specified branch of the repository.

HTTP Status Code: 400

ParentCommitIdOutdatedException

The file could not be added because the provided parent commit ID is not the current tip of the specified branch. To view the full commit ID of the current head of the branch, use GetBranch.

HTTP Status Code: 400

ParentCommitIdRequiredException

A parent commit ID is required. To view the full commit ID of a branch in a repository, use GetBranch or a Git command (for example, git pull or git log).

HTTP Status Code: 400

PathRequiredException

The folderPath for a location cannot be null.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of DeleteFile.

Sample Request

```
POST / HTTP/1.1
    Host: codecommit.us-east-1.amazonaws.com
    Accept-Encoding: identity
    Content-Length: 182
    X-Amz-Target: CodeCommit_20150413.DeleteFile
    X-Amz-Date: 20180914T223454Z
    User-Agent: aws-cli/1.15.9 Python/2.7.11 Darwin/16.7.0 botocore/1.10.9
    Content-Type: application/x-amz-json-1.1
    Authorization: AWS4-HMAC-SHA256 Credential=AKIAEXAMPLE/20180914/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
 Signature=0be88b13EXAMPLE
```

```
{
      "branchName": "main",
      "repositoryName": "MyDemoRepo",
      "parentCommitId": "c5709475EXAMPLE",
      "filePath": "README.md"
}
```

Sample Response

```
HTTP/1.1 200 OK
    x-amzn-RequestId: 66c7bf89-EXAMPLE
    Content-Type: application/x-amz-json-1.1
    Content-Length: 182
    Date: Fri, 14 Sep 2018 22:34:56 GMT
    Connection: keep-alive

{
    "blobId":"559b44fEXAMPLE",
    "commitId":"353cf655EXAMPLE",
    "filePath":"README.md",
    "treeId":"6bc824cEXAMPLE"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python

- AWS SDK for Ruby V3

DeletePullRequestApprovalRule

Deletes an approval rule from a specified pull request. Approval rules can be deleted from a pull request only if the pull request is open, and if the approval rule was created specifically for a pull request and not generated from an approval rule template associated with the repository where the pull request was created. You cannot delete an approval rule from a merged or closed pull request.

Request Syntax

```
{
   "approvalRuleName": "string",
   "pullRequestId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

approvalRuleName

The name of the approval rule you want to delete.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: Yes

pullRequestId

The system-generated ID of the pull request that contains the approval rule you want to delete.

Type: String

Required: Yes

Response Syntax

```
{
   "approvalRuleId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

approvalRuleId

The ID of the deleted approval rule.

> **Note**
>
> If the approval rule was deleted in an earlier API call, the response is 200 OK without content.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

ApprovalRuleNameRequiredException

An approval rule name is required, but was not specified.

HTTP Status Code: 400

CannotDeleteApprovalRuleFromTemplateException

The approval rule cannot be deleted from the pull request because it was created by an approval rule template and applied to the pull request automatically.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidApprovalRuleNameException

The name for the approval rule is not valid.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

PullRequestAlreadyClosedException

The pull request status cannot be updated because it is already closed.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of DeletePullRequestApprovalRule.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 31
X-Amz-Target: CodeCommit_20150413.DeletePullRequestApprovalRule
X-Amz-Date: 20191021T225354Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "pullRequestId": "15",
  "approvalRuleName": "My Approval Rule"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 55
Date: Mon, 21 Oct 2019 22:53:56 GMT

{
    "approvalRuleId": "077d8e8a8-EXAMPLE"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DeleteRepository

Deletes a repository. If a specified repository was already deleted, a null repository ID is returned.

> ⚠ **Important**
>
> Deleting a repository also deletes all associated objects and metadata. After a repository is deleted, all future push calls to the deleted repository fail.

Request Syntax

```
{
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

repositoryName

The name of the repository to delete.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "repositoryId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

repositoryId

The ID of the repository that was deleted.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of DeleteRepository.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 31
X-Amz-Target: CodeCommit_20150413.DeleteRepository
X-Amz-Date: 20151028T225354Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 55
Date: Wed, 28 Oct 2015 22:53:56 GMT

{
  "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DescribeMergeConflicts

Returns information about one or more merge conflicts in the attempted merge of two commit specifiers using the squash or three-way merge strategy. If the merge option for the attempted merge is specified as FAST_FORWARD_MERGE, an exception is thrown.

Request Syntax

```
{
   "conflictDetailLevel": "string",
   "conflictResolutionStrategy": "string",
   "destinationCommitSpecifier": "string",
   "filePath": "string",
   "maxMergeHunks": number,
   "mergeOption": "string",
   "nextToken": "string",
   "repositoryName": "string",
   "sourceCommitSpecifier": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

conflictDetailLevel

The level of conflict detail to use. If unspecified, the default FILE_LEVEL is used, which returns a not-mergeable result if the same file has differences in both branches. If LINE_LEVEL is specified, a conflict is considered not mergeable if the same file in both branches has differences on the same line.

Type: String

Valid Values: `FILE_LEVEL | LINE_LEVEL`

Required: No

conflictResolutionStrategy

Specifies which branch to use when resolving conflicts, or whether to attempt automatically merging two versions of a file. The default is NONE, which requires any conflicts to be resolved manually before the merge operation is successful.

Type: String

Valid Values: `NONE | ACCEPT_SOURCE | ACCEPT_DESTINATION | AUTOMERGE`

Required: No

destinationCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

filePath

The path of the target files used to describe the conflicts.

Type: String

Required: Yes

maxMergeHunks

The maximum number of merge hunks to include in the output.

Type: Integer

Required: No

mergeOption

The merge option or strategy you want to use to merge the code.

Type: String

Valid Values: `FAST_FORWARD_MERGE | SQUASH_MERGE | THREE_WAY_MERGE`

Required: Yes

nextToken

An enumeration token that, when provided in a request, returns the next batch of the results.

Type: String

Required: No

repositoryName

The name of the repository where you want to get information about a merge conflict.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

sourceCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

Response Syntax

```
{
   "baseCommitId": "string",
   "conflictMetadata": {
      "contentConflict": boolean,
      "fileModeConflict": boolean,
      "fileModes": {
         "base": "string",
         "destination": "string",
         "source": "string"
      },
      "filePath": "string",
      "fileSizes": {
         "base": number,
         "destination": number,
         "source": number
      },
      "isBinaryFile": {
         "base": boolean,
         "destination": boolean,
         "source": boolean
      },
      "mergeOperations": {
         "destination": "string",
         "source": "string"
      },
      "numberOfConflicts": number,
      "objectTypeConflict": boolean,
      "objectTypes": {
         "base": "string",
         "destination": "string",
         "source": "string"
      }
   },
   "destinationCommitId": "string",
   "mergeHunks": [
      {
         "base": {
            "endLine": number,
            "hunkContent": "string",
            "startLine": number
         },
         "destination": {
            "endLine": number,
            "hunkContent": "string",
            "startLine": number
         },
         "isConflict": boolean,
         "source": {
            "endLine": number,
            "hunkContent": "string",
            "startLine": number
         }
      }
   ],
   "nextToken": "string",
   "sourceCommitId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

baseCommitId

The commit ID of the merge base.

Type: String

conflictMetadata

Contains metadata about the conflicts found in the merge.

Type: ConflictMetadata object

destinationCommitId

The commit ID of the destination commit specifier that was used in the merge evaluation.

Type: String

mergeHunks

A list of merge hunks of the differences between the files or lines.

Type: Array of MergeHunk objects

nextToken

An enumeration token that can be used in a request to return the next batch of the results.

Type: String

sourceCommitId

The commit ID of the source commit specifier that was used in the merge evaluation.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

CommitRequiredException

A commit was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

FileDoesNotExistException

The specified file does not exist. Verify that you have used the correct file name, full path, and extension.

HTTP Status Code: 400

InvalidCommitException

The specified commit is not valid.

HTTP Status Code: 400

InvalidConflictDetailLevelException

The specified conflict detail level is not valid.

HTTP Status Code: 400

InvalidConflictResolutionStrategyException

The specified conflict resolution strategy is not valid.

HTTP Status Code: 400

InvalidContinuationTokenException

The specified continuation token is not valid.

HTTP Status Code: 400

InvalidMaxMergeHunksException

The specified value for the number of merge hunks to return is not valid.

HTTP Status Code: 400

InvalidMergeOptionException

The specified merge option is not valid for this operation. Not all merge strategies are supported for all operations.

HTTP Status Code: 400

InvalidPathException

The specified path is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

MaximumFileContentToLoadExceededException

The number of files to load exceeds the allowed limit.

HTTP Status Code: 400

MaximumItemsToCompareExceededException

The number of items to compare between the source or destination branches and the merge base has exceeded the maximum allowed.

HTTP Status Code: 400

MergeOptionRequiredException

A merge option or stategy is required, and none was provided.

HTTP Status Code: 400

PathRequiredException

The folderPath for a location cannot be null.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

TipsDivergenceExceededException

The divergence between the tips of the provided commit specifiers is too great to determine whether there might be any merge conflicts. Locally compare the specifiers using `git diff` or a diff tool.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of DescribeMergeConflicts.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 31
X-Amz-Target: CodeCommit_20150413.DescribeMergeConflicts
X-Amz-Date: 20190428T213222Z
User-Agent: aws-cli/1.16.137 Python/3.6.0 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "sourceCommitSpecifier": "bugfix-bug1234",
  "destinationCommitSpecifier": "main",
  "mergeOption": "THREE_WAY_MERGE",
  "filePath": "/results"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 55
```

```
Date: Sun, 28 Apr 2019 22:43:13 GMT

{
    "sourceCommitId": "c5709475EXAMPLE",
    "destinationCommitId": "317f8570EXAMPLE",
    "baseCommitId": "fb12a539EXAMPLE",
    "conflictMetadata": {
           "filePath": "file1.py",
           "fileSizes": {
             "source": 123,
             "destination": 125,
             "base": 124
        },
        "fileModes": {
            "source": "EXECUTABLE",
            "destination": "EXECUTABLE",
            "base": "EXECUTABLE"
            },
        "numberOfConflicts", 4,
        "isBinaryFile": {
            "source": false,
            "destination": false,
            "base": false
        },
        "contentConflict": true,
        "fileModeConflict": false,
        "mergeOperations": {
            "source": "M",
            "destination": "M"
        }
    "mergeHunks":[
        {
            "mergeHunk": {
                "isConflict": true
                "source": {
                    "startLine": 123,
                    "endLine": 123,
                    "hunkContent" "JzCQbIVyEXAMPLE="
                }
                "destination": {
                    "startLine": 125,
                    "endLine": 125,
                    "hunkContent" "BytPbuMiEXAMPLE="
                }
                "base": {
                    "startLine": 124,
                    "endLine": 124,
                    "hunkContent" "MnKCdITaEXAMPLE="
                }
            }
        }
    ]
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DescribePullRequestEvents

Returns information about one or more pull request events.

Request Syntax

```
{
   "actorArn": "string",
   "maxResults": number,
   "nextToken": "string",
   "pullRequestEventType": "string",
   "pullRequestId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

actorArn

The Amazon Resource Name (ARN) of the user whose actions resulted in the event. Examples include updating the pull request with more commits or changing the status of a pull request.

Type: String

Required: No

maxResults

A non-zero, non-negative integer used to limit the number of returned results. The default is 100 events, which is also the maximum number of events that can be returned in a result.

Type: Integer

Required: No

nextToken

An enumeration token that, when provided in a request, returns the next batch of the results.

Type: String

Required: No

pullRequestEventType

Optional. The pull request event type about which you want to return information.

Type: String

Valid Values: `PULL_REQUEST_CREATED | PULL_REQUEST_STATUS_CHANGED | PULL_REQUEST_SOURCE_REFERENCE_UPDATED | PULL_REQUEST_MERGE_STATE_CHANGED | PULL_REQUEST_APPROVAL_RULE_CREATED | PULL_REQUEST_APPROVAL_RULE_UPDATED | PULL_REQUEST_APPROVAL_RULE_DELETED | PULL_REQUEST_APPROVAL_RULE_OVERRIDDEN | PULL_REQUEST_APPROVAL_STATE_CHANGED`

Required: No

pullRequestId

The system-generated ID of the pull request. To get this ID, use ListPullRequests.

Type: String

Required: Yes

Response Syntax

```
{
   "nextToken": "string",
   "pullRequestEvents": [
      {
         "actorArn": "string",
         "approvalRuleEventMetadata": {
            "approvalRuleContent": "string",
            "approvalRuleId": "string",
            "approvalRuleName": "string"
         },
         "approvalRuleOverriddenEventMetadata": {
            "overrideStatus": "string",
            "revisionId": "string"
         },
         "approvalStateChangedEventMetadata": {
            "approvalStatus": "string",
            "revisionId": "string"
         },
         "eventDate": number,
         "pullRequestCreatedEventMetadata": {
            "destinationCommitId": "string",
            "mergeBase": "string",
            "repositoryName": "string",
            "sourceCommitId": "string"
         },
         "pullRequestEventType": "string",
         "pullRequestId": "string",
         "pullRequestMergedStateChangedEventMetadata": {
            "destinationReference": "string",
            "mergeMetadata": {
               "isMerged": boolean,
               "mergeCommitId": "string",
               "mergedBy": "string",
               "mergeOption": "string"
            },
            "repositoryName": "string"
         },
         "pullRequestSourceReferenceUpdatedEventMetadata": {
            "afterCommitId": "string",
            "beforeCommitId": "string",
            "mergeBase": "string",
            "repositoryName": "string"
         },
         "pullRequestStatusChangedEventMetadata": {
            "pullRequestStatus": "string"
         }
      }
   ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

nextToken

An enumeration token that can be used in a request to return the next batch of the results.

Type: String

pullRequestEvents

Information about the pull request events.

Type: Array of PullRequestEvent objects

Errors

For information about the errors that are common to all actions, see Common Errors.

ActorDoesNotExistException

The specified Amazon Resource Name (ARN) does not exist in the AWS account.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidActorArnException

The Amazon Resource Name (ARN) is not valid. Make sure that you have provided the full ARN for the user who initiated the change for the pull request, and then try again.

HTTP Status Code: 400

InvalidContinuationTokenException

The specified continuation token is not valid.

HTTP Status Code: 400

InvalidMaxResultsException

The specified number of maximum results is not valid.

HTTP Status Code: 400

InvalidPullRequestEventTypeException

The pull request event type is not valid.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of DescribePullRequestEvents.

Sample Request

```
>>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.DescribePullRequestEvents
X-Amz-Date: 20171110T235323Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "pullRequestId": "8"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 660
Date: Fri, 10 Nov 2017 11:53:59 GMT


{
    "pullRequestEvents": [
        {
            "pullRequestId": "8",
            "pullRequestEventType": "PULL_REQUEST_CREATED",
            "eventDate": 1510341779.53,
            "actor": "arn:aws:iam::123456789012:user/Zhang_Wei"
        },
        {
            "pullRequestStatusChangedEventMetadata": {
                "pullRequestStatus": "CLOSED"
            },
            "pullRequestId": "8",
            "pullRequestEventType": "PULL_REQUEST_STATUS_CHANGED",
            "eventDate": 1510341930.72,
            "actor": "arn:aws:iam::123456789012:user/Jane_Doe"
        }
    ]
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DisassociateApprovalRuleTemplateFromRepository

Removes the association between a template and a repository so that approval rules based on the template are not automatically created when pull requests are created in the specified repository. This does not delete any approval rules previously created for pull requests through the template association.

Request Syntax

```
{
   "approvalRuleTemplateName": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

approvalRuleTemplateName

The name of the approval rule template to disassociate from a specified repository.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: Yes

repositoryName

The name of the repository you want to disassociate from the template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors.

ApprovalRuleTemplateDoesNotExistException

The specified approval rule template does not exist. Verify that the name is correct and that you are signed in to the AWS Region where the template was created, and then try again.

HTTP Status Code: 400

ApprovalRuleTemplateNameRequiredException

An approval rule template name is required, but was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidApprovalRuleTemplateNameException

The name of the approval rule template is not valid. Template names must be between 1 and 100 valid characters in length. For more information about limits in AWS CodeCommit, see Quotas in the *AWS CodeCommit User Guide*.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of DisassociateApprovalRuleTemplateFromRepository.

Sample Request

```
POST / HTTP/1.1
```

```
    Host: codecommit.us-east-1.amazonaws.com
    Accept-Encoding: identity
    Content-Length: 124
    X-Amz-Target: CodeCommit_20150413.DisassociateApprovalRuleTemplateFromRepository
    X-Amz-Date: 20191021T223055Z
    User-Agent: aws-cli/1.15.9 Python/2.7.11 Darwin/16.7.0 botocore/1.10.9
    Content-Type: application/x-amz-json-1.1
    Authorization: AWS4-HMAC-SHA256 Credential=AKIAEXAMPLE/20180914/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
 Signature==8d9b5998EXAMPLE
{
   "repositoryName": "MyDemoRepo"
   "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests"
}
```

Sample Response

```
HTTP/1.1 200 OK
    x-amzn-RequestId: d8ad1d21-EXAMPLE
    Content-Type: application/x-amz-json-1.1
    Content-Length: 2267
    Date: Mon, 21 Oct 2019 22:30:56 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

EvaluatePullRequestApprovalRules

Evaluates whether a pull request has met all the conditions specified in its associated approval rules.

Request Syntax

```
{
   "pullRequestId": "string",
   "revisionId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

pullRequestId

The system-generated ID of the pull request you want to evaluate.

Type: String

Required: Yes

revisionId

The system-generated ID for the pull request revision. To retrieve the most recent revision ID for a pull request, use GetPullRequest.

Type: String

Required: Yes

Response Syntax

```
{
   "evaluation": {
      "approvalRulesNotSatisfied": [ "string" ],
      "approvalRulesSatisfied": [ "string" ],
      "approved": boolean,
      "overridden": boolean
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

evaluation

The result of the evaluation, including the names of the rules whose conditions have been met (if any), the names of the rules whose conditions have not been met (if any), whether the pull request is in the approved state, and whether the pull request approval rule has been set aside by an override.

Type: Evaluation object

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

InvalidRevisionIdException

The revision ID is not valid. Use GetPullRequest to determine the value.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

RevisionIdRequiredException

A revision ID is required, but was not provided.

HTTP Status Code: 400

RevisionNotCurrentException

The revision ID provided in the request does not match the current revision ID. Use GetPullRequest to retrieve the current revision ID.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of EvaluatePullRequestApprovalRules.

Sample Request

```
POST / HTTP/1.1
    Host: codecommit.us-east-1.amazonaws.com
    Accept-Encoding: identity
    Content-Length: 124
    X-Amz-Target: CodeCommit_20150413.EvaluatePullRequestApprovalRules
    X-Amz-Date: 20191021T223055Z
    User-Agent: aws-cli/1.15.9 Python/2.7.11 Darwin/16.7.0 botocore/1.10.9
    Content-Type: application/x-amz-json-1.1
    Authorization: AWS4-HMAC-SHA256 Credential=AKIAEXAMPLE/20180914/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
 Signature==8d9b5998EXAMPLE
{
   "pullRequestId": "27",
   "revisionId": "9f29d167EXAMPLE"
}
```

Sample Response

```
HTTP/1.1 200 OK
    x-amzn-RequestId: d8ad1d21-EXAMPLE
    Content-Type: application/x-amz-json-1.1
    Content-Length: 2267
    Date: Mon, 21 Oct 2019 22:30:56 GMT
    Connection: keep-alive

{
    "evaluation": {
        "approved": false,
        "approvalRulesNotSatisfied": [
            "Require two approved approvers"
        ],
        "overridden": false,
        "approvalRulesSatisfied": []
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetApprovalRuleTemplate

Returns information about a specified approval rule template.

Request Syntax

```
{
   "approvalRuleTemplateName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

approvalRuleTemplateName

The name of the approval rule template for which you want to get information.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: Yes

Response Syntax

```
{
   "approvalRuleTemplate": {
      "approvalRuleTemplateContent": "string",
      "approvalRuleTemplateDescription": "string",
      "approvalRuleTemplateId": "string",
      "approvalRuleTemplateName": "string",
      "creationDate": number,
      "lastModifiedDate": number,
      "lastModifiedUser": "string",
      "ruleContentSha256": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

approvalRuleTemplate

The content and structure of the approval rule template.

Type: ApprovalRuleTemplate object

Errors

For information about the errors that are common to all actions, see Common Errors.

ApprovalRuleTemplateDoesNotExistException

The specified approval rule template does not exist. Verify that the name is correct and that you are signed in to the AWS Region where the template was created, and then try again.

HTTP Status Code: 400

ApprovalRuleTemplateNameRequiredException

An approval rule template name is required, but was not specified.

HTTP Status Code: 400

InvalidApprovalRuleTemplateNameException

The name of the approval rule template is not valid. Template names must be between 1 and 100 valid characters in length. For more information about limits in AWS CodeCommit, see Quotas in the *AWS CodeCommit User Guide*.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetApprovalRuleTemplate.

Sample Request

```
POST / HTTP/1.1
    Host: codecommit.us-east-1.amazonaws.com
    Accept-Encoding: identity
    Content-Length: 124
    X-Amz-Target: CodeCommit_20150413.GetApprovalRuleTemplate
    X-Amz-Date: 20191021T223055Z
    User-Agent: aws-cli/1.15.9 Python/2.7.11 Darwin/16.7.0 botocore/1.10.9
    Content-Type: application/x-amz-json-1.1
    Authorization: AWS4-HMAC-SHA256 Credential=AKIAEXAMPLE/20180914/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
 Signature==8d9b5998EXAMPLE
{
   "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests"
}
```

Sample Response

```
HTTP/1.1 200 OK
    x-amzn-RequestId: d8ad1d21-EXAMPLE
    Content-Type: application/x-amz-json-1.1
    Content-Length: 2267
    Date: Mon, 21 Oct 2019 22:30:56 GMT
    Connection: keep-alive

{
    "approvalRuleTemplate": {
        "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\",
\"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "ruleContentSha256": "621181bbEXAMPLE",
        "lastModifiedDate": 1571356106.936,
        "creationDate": 1571356106.936,
        "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Li_Juan",
        "approvalRuleTemplateId": "a29abb15-EXAMPLE",
        "approvalRuleTemplateDescription": "All pull requests must be approved by one
 developer on the team."
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetBlob

Returns the base-64 encoded content of an individual blob in a repository.

Request Syntax

```
{
   "blobId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

blobId

The ID of the blob, which is its SHA-1 pointer.

Type: String

Required: Yes

repositoryName

The name of the repository that contains the blob.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "content": blob
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

content

The content of the blob, usually a file.

Type: Base64-encoded binary data object

Errors

For information about the errors that are common to all actions, see Common Errors.

BlobIdDoesNotExistException

The specified blob does not exist.

HTTP Status Code: 400

BlobIdRequiredException

A blob ID is required, but was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

FileTooLargeException

The specified file exceeds the file size limit for AWS CodeCommit. For more information about limits in AWS CodeCommit, see Quotas in the *AWS CodeCommit User Guide*.

HTTP Status Code: 400

InvalidBlobIdException

The specified blob is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetBranch

Returns information about a repository branch, including its name and the last commit ID.

Request Syntax

```
{
   "branchName": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

branchName

The name of the branch for which you want to retrieve information.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

repositoryName

The name of the repository that contains the branch for which you want to retrieve information.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: No

Response Syntax

```
{
   "branch": {
      "branchName": "string",
      "commitId": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

branch

The name of the branch.

Type: BranchInfo object

Errors

For information about the errors that are common to all actions, see Common Errors.

BranchDoesNotExistException

The specified branch does not exist.

HTTP Status Code: 400

BranchNameRequiredException

A branch name is required, but was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidBranchNameException

The specified reference name is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetBranch.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 57
X-Amz-Target: CodeCommit_20150413.GetBranch
X-Amz-Date: 20151028T224311Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "branchName": "MyNewBranch"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 88
Date: Wed, 28 Oct 2015 22:43:13 GMT

{
  "branch": {
    "commitId": "317f8570EXAMPLE",
    "branchName": "MyNewBranch"
  }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetComment

Returns the content of a comment made on a change, file, or commit in a repository.

> **Note**
>
> Reaction counts might include numbers from user identities who were deleted after the reaction was made. For a count of reactions from active identities, use GetCommentReactions.

Request Syntax

```
{
   "commentId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

commentId

The unique, system-generated ID of the comment. To get this ID, use GetCommentsForComparedCommit or GetCommentsForPullRequest.

Type: String

Required: Yes

Response Syntax

```
{
   "comment": {
      "authorArn": "string",
      "callerReactions": [ "string" ],
      "clientRequestToken": "string",
      "commentId": "string",
      "content": "string",
      "creationDate": number,
      "deleted": boolean,
      "inReplyTo": "string",
      "lastModifiedDate": number,
      "reactionCounts": { 
         "string" : number 
      }
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

comment

The contents of the comment.

Type: Comment object

Errors

For information about the errors that are common to all actions, see Common Errors.

CommentDeletedException

This comment has already been deleted. You cannot edit or delete a deleted comment.

HTTP Status Code: 400

CommentDoesNotExistException

No comment exists with the provided ID. Verify that you have used the correct ID, and then try again.

HTTP Status Code: 400

CommentIdRequiredException

The comment ID is missing or null. A comment ID is required.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidCommentIdException

The comment ID is not in a valid format. Make sure that you have provided the full comment ID.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetComment.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.GetComment
X-Amz-Date: 20171025T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "commentId": "ff30b348EXAMPLEb9aa670f"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Wed, 25 Oct 2017 20:18:13 GMT

{
   "comment": {
      "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
      "clientRequestToken": "123Example",
      "commentId": "ff30b348EXAMPLEb9aa670f",
      "content": "Whoops - I meant to add this comment to the line, but I don't see how
 to delete it.",
      "creationDate": 1508369768.142,
      "deleted": false,
      "commentId": "",
      "lastModifiedDate": 1508369842.278,
      "reactionCounts":{
         "THUMBSUP": 20,
         "THUMBSDOWN": 2,
         "SMILE": 5,
         "ANGRY": 7
        }
```

```
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetCommentReactions

Returns information about reactions to a specified comment ID. Reactions from users who have been deleted will not be included in the count.

Request Syntax

```
{
   "commentId": "string",
   "maxResults": number,
   "nextToken": "string",
   "reactionUserArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

commentId

The ID of the comment for which you want to get reactions information.

Type: String

Required: Yes

maxResults

A non-zero, non-negative integer used to limit the number of returned results. The default is the same as the allowed maximum, 1,000.

Type: Integer

Required: No

nextToken

An enumeration token that, when provided in a request, returns the next batch of the results.

Type: String

Required: No

reactionUserArn

Optional. The Amazon Resource Name (ARN) of the user or identity for which you want to get reaction information.

Type: String

Required: No

Response Syntax

```
{
   "nextToken": "string",
   "reactionsForComment": [
      {
         "reaction": {
            "emoji": "string",
            "shortCode": "string",
            "unicode": "string"
         },
         "reactionsFromDeletedUsersCount": number,
         "reactionUsers": [ "string" ]
      }
   ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

nextToken

An enumeration token that can be used in a request to return the next batch of the results.

Type: String

reactionsForComment

An array of reactions to the specified comment.

Type: Array of ReactionForComment objects

Errors

For information about the errors that are common to all actions, see Common Errors.

CommentDeletedException

This comment has already been deleted. You cannot edit or delete a deleted comment.

HTTP Status Code: 400

CommentDoesNotExistException

No comment exists with the provided ID. Verify that you have used the correct ID, and then try again.

HTTP Status Code: 400

CommentIdRequiredException

The comment ID is missing or null. A comment ID is required.

HTTP Status Code: 400

InvalidCommentIdException

The comment ID is not in a valid format. Make sure that you have provided the full comment ID.

HTTP Status Code: 400

InvalidContinuationTokenException

The specified continuation token is not valid.

HTTP Status Code: 400

InvalidMaxResultsException

The specified number of maximum results is not valid.

HTTP Status Code: 400

InvalidReactionUserArnException

The Amazon Resource Name (ARN) of the user or identity is not valid.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetCommentReactions.

Sample Request

```
POST / HTTP/1.1
    Host: codecommit.us-east-1.amazonaws.com
    Accept-Encoding: identity
    Content-Length: 124
    X-Amz-Target: CodeCommit_20150413.GetCommentReactions
    X-Amz-Date: 20200406T223055Z
    User-Agent: aws-cli/1.18.36 Python/3.6.10
 Linux/4.9.184-0.1.ac.235.83.329.metal1.x86_64 botocore/1.15.36
    Content-Type: application/x-amz-json-1.1
    Authorization: AWS4-HMAC-SHA256 Credential=AKIAEXAMPLE/20180914/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
 Signature==8d9b5998EXAMPLE
{
   "commentId": "ff30b348EXAMPLE1EXAMPLE"
}
```

Sample Response

```
HTTP/1.1 200 OK
    x-amzn-RequestId: d8ad1d21-EXAMPLE
    Content-Type: application/x-amz-json-1.1
    Content-Length: 2267
    Date: Mon, 6 Apr 2020 22:30:56 GMT
    Connection: keep-alive

{
    "reactionsForComment": {
        [
           {
                "reaction": {
                    "emoji:"#",
                    "shortCode": "thumbsup",
                    "unicode": "U+1F44D"
                },
                "users": [
                    "arn:aws:iam::123456789012:user/Li_Juan",
                    "arn:aws:iam::123456789012:user/Mary_Major",
                    "arn:aws:iam::123456789012:user/Jorge_Souza"
                ]
            },
            {
                "reaction": {
                    "emoji": "#",
                    "shortCode": "thumbsdown",
                    "unicode": "U+1F44E"
                },
                "users": [
                    "arn:aws:iam::123456789012:user/Nikhil_Jayashankar"
                ]
            },
            {
                "reaction": {
                    "emoji": "#",
                    "shortCode": "confused",
                    "unicode": "U+1F615"
                },
                "users": [
                    "arn:aws:iam::123456789012:user/Saanvi_Sarkar"
                ]
            }
        ]
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3

- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetCommentsForComparedCommit

Returns information about comments made on the comparison between two commits.

> **Note**
>
> Reaction counts might include numbers from user identities who were deleted after the reaction was made. For a count of reactions from active identities, use GetCommentReactions.

Request Syntax

```
{
   "afterCommitId": "string",
   "beforeCommitId": "string",
   "maxResults": number,
   "nextToken": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

afterCommitId

To establish the directionality of the comparison, the full commit ID of the after commit.

Type: String

Required: Yes

beforeCommitId

To establish the directionality of the comparison, the full commit ID of the before commit.

Type: String

Required: No

maxResults

A non-zero, non-negative integer used to limit the number of returned results. The default is 100 comments, but you can configure up to 500.

Type: Integer

Required: No

nextToken

An enumeration token that when provided in a request, returns the next batch of the results.

Type: String

Required: No

repositoryName

The name of the repository where you want to compare commits.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "commentsForComparedCommitData": [
      {
         "afterBlobId": "string",
         "afterCommitId": "string",
         "beforeBlobId": "string",
         "beforeCommitId": "string",
         "comments": [
            {
               "authorArn": "string",
               "callerReactions": [ "string" ],
               "clientRequestToken": "string",
               "commentId": "string",
               "content": "string",
               "creationDate": number,
               "deleted": boolean,
               "inReplyTo": "string",
               "lastModifiedDate": number,
               "reactionCounts": { 
                  "string" : number 
               }
            }
         ],
         "location": { 
            "filePath": "string",
            "filePosition": number,
            "relativeFileVersion": "string"
         },
         "repositoryName": "string"
      }
   ],
   "nextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

commentsForComparedCommitData

A list of comment objects on the compared commit.

Type: Array of CommentsForComparedCommit objects

nextToken

An enumeration token that can be used in a request to return the next batch of the results.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

CommitIdRequiredException

A commit ID was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidCommitIdException

The specified commit ID is not valid.

HTTP Status Code: 400

InvalidContinuationTokenException

The specified continuation token is not valid.

HTTP Status Code: 400

InvalidMaxResultsException

The specified number of maximum results is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetCommentsForComparedCommit.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 176
X-Amz-Target: CodeCommit_20150413.GetCommentsForComparedCommit
X-Amz-Date: 20171025T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "afterCommitId": "317f8570EXAMPLE",
   "beforeCommitId": "6e147360EXAMPLE",
   "maxResults": 250,
   "nextToken": "exampleToken",
   "repositoryName": "MyDemoRepo"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 2101
Date: Wed, 25 Oct 2017 20:18:13 GMT

{
   "commentsForComparedCommitData": [
      {
         "afterBlobId": "1f330709EXAMPLE",
         "afterCommitId": "317f8570EXAMPLE",
         "beforeBlobId": "80906a4cEXAMPLE",
         "beforeCommitId": "6e147360EXAMPLE",
         "comments": [
            {
               "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
               "clientRequestToken": "123Example",
               "commentId": "ff30b348EXAMPLEb9aa670f",
               "content": "Whoops - I meant to add this comment to the line, not the
 file, but I don't see how to delete it.",
               "creationDate": 1508369768.142,
               "deleted": false,
               "lastModifiedDate": 1508369842.278,
               "callerReactions": [],
               "reactionCounts":
                {
                  "THUMBSUP" : 1
                }
            },
            {
               "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
               "clientRequestToken": "123Example",
               "commentId": "553b509bEXAMPLE56198325",
               "content": "Can you add a test case for this?",
               "creationDate": 1508369612.240,
               "deleted": false,
               "commentId": "456def-EXAMPLE",
               "lastModifiedDate": 1508369612.240,
               "callerReactions": ["THUMBSUP"]
               "reactionCounts":
                {
                  "THUMBSUP" : 10,
                  "THUMBSDOWN" : 1,
                  "SMILE" : 2,
                  "HEART" : 2
                }
             }
         ],
         "location": {
            "filePath": "cl_sample.js",
            "filePosition": 1232,
            "relativeFileVersion": "after"
         },
         "repositoryName": "MyDemoRepo"
      }
   ],
   "nextToken": "exampleToken"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetCommentsForPullRequest

Returns comments made on a pull request.

> **Note**
>
> Reaction counts might include numbers from user identities who were deleted after the reaction was made. For a count of reactions from active identities, use GetCommentReactions.

Request Syntax

```
{
   "afterCommitId": "string",
   "beforeCommitId": "string",
   "maxResults": number,
   "nextToken": "string",
   "pullRequestId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

afterCommitId

The full commit ID of the commit in the source branch that was the tip of the branch at the time the comment was made. Requirement is conditional: `afterCommitId` must be specified when `repositoryName` is included.

Type: String

Required: No

beforeCommitId

The full commit ID of the commit in the destination branch that was the tip of the branch at the time the pull request was created. Requirement is conditional: `beforeCommitId` must be specified when `repositoryName` is included.

Type: String

Required: No

maxResults

A non-zero, non-negative integer used to limit the number of returned results. The default is 100 comments. You can return up to 500 comments with a single request.

Type: Integer

Required: No

nextToken

An enumeration token that, when provided in a request, returns the next batch of the results.

Type: String

Required: No

pullRequestId

The system-generated ID of the pull request. To get this ID, use ListPullRequests.

Type: String

Required: Yes

repositoryName

The name of the repository that contains the pull request. Requirement is conditional: `repositoryName` must be specified when `beforeCommitId` and `afterCommitId` are included.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: No

Response Syntax

```
{
   "commentsForPullRequestData": [
      {
         "afterBlobId": "string",
         "afterCommitId": "string",
         "beforeBlobId": "string",
         "beforeCommitId": "string",
         "comments": [
            {
               "authorArn": "string",
               "callerReactions": [ "string" ],
               "clientRequestToken": "string",
               "commentId": "string",
               "content": "string",
               "creationDate": number,
               "deleted": boolean,
               "inReplyTo": "string",
               "lastModifiedDate": number,
               "reactionCounts": {
                  "string" : number
               }
            }
         ],
         "location": {
            "filePath": "string",
            "filePosition": number,
            "relativeFileVersion": "string"
         },
         "pullRequestId": "string",
         "repositoryName": "string"
      }
   ],
   "nextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

commentsForPullRequestData

An array of comment objects on the pull request.

Type: Array of CommentsForPullRequest objects

nextToken

An enumeration token that can be used in a request to return the next batch of the results.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

CommitIdRequiredException

A commit ID was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidCommitIdException

The specified commit ID is not valid.

HTTP Status Code: 400

InvalidContinuationTokenException

The specified continuation token is not valid.

HTTP Status Code: 400

InvalidMaxResultsException

The specified number of maximum results is not valid.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

RepositoryNotAssociatedWithPullRequestException

The repository does not contain any pull requests with that pull request ID. Use GetPullRequest to verify the correct repository name for the pull request ID.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetCommentsForPullRequest.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
```

```
Content-Length: 176
X-Amz-Target: CodeCommit_20150413.GetCommentsForPullRequest
X-Amz-Date: 20171025T134323Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "afterCommitId": "5d036259EXAMPLE",
   "beforeCommitId": "317f8570EXAMPLE",
   "maxResults": 250,
   "nextToken": "exampleToken",
   "pullRequestId": 42,
   "repositoryName": "MyDemoRepo"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 2101
Date: Wed, 25 Oct 2017 20:18:13 GMT

{
   "commentsForPullRequestData": [
      {
         "afterBlobId": "1f330709EXAMPLE",
         "afterCommitId": "5d036259EXAMPLE",
         "beforeBlobId": "80906a4cEXAMPLE",
         "beforeCommitId": "317f8570EXAMPLE",
         "comments": [
            {
               "authorArn": "arn:aws:iam::123456789012:user/Saanvi_Sarkar",
               "clientRequestToken": "",
               "commentId": "abcd1234EXAMPLEb5678efgh",
               "content": "These don't appear to be used anywhere. Can we remove
 them?",
               "creationDate": 1508369622.123,
               "deleted": false,
               "lastModifiedDate": 1508369622.123,
               "callerReactions": [],
               "reactionCounts":
                {
                  "THUMBSUP" : 6,
                  "CONFUSED" : 1
                }
            },
            {
               "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
               "clientRequestToken": "",
               "commentId": "442b498bEXAMPLE5756813",
               "content": "Good catch. I'll remove them.",
               "creationDate": 1508369829.104,
               "deleted": false,
               "commentId": "abcd1234EXAMPLEb5678efgh",
               "lastModifiedDate": 150836912.273,
               "callerReactions": ["THUMBSUP],
               "reactionCounts":
                {
                  "THUMBSUP" : 14
                }
             }
         ],
         "location": {
            "filePath": "ahs_count.py",
            "filePosition": 367,
            "relativeFileVersion": "AFTER"
         },
         "repositoryName": "MyDemoRepo",
         "pullRequestId": "42"
      }
   ],
   "nextToken": "exampleToken"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4

- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetCommit

Returns information about a commit, including commit message and committer information.

Request Syntax

```
{
   "commitId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

commitId

The commit ID. Commit IDs are the full SHA ID of the commit.

Type: String

Required: Yes

repositoryName

The name of the repository to which the commit was made.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "commit": {
      "additionalData": "string",
      "author": {
         "date": "string",
         "email": "string",
         "name": "string"
      },
      "commitId": "string",
      "committer": {
         "date": "string",
         "email": "string",
         "name": "string"
      },
      "message": "string",
      "parents": [ "string" ],
      "treeId": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

commit

A commit data type object that contains information about the specified commit.

Type: Commit object

Errors

For information about the errors that are common to all actions, see Common Errors.

CommitIdDoesNotExistException

The specified commit ID does not exist.

HTTP Status Code: 400

CommitIdRequiredException

A commit ID was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidCommitIdException

The specified commit ID is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetCommit.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 57
X-Amz-Target: CodeCommit_20150413.GetCommit
X-Amz-Date: 20170111T224311Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "commitId": "12345678EXAMPLE"
}
```

Sample Response

```
HTTP/1.1 200 OK
```

```
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 720
Date: Wed, 11 Jan 2017 22:43:13 GMT

{
    "commit": {
        "commitId": "12345678EXAMPLE",
        "additionalData": "",
        "committer": {
            "date": "1484167798 -0800",
            "name": "Mary Major",
            "email": "mary_major@example.com"
        },
        "author": {
            "date": "1484167798 -0800",
            "name": "Mary Major",
            "email": "mary_major@example.com"
        },
        "treeId": "347a3408EXAMPLE",
        "parents": [
            "7aa87a0EXAMPLE"
        ],
        "message": "Fix incorrect variable name\n"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3

- AWS SDK for Python
- AWS SDK for Ruby V3

GetDifferences

Returns information about the differences in a valid commit specifier (such as a branch, tag, HEAD, commit ID, or other fully qualified reference). Results can be limited to a specified path.

Request Syntax

```
{
   "afterCommitSpecifier": "string",
   "afterPath": "string",
   "beforeCommitSpecifier": "string",
   "beforePath": "string",
   "MaxResults": number,
   "NextToken": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

afterCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit.

Type: String

Required: Yes

afterPath

The file path in which to check differences. Limits the results to this path. Can also be used to specify the changed name of a directory or folder, if it has changed. If not specified, differences are shown for all paths.

Type: String

Required: No

beforeCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, the full commit ID). Optional. If not specified, all changes before the `afterCommitSpecifier` value are shown. If you do not use `beforeCommitSpecifier` in your request, consider limiting the results with `maxResults`.

Type: String

Required: No

beforePath

The file path in which to check for differences. Limits the results to this path. Can also be used to specify the previous name of a directory or folder. If `beforePath` and `afterPath` are not specified, differences are shown for all paths.

Type: String

Required: No

MaxResults

A non-zero, non-negative integer used to limit the number of returned results.

Type: Integer

Required: No

NextToken

An enumeration token that, when provided in a request, returns the next batch of the results.

Type: String

Required: No

repositoryName

The name of the repository where you want to get differences.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "differences": [
      {
         "afterBlob": {
            "blobId": "string",
            "mode": "string",
            "path": "string"
         },
         "beforeBlob": {
            "blobId": "string",
            "mode": "string",
            "path": "string"
         },
         "changeType": "string"
      }
   ],
   "NextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

differences

A data type object that contains information about the differences, including whether the difference is added, modified, or deleted (A, D, M).

Type: Array of Difference objects

NextToken

An enumeration token that can be used in a request to return the next batch of the results.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

CommitRequiredException

A commit was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidCommitException

The specified commit is not valid.

HTTP Status Code: 400

InvalidCommitIdException

The specified commit ID is not valid.

HTTP Status Code: 400

InvalidContinuationTokenException

The specified continuation token is not valid.

HTTP Status Code: 400

InvalidMaxResultsException

The specified number of maximum results is not valid.

HTTP Status Code: 400

InvalidPathException

The specified path is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

PathDoesNotExistException

The specified path does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetDifferences.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 57
X-Amz-Target: CodeCommit_20150413.GetDifferences
X-Amz-Date: 20170111T224311Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "beforeCommitSpecifier": "16d097f03EXAMPLE",
  "afterCommitSpecifier": "fac04518EXAMPLE"
  "beforePath": "tmp/example-folder"
  "afterPath": "tmp/renamed-folder",

}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 770
Date: Wed, 11 Jan 2017 22:43:13 GMT

{
    "differences": [
        {
            "afterBlob": {
                "path": "blob.txt",
                "blobId": "2eb4af3bEXAMPLE",
                "mode": "100644"
            },
            "changeType": "M",
            "beforeBlob": {
                "path": "blob.txt",
                "blobId": "bf7fcf28fEXAMPLE",
                "mode": "100644"
            }
        }
    ]
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3

- AWS SDK for Python
- AWS SDK for Ruby V3

GetFile

Returns the base-64 encoded contents of a specified file and its metadata.

Request Syntax

```
{
   "commitSpecifier": "string",
   "filePath": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

commitSpecifier

The fully quaified reference that identifies the commit that contains the file. For example, you can specify a full commit ID, a tag, a branch name, or a reference such as refs/heads/main. If none is provided, the head commit is used.

Type: String

Required: No

filePath

The fully qualified path to the file, including the full name and extension of the file. For example, /examples/file.md is the fully qualified path to a file named file.md in a folder named examples.

Type: String

Required: Yes

repositoryName

The name of the repository that contains the file.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "blobId": "string",
   "commitId": "string",
   "fileContent": blob,
   "fileMode": "string",
   "filePath": "string",
   "fileSize": number
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

blobId

The blob ID of the object that represents the file content.

Type: String

commitId

The full commit ID of the commit that contains the content returned by GetFile.

Type: String

fileContent

The base-64 encoded binary data object that represents the content of the file.

Type: Base64-encoded binary data object

Length Constraints: Maximum length of 6291456.

fileMode

The extrapolated file mode permissions of the blob. Valid values include strings such as EXECUTABLE and not numeric values.

> **Note**
>
> The file mode permissions returned by this API are not the standard file mode permission values, such as 100644, but rather extrapolated values. See the supported return values.

Type: String

Valid Values: `EXECUTABLE | NORMAL | SYMLINK`

filePath

The fully qualified path to the specified file. Returns the name and extension of the file.

Type: String

fileSize

The size of the contents of the file, in bytes.

Type: Long

Errors

For information about the errors that are common to all actions, see Common Errors.

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

FileDoesNotExistException

The specified file does not exist. Verify that you have used the correct file name, full path, and extension.

HTTP Status Code: 400

FileTooLargeException

The specified file exceeds the file size limit for AWS CodeCommit. For more information about limits in AWS CodeCommit, see Quotas in the *AWS CodeCommit User Guide*.

HTTP Status Code: 400

InvalidCommitException

The specified commit is not valid.

HTTP Status Code: 400

InvalidPathException

The specified path is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

PathRequiredException

The folderPath for a location cannot be null.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetFile.

Sample Request

```
POST / HTTP/1.1
    Host: codecommit.us-east-1.amazonaws.com
    Accept-Encoding: identity
    Content-Length: 124
    X-Amz-Target: CodeCommit_20150413.GetFile
    X-Amz-Date: 20180914T223055Z
    User-Agent: aws-cli/1.15.9 Python/2.7.11 Darwin/16.7.0 botocore/1.10.9
    Content-Type: application/x-amz-json-1.1
    Authorization: AWS4-HMAC-SHA256 Credential=AKIAEXAMPLE/20180914/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
 Signature==8d9b5998EXAMPLE
{
   "commitSpecifier": "main",
   "repositoryName": "MyDemoRepo",
   "filePath": "README.md"
}
```

Sample Response

```
HTTP/1.1 200 OK
    x-amzn-RequestId: d8ad1d21-EXAMPLE
    Content-Type: application/x-amz-json-1.1
    Content-Length: 2267
    Date: Fri, 14 Sep 2018 22:30:56 GMT
    Connection: keep-alive

{
    "blobId":"559b44fEXAMPLE",
    "commitId":"c5709475EXAMPLE",
    "fileContent":"IyBQaHVzEXAMPLE",
    "filePath":"README.md",
    "fileSize":1563
    "fileMode":"NORMAL"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2

- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetFolder

Returns the contents of a specified folder in a repository.

Request Syntax

```
{
   "commitSpecifier": "string",
   "folderPath": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

commitSpecifier

A fully qualified reference used to identify a commit that contains the version of the folder's content to return. A fully qualified reference can be a commit ID, branch name, tag, or reference such as HEAD. If no specifier is provided, the folder content is returned as it exists in the HEAD commit.

Type: String

Required: No

folderPath

The fully qualified path to the folder whose contents are returned, including the folder name. For example, /examples is a fully-qualified path to a folder named examples that was created off of the root directory (/) of a repository.

Type: String

Required: Yes

repositoryName

The name of the repository.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "commitId": "string",
   "files": [
      {
         "absolutePath": "string",
         "blobId": "string",
         "fileMode": "string",
         "relativePath": "string"
      }
   ],
   "folderPath": "string",
   "subFolders": [
      {
         "absolutePath": "string",
         "relativePath": "string",
         "treeId": "string"
      }
   ],
   "subModules": [
      {
         "absolutePath": "string",
         "commitId": "string",
         "relativePath": "string"
      }
   ],
   "symbolicLinks": [
      {
         "absolutePath": "string",
         "blobId": "string",
         "fileMode": "string",
         "relativePath": "string"
      }
   ],
    "treeId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

commitId

The full commit ID used as a reference for the returned version of the folder content.

Type: String

files

The list of files in the specified folder, if any.

Type: Array of File objects

folderPath

The fully qualified path of the folder whose contents are returned.

Type: String

subFolders

The list of folders that exist under the specified folder, if any.

Type: Array of Folder objects

subModules

The list of submodules in the specified folder, if any.

Type: Array of SubModule objects

symbolicLinks

The list of symbolic links to other files and folders in the specified folder, if any.

Type: Array of SymbolicLink objects

treeId

The full SHA-1 pointer of the tree information for the commit that contains the folder.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

FolderDoesNotExistException

The specified folder does not exist. Either the folder name is not correct, or you did not enter the full path to the folder.

HTTP Status Code: 400

InvalidCommitException

The specified commit is not valid.

HTTP Status Code: 400

InvalidPathException

The specified path is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

PathRequiredException

The folderPath for a location cannot be null.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetFolder.

Sample Request

```
POST / HTTP/1.1
    Host: codecommit.us-east-1.amazonaws.com
    Accept-Encoding: identity
    Content-Length: 88
    X-Amz-Target: CodeCommit_20150413.GetFolder
    X-Amz-Date: 20180914T222143Z
    User-Agent: aws-cli/1.15.9 Python/2.7.11 Darwin/16.7.0 botocore/1.10.9
    Content-Type: application/x-amz-json-1.1
    Authorization: AWS4-HMAC-SHA256 Credential=AKIAEXAMPLE/20180914/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
 Signature=0bcf747dEXAMPLE

{
    "folderPath": "",
    "repositoryName": "MyDemoRepo"
}
```

Sample Response

```
HTTP/1.1 200 OK
    x-amzn-RequestId: 8f9d4f13-EXAMPLE
    Content-Type: application/x-amz-json-1.1
    Content-Length: 1157
    Date: Fri, 14 Sep 2018 22:21:44 GMT
    Connection: keep-alive

{
    "commitId":"c5709475EXAMPLE",
    "files":[
        {
          "absolutePath":".gitignore",
          "blobId":"74094e8bEXAMPLE",
          "fileMode":"NORMAL",
          "relativePath":".gitignore"
        },
        {
          "absolutePath":"Gemfile",
          "blobId":"9ceb72f6EXAMPLE",
          "fileMode":"NORMAL",
          "relativePath":"Gemfile"
        },
        {
          "absolutePath":"Gemfile.lock",
          "blobId":"795c4a2aEXAMPLE",
          "fileMode":"NORMAL",
          "relativePath":"Gemfile.lock"
        },
        {
          "absolutePath":"LICENSE.txt",
          "blobId":"0c7932c8EXAMPLE",
          "fileMode":"NORMAL",
          "relativePath":"LICENSE.txt"
        },
        {
          "absolutePath":"README.md",
          "blobId":"559b44feEXAMPLE",
          "fileMode":"NORMAL",
          "relativePath":"README.md"
        }
    ],
    "folderPath":"",
    "subFolders":[
        {
          "absolutePath":"public",
          "relativePath":"public",
          "treeId":"d5e92ae3aEXAMPLE"
        },
        {
          "absolutePath":"tmp",
          "relativePath":"tmp",
          "treeId":"d564d0bcEXAMPLE"
        }
    ],
    "subModules":[],
    "symbolicLinks":[],
    "treeId":"7b3c4dadEXAMPLE"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetMergeCommit

Returns information about a specified merge commit.

Request Syntax

```
{
   "conflictDetailLevel": "string",
   "conflictResolutionStrategy": "string",
   "destinationCommitSpecifier": "string",
   "repositoryName": "string",
   "sourceCommitSpecifier": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

conflictDetailLevel

The level of conflict detail to use. If unspecified, the default FILE_LEVEL is used, which returns a not-mergeable result if the same file has differences in both branches. If LINE_LEVEL is specified, a conflict is considered not mergeable if the same file in both branches has differences on the same line.

Type: String

Valid Values: `FILE_LEVEL | LINE_LEVEL`

Required: No

conflictResolutionStrategy

Specifies which branch to use when resolving conflicts, or whether to attempt automatically merging two versions of a file. The default is NONE, which requires any conflicts to be resolved manually before the merge operation is successful.

Type: String

Valid Values: `NONE | ACCEPT_SOURCE | ACCEPT_DESTINATION | AUTOMERGE`

Required: No

destinationCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

repositoryName

The name of the repository that contains the merge commit about which you want to get information.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

sourceCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

Response Syntax

```
{
   "baseCommitId": "string",
   "destinationCommitId": "string",
   "mergedCommitId": "string",
   "sourceCommitId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

baseCommitId

The commit ID of the merge base.

Type: String

destinationCommitId

The commit ID of the destination commit specifier that was used in the merge evaluation.

Type: String

mergedCommitId

The commit ID for the merge commit created when the source branch was merged into the destination branch. If the fast-forward merge strategy was used, there is no merge commit.

Type: String

sourceCommitId

The commit ID of the source commit specifier that was used in the merge evaluation.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

CommitRequiredException

A commit was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidCommitException

The specified commit is not valid.

HTTP Status Code: 400

InvalidConflictDetailLevelException

The specified conflict detail level is not valid.

HTTP Status Code: 400

InvalidConflictResolutionStrategyException

The specified conflict resolution strategy is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetMergeCommit.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 165
X-Amz-Target: CodeCommit_20150413.GetMergeCommit
X-Amz-Date: 20190428T213222Z
User-Agent: aws-cli/1.16.137 Python/3.6.0 Windows/10
Content-Type: application/x-amz-json-1.1
```

```
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "sourceCommitSpecifier": "bugfix-bug1234",
  "destinationCommitSpecifier": "main",
  "mergeOption": "THREE_WAY_MERGE"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 220
Date: Sun, 28 Apr 2019 22:43:13 GMT

{
    "sourceCommitId": "c5709475EXAMPLE",
    "destinationCommitId": "317f8570EXAMPLE",
    "baseCommitId": "fb12a539EXAMPLE",
    "mergeCommitId": "ffc4d608eEXAMPLE"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3

- AWS SDK for Python
- AWS SDK for Ruby V3

GetMergeConflicts

Returns information about merge conflicts between the before and after commit IDs for a pull request in a repository.

Request Syntax

```
{
   "conflictDetailLevel": "string",
   "conflictResolutionStrategy": "string",
   "destinationCommitSpecifier": "string",
   "maxConflictFiles": number,
   "mergeOption": "string",
   "nextToken": "string",
   "repositoryName": "string",
   "sourceCommitSpecifier": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

conflictDetailLevel

The level of conflict detail to use. If unspecified, the default FILE_LEVEL is used, which returns a not-mergeable result if the same file has differences in both branches. If LINE_LEVEL is specified, a conflict is considered not mergeable if the same file in both branches has differences on the same line.

Type: String

Valid Values: `FILE_LEVEL | LINE_LEVEL`

Required: No

conflictResolutionStrategy

Specifies which branch to use when resolving conflicts, or whether to attempt automatically merging two versions of a file. The default is NONE, which requires any conflicts to be resolved manually before the merge operation is successful.

Type: String

Valid Values: `NONE | ACCEPT_SOURCE | ACCEPT_DESTINATION | AUTOMERGE`

Required: No

destinationCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

maxConflictFiles

The maximum number of files to include in the output.

Type: Integer

Required: No

mergeOption

The merge option or strategy you want to use to merge the code.

Type: String

Valid Values: `FAST_FORWARD_MERGE | SQUASH_MERGE | THREE_WAY_MERGE`

Required: Yes

nextToken

An enumeration token that, when provided in a request, returns the next batch of the results.

Type: String

Required: No

repositoryName

The name of the repository where the pull request was created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

sourceCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

Response Syntax

```
{
   "baseCommitId": "string",
   "conflictMetadataList": [
      {
         "contentConflict": boolean,
         "fileModeConflict": boolean,
         "fileModes": {
            "base": "string",
            "destination": "string",
            "source": "string"
         },
         "filePath": "string",
         "fileSizes": {
            "base": number,
            "destination": number,
            "source": number
         },
         "isBinaryFile": {
            "base": boolean,
            "destination": boolean,
            "source": boolean
         },
         "mergeOperations": {
            "destination": "string",
            "source": "string"
         },
         "numberOfConflicts": number,
         "objectTypeConflict": boolean,
         "objectTypes": {
            "base": "string",
            "destination": "string",
            "source": "string"
         }
      }
   ],
   "destinationCommitId": "string",
   "mergeable": boolean,
   "nextToken": "string",
   "sourceCommitId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

baseCommitId

The commit ID of the merge base.

Type: String

conflictMetadataList

A list of metadata for any conflicting files. If the specified merge strategy is FAST_FORWARD_MERGE, this list is always empty.

Type: Array of ConflictMetadata objects

destinationCommitId

The commit ID of the destination commit specifier that was used in the merge evaluation.

Type: String

mergeable

A Boolean value that indicates whether the code is mergeable by the specified merge option.

Type: Boolean

nextToken

An enumeration token that can be used in a request to return the next batch of the results.

Type: String

sourceCommitId

The commit ID of the source commit specifier that was used in the merge evaluation.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

CommitRequiredException

A commit was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidCommitException

The specified commit is not valid.

HTTP Status Code: 400

InvalidConflictDetailLevelException

The specified conflict detail level is not valid.

HTTP Status Code: 400

InvalidConflictResolutionStrategyException

The specified conflict resolution strategy is not valid.

HTTP Status Code: 400

InvalidContinuationTokenException

The specified continuation token is not valid.

HTTP Status Code: 400

InvalidDestinationCommitSpecifierException

The destination commit specifier is not valid. You must provide a valid branch name, tag, or full commit ID.

HTTP Status Code: 400

InvalidMaxConflictFilesException

The specified value for the number of conflict files to return is not valid.

HTTP Status Code: 400

InvalidMergeOptionException

The specified merge option is not valid for this operation. Not all merge strategies are supported for all operations.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

InvalidSourceCommitSpecifierException

The source commit specifier is not valid. You must provide a valid branch name, tag, or full commit ID.

HTTP Status Code: 400

MaximumFileContentToLoadExceededException

The number of files to load exceeds the allowed limit.

HTTP Status Code: 400

MaximumItemsToCompareExceededException

The number of items to compare between the source or destination branches and the merge base has exceeded the maximum allowed.

HTTP Status Code: 400

MergeOptionRequiredException

A merge option or stategy is required, and none was provided.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

TipsDivergenceExceededException

The divergence between the tips of the provided commit specifiers is too great to determine whether there might be any merge conflicts. Locally compare the specifiers using `git diff` or a diff tool.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetMergeConflicts.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 57
X-Amz-Target: CodeCommit_20150413.GetMergeConflicts
X-Amz-Date: 20190603T213222Z
User-Agent: aws-cli/1.16.137 Python/3.6.0 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "sourceCommitSpecifier": "my-feature-branch",
  "destinationCommitSpecifier": "main",
  "mergeOption": THREE_WAY_MERGE
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 167
Date: Mon, June 3 2019 22:59:13 GMT

{
    "mergeable": false,
    "destinationCommitId": "86958e0aEXAMPLE",
    "sourceCommitId": "6ccd57fdEXAMPLE",
    "baseCommitId": "767b6958EXAMPLE",
    "conflictMetadataList": [
        {
            "filePath": "readme.md",
            "fileSizes": {
                "source": 139,
                "destination": 230,
                "base": 85
            },
            "fileModes": {
                "source": "NORMAL",
                "destination": "NORMAL",
                "base": "NORMAL"
            },
            "objectTypes": {
                "source": "FILE",
                "destination": "FILE",
                "base": "FILE"
            },
            "numberOfConflicts": 1,
            "isBinaryFile": {
                "source": false,
                "destination": false,
                "base": false
            },
            "contentConflict": true,
            "fileModeConflict": false,
            "objectTypeConflict": false,
            "mergeOperations": {
                "source": "M",
                "destination": "M"
            }
        }
    ]
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetMergeOptions

Returns information about the merge options available for merging two specified branches. For details about why a merge option is not available, use GetMergeConflicts or DescribeMergeConflicts.

Request Syntax

```
{
   "conflictDetailLevel": "string",
   "conflictResolutionStrategy": "string",
   "destinationCommitSpecifier": "string",
   "repositoryName": "string",
   "sourceCommitSpecifier": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

conflictDetailLevel

The level of conflict detail to use. If unspecified, the default FILE_LEVEL is used, which returns a not-mergeable result if the same file has differences in both branches. If LINE_LEVEL is specified, a conflict is considered not mergeable if the same file in both branches has differences on the same line.

Type: String

Valid Values: `FILE_LEVEL | LINE_LEVEL`

Required: No

conflictResolutionStrategy

Specifies which branch to use when resolving conflicts, or whether to attempt automatically merging two versions of a file. The default is NONE, which requires any conflicts to be resolved manually before the merge operation is successful.

Type: String

Valid Values: NONE | ACCEPT_SOURCE | ACCEPT_DESTINATION | AUTOMERGE

Required: No

destinationCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

repositoryName

The name of the repository that contains the commits about which you want to get merge options.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: [\w\.-]+

Required: Yes

sourceCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

Response Syntax

```
{
   "baseCommitId": "string",
   "destinationCommitId": "string",
   "mergeOptions": [ "string" ],
   "sourceCommitId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

baseCommitId

The commit ID of the merge base.

Type: String

destinationCommitId

The commit ID of the destination commit specifier that was used in the merge evaluation.

Type: String

mergeOptions

The merge option or strategy used to merge the code.

Type: Array of strings

Valid Values: `FAST_FORWARD_MERGE | SQUASH_MERGE | THREE_WAY_MERGE`

sourceCommitId

The commit ID of the source commit specifier that was used in the merge evaluation.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

CommitRequiredException

A commit was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidCommitException

The specified commit is not valid.

HTTP Status Code: 400

InvalidConflictDetailLevelException

The specified conflict detail level is not valid.

HTTP Status Code: 400

InvalidConflictResolutionStrategyException

The specified conflict resolution strategy is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

MaximumFileContentToLoadExceededException

The number of files to load exceeds the allowed limit.

HTTP Status Code: 400

MaximumItemsToCompareExceededException

The number of items to compare between the source or destination branches and the merge base has exceeded the maximum allowed.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

TipsDivergenceExceededException

The divergence between the tips of the provided commit specifiers is too great to determine whether there might be any merge conflicts. Locally compare the specifiers using `git diff` or a diff tool.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetMergeOptions.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 57
X-Amz-Target: CodeCommit_20150413.GetMergeOptions
X-Amz-Date: 20190428T213222Z
User-Agent: aws-cli/1.16.137 Python/3.6.0 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "sourceCommitSpecifier": "bugfix-bug1234",
  "destinationCommitSpecifier": "main"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 167
Date: Sat, 11 Nov 2017 22:59:13 GMT
```

```
{
    "mergeOptions": [
        "FAST_FORWARD_MERGE",
        "SQUASH_MERGE",
        "THREE_WAY_MERGE"
    ],
    "sourceCommitId": "18059494EXAMPLE",
    "destinationCommitId": "ffd3311dEXAMPLE",
    "baseCommitId": "ffd3311dEXAMPLE"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetPullRequest

Gets information about a pull request in a specified repository.

Request Syntax

```
{
   "pullRequestId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

pullRequestId

The system-generated ID of the pull request. To get this ID, use ListPullRequests.

Type: String

Required: Yes

Response Syntax

```
{
   "pullRequest": {
      "approvalRules": [
         {
            "approvalRuleContent": "string",
            "approvalRuleId": "string",
            "approvalRuleName": "string",
            "creationDate": number,
            "lastModifiedDate": number,
            "lastModifiedUser": "string",
            "originApprovalRuleTemplate": {
               "approvalRuleTemplateId": "string",
               "approvalRuleTemplateName": "string"
            },
            "ruleContentSha256": "string"
         }
      ],
      "authorArn": "string",
      "clientRequestToken": "string",
      "creationDate": number,
      "description": "string",
      "lastActivityDate": number,
      "pullRequestId": "string",
      "pullRequestStatus": "string",
      "pullRequestTargets": [
         {
            "destinationCommit": "string",
            "destinationReference": "string",
            "mergeBase": "string",
            "mergeMetadata": {
               "isMerged": boolean,
               "mergeCommitId": "string",
               "mergedBy": "string",
               "mergeOption": "string"
            },
            "repositoryName": "string",
            "sourceCommit": "string",
            "sourceReference": "string"
         }
      ],
      "revisionId": "string",
      "title": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

pullRequest

Information about the specified pull request.

Type: PullRequest object

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetPullRequest.

Sample Request

```
>>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.GetPullRequest
X-Amz-Date: 20191021T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "pullRequestId": "27"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Mon, 21 Oct 2019 20:20:13 GMT

{
    "pullRequest": {
        "approvalRules": [
            {
                "approvalRuleContent": "{\"Version\": \"2018-11-08\",\"Statements\":
 [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
                "approvalRuleId": "dd8b17fe-EXAMPLE",
                "approvalRuleName": "2-approver-rule-for-main",
                "creationDate": 1571356106.936,
                "lastModifiedDate": 571356106.936,
                "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
                "ruleContentSha256": "4711b576EXAMPLE"
            }
        ],
        "lastActivityDate": 1562619583.565,
        "pullRequestTargets": [
            {
                "sourceCommit": "ca45e279EXAMPLE",
                "sourceReference": "refs/heads/bugfix-1234",
                "mergeBase": "a99f5ddbEXAMPLE",
                "destinationReference": "refs/heads/main",
                "mergeMetadata": {
                    "isMerged": false
                },
                "destinationCommit": "2abfc6beEXAMPLE",
                "repositoryName": "MyDemoRepo"
            }
        ],
        "revisionId": "e47def21EXAMPLE",
        "title": "Quick fix for bug 1234",
        "authorArn": "arn:aws:iam::123456789012:user/Nikhil_Jayashankar",
        "clientRequestToken": "d8d7612e-EXAMPLE",
        "creationDate": 1562619583.565,
        "pullRequestId": "27",
        "pullRequestStatus": "OPEN"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4

- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetPullRequestApprovalStates

Gets information about the approval states for a specified pull request. Approval states only apply to pull requests that have one or more approval rules applied to them.

Request Syntax

```
{
   "pullRequestId": "string",
   "revisionId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

pullRequestId

The system-generated ID for the pull request.

Type: String

Required: Yes

revisionId

The system-generated ID for the pull request revision.

Type: String

Required: Yes

Response Syntax

```
{
   "approvals": [
      {
         "approvalState": "string",
         "userArn": "string"
      }
```

```
    ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

approvals

Information about users who have approved the pull request.

Type: Array of Approval objects

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

InvalidRevisionIdException

The revision ID is not valid. Use GetPullRequest to determine the value.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

RevisionIdRequiredException

A revision ID is required, but was not provided.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetPullRequestApprovalStates.

Sample Request

```
>>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
```

```
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.GetPullRequestApprovalStates
X-Amz-Date: 20191021T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "pullRequestId": "8",
   "revisionId": "9f29d1673EXAMPLE"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Mon, 21 Oct 2019 20:20:13 GMT

{
    "approvals": [
        {
            "userArn": "arn:aws:iam::123456789012:user/Mary_Major",
            "approvalState": "APPROVE"
        }
    ]
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2

- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetPullRequestOverrideState

Returns information about whether approval rules have been set aside (overridden) for a pull request, and if so, the Amazon Resource Name (ARN) of the user or identity that overrode the rules and their requirements for the pull request.

Request Syntax

```
{
   "pullRequestId": "string",
   "revisionId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

pullRequestId

The ID of the pull request for which you want to get information about whether approval rules have been set aside (overridden).

Type: String

Required: Yes

revisionId

The system-generated ID of the revision for the pull request. To retrieve the most recent revision ID, use GetPullRequest.

Type: String

Required: Yes

Response Syntax

```
{
    "overridden": boolean,
    "overrider": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

overridden

A Boolean value that indicates whether a pull request has had its rules set aside (TRUE) or whether all approval rules still apply (FALSE).

Type: Boolean

overrider

The Amazon Resource Name (ARN) of the user or identity that overrode the rules and their requirements for the pull request.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

InvalidRevisionIdException

The revision ID is not valid. Use GetPullRequest to determine the value.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

RevisionIdRequiredException

A revision ID is required, but was not provided.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetPullRequestOverrideState.

Sample Request

```
>>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.GetPullRequestOverrideState
X-Amz-Date: 20191021T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "pullRequestId": "34",
   "revisionId": "9f29d1673EXAMPLE"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Mon, 21 Oct 2019 20:20:13 GMT

{
    "overridden": true,
    "overrider": "arn:aws:iam::123456789012:user/Mary_Major"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetRepository

Returns information about a repository.

> **Note**
>
> The description field for a repository accepts all HTML characters and all valid Unicode characters. Applications that do not HTML-encode the description and display it in a webpage can expose users to potentially malicious code. Make sure that you HTML-encode the description field in any application that uses this API to display the repository description on a webpage.

Request Syntax

```
{
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

repositoryName

The name of the repository to get information about.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "repositoryMetadata": { 
      "accountId": "string",
      "Arn": "string",
      "cloneUrlHttp": "string",
      "cloneUrlSsh": "string",
      "creationDate": number,
      "defaultBranch": "string",
      "kmsKeyId": "string",
      "lastModifiedDate": number,
      "repositoryDescription": "string",
      "repositoryId": "string",
      "repositoryName": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

repositoryMetadata

Information about the repository.

Type: RepositoryMetadata object

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetRepository.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 33
X-Amz-Target: CodeCommit_20150413.GetRepository
X-Amz-Date: 20151028T225711Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 516
Date: Wed, 28 Oct 2015 22:57:13 GMT

{
  "repositoryMetadata": {
        "creationDate": 1429203623.625,
        "defaultBranch": "main",
        "repositoryName": "MyDemoRepo",
        "cloneUrlSsh": "ssh://git-codecommit.us-east-1.amazonaws.com/v1/repos//v1/
repos/MyDemoRepo",
        "lastModifiedDate": 1430783812.0869999,
        "repositoryDescription": "My demonstration repository",
        "cloneUrlHttp": "https://codecommit.us-east-1.amazonaws.com/v1/repos/
MyDemoRepo",
        "defaultBranch": "main",
        "kmsKeyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
        "Arn": "arn:aws:codecommit:us-east-1:123456789012EXAMPLE:MyDemoRepo
        "accountId": "123456789012"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetRepositoryTriggers

Gets information about triggers configured for a repository.

Request Syntax

```
{
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

repositoryName

The name of the repository for which the trigger is configured.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "configurationId": "string",
   "triggers": [
      {
         "branches": [ "string" ],
         "customData": "string",
         "destinationArn": "string",
         "events": [ "string" ],
         "name": "string"
      }
   ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

configurationId

The system-generated unique ID for the trigger.

Type: String

triggers

The JSON block of configuration information for each trigger.

Type: Array of RepositoryTrigger objects

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of GetRepositoryTriggers.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 33
```

```
X-Amz-Target: CodeCommit_20150413.GetRepositoryTriggers
X-Amz-Date: 20151028T230050Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 248
Date: Wed, 28 Oct 2015 23:00:52 GMT

{
  "configurationId":"e1599578-3a14-EXAMPLE",
  "triggers":[
    {
   "branches":[
     "main",
     "MyNewBranch"
  ],
   "customData":"",
   "destinationArn":"arn:aws:sns:us-east-1:123456789012EXAMPLE:MyCodeCommitTopic",
   "events":[
     "all"
     ],
   "name":"MyFirstTrigger"
 }
  ]
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListApprovalRuleTemplates

Lists all approval rule templates in the specified AWS Region in your AWS account. If an AWS Region is not specified, the AWS Region where you are signed in is used.

Request Syntax

```
{
   "maxResults": number,
   "nextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

maxResults

A non-zero, non-negative integer used to limit the number of returned results.

Type: Integer

Required: No

nextToken

An enumeration token that, when provided in a request, returns the next batch of the results.

Type: String

Required: No

Response Syntax

```
{
   "approvalRuleTemplateNames": [ "string" ],
   "nextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

approvalRuleTemplateNames

The names of all the approval rule templates found in the AWS Region for your AWS account.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 100.

nextToken

An enumeration token that allows the operation to batch the next results of the operation.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

InvalidContinuationTokenException

The specified continuation token is not valid.

HTTP Status Code: 400

InvalidMaxResultsException

The specified number of maximum results is not valid.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of ListApprovalRuleTemplates.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 33
X-Amz-Target: CodeCommit_20150413.ListApprovalRuleTemplates
X-Amz-Date: 20191021T230050Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 248
Date: Mon, 21 Oct 2019 23:00:52 GMT

{
    "approvalRuleTemplateNames": [
        "2-approver-rule-for-main",
        "1-approver-rule-for-all-pull-requests"
    ]
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2

- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListAssociatedApprovalRuleTemplatesForRepository

Lists all approval rule templates that are associated with a specified repository.

Request Syntax

```
{
   "maxResults": number,
   "nextToken": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

maxResults

A non-zero, non-negative integer used to limit the number of returned results.

Type: Integer

Required: No

nextToken

An enumeration token that, when provided in a request, returns the next batch of the results.

Type: String

Required: No

repositoryName

The name of the repository for which you want to list all associated approval rule templates.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "approvalRuleTemplateNames": [ "string" ],
   "nextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

approvalRuleTemplateNames

The names of all approval rule templates associated with the repository.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 100.

nextToken

An enumeration token that allows the operation to batch the next results of the operation.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidContinuationTokenException

The specified continuation token is not valid.

HTTP Status Code: 400

InvalidMaxResultsException

The specified number of maximum results is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of ListAssociatedApprovalRuleTemplatesForRepository.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 33
X-Amz-Target: CodeCommit_20150413.ListAssociatedApprovalRuleTemplatesForRepository
X-Amz-Date: 20191021T231012Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 55
Date: Mon, 21 Oct 2019 23:10:15 GMT

{
    "approvalRuleTemplateNames": [
        "2-approver-rule-for-main",
        "1-approver-rule-for-all-pull-requests"
    ]
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListBranches

Gets information about one or more branches in a repository.

Request Syntax

```
{
   "nextToken": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

nextToken

An enumeration token that allows the operation to batch the results.

Type: String

Required: No

repositoryName

The name of the repository that contains the branches.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "branches": [ "string" ],
   "nextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

branches

The list of branch names.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 256.

nextToken

An enumeration token that returns the batch of the results.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidContinuationTokenException

The specified continuation token is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of ListBranches.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 33
X-Amz-Target: CodeCommit_20150413.ListBranches
X-Amz-Date: 20151028T231012Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 55
Date: Wed, 28 Oct 2015 23:10:15 GMT

{
  "branches":[
    "main",
 "MyNewBranch"
 ]
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++

- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListFileCommitHistory

Retrieves a list of commits and changes to a specified file.

Request Syntax

```
{
   "commitSpecifier": "string",
   "filePath": "string",
   "maxResults": number,
   "nextToken": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

commitSpecifier

The fully qualified reference that identifies the commit that contains the file. For example, you can specify a full commit ID, a tag, a branch name, or a reference such as `refs/heads/main`. If none is provided, the head commit is used.

Type: String

Required: No

filePath

The full path of the file whose history you want to retrieve, including the name of the file.

Type: String

Required: Yes

maxResults

A non-zero, non-negative integer used to limit the number of returned results.

Type: Integer

Required: No

nextToken

An enumeration token that allows the operation to batch the results.

Type: String

Required: No

repositoryName

The name of the repository that contains the file.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "nextToken": "string",
   "revisionDag": [
      {
         "blobId": "string",
         "commit": {
            "additionalData": "string",
            "author": {
               "date": "string",
               "email": "string",
               "name": "string"
            },
            "commitId": "string",
            "committer": {
               "date": "string",
               "email": "string",
               "name": "string"
            },
            "message": "string",
            "parents": [ "string" ],
            "treeId": "string"
         },
         "path": "string",
         "revisionChildren": [ "string" ]
      }
   ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

nextToken

An enumeration token that can be used to return the next batch of results.

Type: String

revisionDag

An array of FileVersion objects that form a directed acyclic graph (DAG) of the changes to the file made by the commits that changed the file.

Type: Array of FileVersion objects

Errors

For information about the errors that are common to all actions, see Common Errors.

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

CommitRequiredException

A commit was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidCommitException

The specified commit is not valid.

HTTP Status Code: 400

InvalidContinuationTokenException

The specified continuation token is not valid.

HTTP Status Code: 400

InvalidMaxResultsException

The specified number of maximum results is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

TipsDivergenceExceededException

The divergence between the tips of the provided commit specifiers is too great to determine whether there might be any merge conflicts. Locally compare the specifiers using `git diff` or a diff tool.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of ListFileCommitHistory.

Sample Request

```
POST / HTTP/1.1
```

```
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 54
X-Amz-Target: CodeCommit_20150413.ListFileCommitHistory
X-Amz-Date: 20230816T224020Z
User-Agent: aws-cli/1.29.30 Python/3.6.10 Linux/4.9.184-0.1.ac.235.83.329.metal1.x86_64
 botocore/1.15.36
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "filePath": "README.md"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 300
Date: Wed, 16 Aug 2023 22:40:20 GMT

{
  "revisionDag": [
    { "commit": {…, "commitId": "4f178133EXAMPLE", …}, "blobId": "2eb4af3bEXAMPLE",
 "filePath": "README.md", "revisionChildren": [] },
    { "commit": {…, "commitId": "317f8570EXAMPLE", …}, "blobId": "bf7fcf28fEXAMPLE",
 "filePath": "README.md", "revisionChildren": ["4f178133EXAMPLE"] },
  ],
  "nextToken": "exampleToken",
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4

- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListPullRequests

Returns a list of pull requests for a specified repository. The return list can be refined by pull request status or pull request author ARN.

Request Syntax

```
{
   "authorArn": "string",
   "maxResults": number,
   "nextToken": "string",
   "pullRequestStatus": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

authorArn

Optional. The Amazon Resource Name (ARN) of the user who created the pull request. If used, this filters the results to pull requests created by that user.

Type: String

Required: No

maxResults

A non-zero, non-negative integer used to limit the number of returned results.

Type: Integer

Required: No

nextToken

An enumeration token that, when provided in a request, returns the next batch of the results.

Type: String

Required: No

pullRequestStatus

Optional. The status of the pull request. If used, this refines the results to the pull requests that match the specified status.

Type: String

Valid Values: `OPEN | CLOSED`

Required: No

repositoryName

The name of the repository for which you want to list pull requests.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "nextToken": "string",
   "pullRequestIds": [ "string" ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

nextToken

An enumeration token that allows the operation to batch the next results of the operation.

Type: String

pullRequestIds

The system-generated IDs of the pull requests.

Type: Array of strings

Errors

For information about the errors that are common to all actions, see Common Errors.

AuthorDoesNotExistException

The specified Amazon Resource Name (ARN) does not exist in the AWS account.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidAuthorArnException

The Amazon Resource Name (ARN) is not valid. Make sure that you have provided the full ARN for the author of the pull request, and then try again.

HTTP Status Code: 400

InvalidContinuationTokenException

The specified continuation token is not valid.

HTTP Status Code: 400

InvalidMaxResultsException

The specified number of maximum results is not valid.

HTTP Status Code: 400

InvalidPullRequestStatusException

The pull request status is not valid. The only valid values are `OPEN` and `CLOSED`.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of ListPullRequests.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 174
X-Amz-Target: CodeCommit_20150413.ListPullRequests
X-Amz-Date: 20171025T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
   "maxResults": 20,
   "nextToken": "",
   "pullRequestStatus": "CLOSED",
   "repositoryName": "MyDemoRepo"
}
```

Sample Response

```



{
   "nextToken": "exampleToken",
   "pullRequestIds": ["2","12","16","22","30","23","35","39","47"]
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListRepositories

Gets information about one or more repositories.

Request Syntax

```
{
   "nextToken": "string",
   "order": "string",
   "sortBy": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

nextToken

An enumeration token that allows the operation to batch the results of the operation. Batch sizes are 1,000 for list repository operations. When the client sends the token back to AWS CodeCommit, another page of 1,000 records is retrieved.

Type: String

Required: No

order

The order in which to sort the results of a list repositories operation.

Type: String

Valid Values: `ascending | descending`

Required: No

sortBy

The criteria used to sort the results of a list repositories operation.

Type: String

Valid Values: `repositoryName | lastModifiedDate`

Required: No

Response Syntax

```
{
   "nextToken": "string",
   "repositories": [
      {
         "repositoryId": "string",
         "repositoryName": "string"
      }
   ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

nextToken

An enumeration token that allows the operation to batch the results of the operation. Batch sizes are 1,000 for list repository operations. When the client sends the token back to AWS CodeCommit, another page of 1,000 records is retrieved.

Type: String

repositories

Lists the repositories called by the list repositories operation.

Type: Array of RepositoryNameIdPair objects

Errors

For information about the errors that are common to all actions, see Common Errors.

InvalidContinuationTokenException

The specified continuation token is not valid.

HTTP Status Code: 400

InvalidOrderException

The specified sort order is not valid.

HTTP Status Code: 400

InvalidSortByException

The specified sort by value is not valid.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of ListRepositories.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 2
X-Amz-Target: CodeCommit_20150413.ListRepositories
X-Amz-Date: 20151028T212036Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{}
```

Sample Response

```
HTTP/1.1 200 OK
```

```
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 721
Date: Wed, 28 Oct 2015 21:20:37 GMT

{
  "repositories":[
    {
   "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
      "repositoryName": "MyDemoRepo"
    },
    {
      "repositoryId": "cfc29ac4-b0cb-44dc-9990-f6f51EXAMPLE"
   "repositoryName": "MyOtherDemoRepo"
    }
  ]
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListRepositoriesForApprovalRuleTemplate

Lists all repositories associated with the specified approval rule template.

Request Syntax

```
{
   "approvalRuleTemplateName": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

approvalRuleTemplateName

The name of the approval rule template for which you want to list repositories that are associated with that template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: Yes

maxResults

A non-zero, non-negative integer used to limit the number of returned results.

Type: Integer

Required: No

nextToken

An enumeration token that, when provided in a request, returns the next batch of the results.

Type: String

Required: No

Response Syntax

```
{
   "nextToken": "string",
   "repositoryNames": [ "string" ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

nextToken

An enumeration token that allows the operation to batch the next results of the operation.

Type: String

repositoryNames

A list of repository names that are associated with the specified approval rule template.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Errors

For information about the errors that are common to all actions, see Common Errors.

ApprovalRuleTemplateDoesNotExistException

The specified approval rule template does not exist. Verify that the name is correct and that you are signed in to the AWS Region where the template was created, and then try again.

HTTP Status Code: 400

ApprovalRuleTemplateNameRequiredException

An approval rule template name is required, but was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidApprovalRuleTemplateNameException

The name of the approval rule template is not valid. Template names must be between 1 and 100 valid characters in length. For more information about limits in AWS CodeCommit, see Quotas in the *AWS CodeCommit User Guide*.

HTTP Status Code: 400

InvalidContinuationTokenException

The specified continuation token is not valid.

HTTP Status Code: 400

InvalidMaxResultsException

The specified number of maximum results is not valid.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of ListRepositoriesForApprovalRuleTemplate.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 2
X-Amz-Target: CodeCommit_20150413.ListRepositoriesForApprovalRuleTemplate
X-Amz-Date: 20191021T212036Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "approvalRuleTemplateName": "2-approver-rule-for-main"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 721
Date: Mon, 21 Oct 2019 21:20:37 GMT

{
    "repositoryNames": [
        "MyDemoRepo",
        "MyClonedRepo"
    ]
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListTagsForResource

Gets information about AWStags for a specified Amazon Resource Name (ARN) in AWS CodeCommit. For a list of valid resources in AWS CodeCommit, see CodeCommit Resources and Operations in the *AWS CodeCommit User Guide*.

Request Syntax

```
{
   "nextToken": "string",
   "resourceArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

nextToken

An enumeration token that, when provided in a request, returns the next batch of the results.

Type: String

Required: No

resourceArn

The Amazon Resource Name (ARN) of the resource for which you want to get information about tags, if any.

Type: String

Required: Yes

Response Syntax

```
{
   "nextToken": "string",
   "tags": {
      "string" : "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

nextToken

An enumeration token that allows the operation to batch the next results of the operation.

Type: String

tags

A list of tag key and value pairs associated with the specified resource.

Type: String to string map

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Errors

For information about the errors that are common to all actions, see Common Errors.

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

InvalidResourceArnException

The value for the resource ARN is not valid. For more information about resources in AWS CodeCommit, see CodeCommit Resources and Operations in the AWS CodeCommit User Guide.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

ResourceArnRequiredException

A valid Amazon Resource Name (ARN) for an AWS CodeCommit resource is required. For a list of valid resources in AWS CodeCommit, see CodeCommit Resources and Operations in the AWS CodeCommit User Guide.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of ListTagsForResource.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 2
X-Amz-Target: CodeCommit_20150413.ListTagsForResource
X-Amz-Date: 20190228T224019Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "resourceArn": "arn:aws:codecommit:us-east-2:123456789012:MySecretProjectRepo"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 81
Date: Thu, 28 Feb 2019 22:40:19 GMT

{
    "tags": {
        "Status": "Secret",
        "Team": "Saanvi"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

MergeBranchesByFastForward

Merges two branches using the fast-forward merge strategy.

Request Syntax

```
{
   "destinationCommitSpecifier": "string",
   "repositoryName": "string",
   "sourceCommitSpecifier": "string",
   "targetBranch": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

destinationCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

repositoryName

The name of the repository where you want to merge two branches.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

sourceCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

targetBranch

The branch where the merge is applied.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

Response Syntax

```
{
   "commitId": "string",
   "treeId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

commitId

The commit ID of the merge in the destination or target branch.

Type: String

treeId

The tree ID of the merge in the destination or target branch.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

BranchDoesNotExistException

The specified branch does not exist.

HTTP Status Code: 400

BranchNameIsTagNameException

The specified branch name is not valid because it is a tag name. Enter the name of a branch in the repository. For a list of valid branch names, use ListBranches.

HTTP Status Code: 400

BranchNameRequiredException

A branch name is required, but was not specified.

HTTP Status Code: 400

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

CommitRequiredException

A commit was not specified.

HTTP Status Code: 400

ConcurrentReferenceUpdateException

The merge cannot be completed because the target branch has been modified. Another user might have modified the target branch while the merge was in progress. Wait a few minutes, and then try again.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidBranchNameException

The specified reference name is not valid.

HTTP Status Code: 400

InvalidCommitException

The specified commit is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

InvalidTargetBranchException

The specified target branch is not valid.

HTTP Status Code: 400

ManualMergeRequiredException

The pull request cannot be merged automatically into the destination branch. You must manually merge the branches and resolve any conflicts.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

TipsDivergenceExceededException

The divergence between the tips of the provided commit specifiers is too great to determine whether there might be any merge conflicts. Locally compare the specifiers using `git diff` or a diff tool.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of MergeBranchesByFastForward.

Sample Request

```
>POST / HTTP/1.1
```

```
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 226
X-Amz-Target: CodeCommit_20150413.MergeBranchesByFastForward
X-Amz-Date: 20190428T213222Z
User-Agent: aws-cli/1.16.137 Python/3.6.0 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "destinationCommitSpecifier": "bugfix-bug1234",
  "sourceCommitSpecifier": "main",
  "targetBranch": "main-preprod"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 1681
Date: Sun, 28 Apr 2019 22:43:13 GMT

{
    "commitId": "4f178133EXAMPLE",
    "treeId": "389765daEXAMPLE"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2

- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

MergeBranchesBySquash

Merges two branches using the squash merge strategy.

Request Syntax

```
{
   "authorName": "string",
   "commitMessage": "string",
   "conflictDetailLevel": "string",
   "conflictResolution": {
      "deleteFiles": [
         {
            "filePath": "string"
         }
      ],
      "replaceContents": [
         {
            "content": blob,
            "fileMode": "string",
            "filePath": "string",
            "replacementType": "string"
         }
      ],
      "setFileModes": [
         {
            "fileMode": "string",
            "filePath": "string"
         }
      ]
   },
   "conflictResolutionStrategy": "string",
   "destinationCommitSpecifier": "string",
   "email": "string",
   "keepEmptyFolders": boolean,
   "repositoryName": "string",
   "sourceCommitSpecifier": "string",
   "targetBranch": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

authorName

The name of the author who created the commit. This information is used as both the author and committer for the commit.

Type: String

Required: No

commitMessage

The commit message for the merge.

Type: String

Required: No

conflictDetailLevel

The level of conflict detail to use. If unspecified, the default FILE_LEVEL is used, which returns a not-mergeable result if the same file has differences in both branches. If LINE_LEVEL is specified, a conflict is considered not mergeable if the same file in both branches has differences on the same line.

Type: String

Valid Values: `FILE_LEVEL | LINE_LEVEL`

Required: No

conflictResolution

If AUTOMERGE is the conflict resolution strategy, a list of inputs to use when resolving conflicts during a merge.

Type: ConflictResolution object

Required: No

conflictResolutionStrategy

Specifies which branch to use when resolving conflicts, or whether to attempt automatically merging two versions of a file. The default is NONE, which requires any conflicts to be resolved manually before the merge operation is successful.

Type: String

Valid Values: `NONE | ACCEPT_SOURCE | ACCEPT_DESTINATION | AUTOMERGE`

Required: No

destinationCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

email

The email address of the person merging the branches. This information is used in the commit information for the merge.

Type: String

Required: No

keepEmptyFolders

If the commit contains deletions, whether to keep a folder or folder structure if the changes leave the folders empty. If this is specified as true, a .gitkeep file is created for empty folders. The default is false.

Type: Boolean

Required: No

repositoryName

The name of the repository where you want to merge two branches.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

sourceCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

targetBranch

The branch where the merge is applied.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

Response Syntax

```
{
   "commitId": "string",
   "treeId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

commitId

The commit ID of the merge in the destination or target branch.

Type: String

treeId

The tree ID of the merge in the destination or target branch.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

BranchDoesNotExistException

The specified branch does not exist.

HTTP Status Code: 400

BranchNameIsTagNameException

The specified branch name is not valid because it is a tag name. Enter the name of a branch in the repository. For a list of valid branch names, use ListBranches.

HTTP Status Code: 400

BranchNameRequiredException

A branch name is required, but was not specified.

HTTP Status Code: 400

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

CommitMessageLengthExceededException

The commit message is too long. Provide a shorter string.

HTTP Status Code: 400

CommitRequiredException

A commit was not specified.

HTTP Status Code: 400

ConcurrentReferenceUpdateException

The merge cannot be completed because the target branch has been modified. Another user might have modified the target branch while the merge was in progress. Wait a few minutes, and then try again.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

FileContentSizeLimitExceededException

The file cannot be added because it is too large. The maximum file size is 6 MB, and the combined file content change size is 7 MB. Consider making these changes using a Git client.

HTTP Status Code: 400

FileModeRequiredException

The commit cannot be created because no file mode has been specified. A file mode is required to update mode permissions for a file.

HTTP Status Code: 400

FolderContentSizeLimitExceededException

The commit cannot be created because at least one of the overall changes in the commit results in a folder whose contents exceed the limit of 6 MB. Either reduce the number and size of your changes, or split the changes across multiple folders.

HTTP Status Code: 400

InvalidBranchNameException

The specified reference name is not valid.

HTTP Status Code: 400

InvalidCommitException

The specified commit is not valid.

HTTP Status Code: 400

InvalidConflictDetailLevelException

The specified conflict detail level is not valid.

HTTP Status Code: 400

InvalidConflictResolutionException

The specified conflict resolution list is not valid.

HTTP Status Code: 400

InvalidConflictResolutionStrategyException

The specified conflict resolution strategy is not valid.

HTTP Status Code: 400

InvalidEmailException

The specified email address either contains one or more characters that are not allowed, or it exceeds the maximum number of characters allowed for an email address.

HTTP Status Code: 400

InvalidFileModeException

The specified file mode permission is not valid. For a list of valid file mode permissions, see PutFile.

HTTP Status Code: 400

InvalidPathException

The specified path is not valid.

HTTP Status Code: 400

InvalidReplacementContentException

Automerge was specified for resolving the conflict, but the replacement type is not valid or content is missing.

HTTP Status Code: 400

InvalidReplacementTypeException

Automerge was specified for resolving the conflict, but the specified replacement type is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

InvalidTargetBranchException

The specified target branch is not valid.

HTTP Status Code: 400

ManualMergeRequiredException

The pull request cannot be merged automatically into the destination branch. You must manually merge the branches and resolve any conflicts.

HTTP Status Code: 400

MaximumConflictResolutionEntriesExceededException

The number of allowed conflict resolution entries was exceeded.

HTTP Status Code: 400

MaximumFileContentToLoadExceededException

The number of files to load exceeds the allowed limit.

HTTP Status Code: 400

MaximumItemsToCompareExceededException

The number of items to compare between the source or destination branches and the merge base has exceeded the maximum allowed.

HTTP Status Code: 400

MultipleConflictResolutionEntriesException

More than one conflict resolution entries exists for the conflict. A conflict can have only one conflict resolution entry.

HTTP Status Code: 400

NameLengthExceededException

The user name is not valid because it has exceeded the character limit for author names.

HTTP Status Code: 400

PathRequiredException

The folderPath for a location cannot be null.

HTTP Status Code: 400

ReplacementContentRequiredException

USE_NEW_CONTENT was specified, but no replacement content has been provided.

HTTP Status Code: 400

ReplacementTypeRequiredException

A replacement type is required.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

TipsDivergenceExceededException

The divergence between the tips of the provided commit specifiers is too great to determine whether there might be any merge conflicts. Locally compare the specifiers using `git diff` or a diff tool.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of MergeBranchesBySquash.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 226
X-Amz-Target: CodeCommit_20150413.MergeBranchesBySquash
```

```
X-Amz-Date: 20190428T213222Z
User-Agent: aws-cli/1.16.137 Python/3.6.0 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "destinationCommitSpecifier": "bugfix-bug1234",
  "sourceCommitSpecifier": "main",
  "authorName": "Maria Garcia",
  "email": "maria_garcia@example.com",
  "commitMessage": "Merging changes in main to the bugfix branch"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 1681
Date: Sun, 28 Apr 2019 22:43:13 GMT

{
    "commitId": "4f178133EXAMPLE",
    "treeId": "389765daEXAMPLE"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3

- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

MergeBranchesByThreeWay

Merges two specified branches using the three-way merge strategy.

Request Syntax

```
{
   "authorName": "string",
   "commitMessage": "string",
   "conflictDetailLevel": "string",
   "conflictResolution": {
      "deleteFiles": [
         {
            "filePath": "string"
         }
      ],
      "replaceContents": [
         {
            "content": blob,
            "fileMode": "string",
            "filePath": "string",
            "replacementType": "string"
         }
      ],
      "setFileModes": [
         {
            "fileMode": "string",
            "filePath": "string"
         }
      ]
   },
   "conflictResolutionStrategy": "string",
   "destinationCommitSpecifier": "string",
   "email": "string",
   "keepEmptyFolders": boolean,
   "repositoryName": "string",
   "sourceCommitSpecifier": "string",
   "targetBranch": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

authorName

The name of the author who created the commit. This information is used as both the author and committer for the commit.

Type: String

Required: No

commitMessage

The commit message to include in the commit information for the merge.

Type: String

Required: No

conflictDetailLevel

The level of conflict detail to use. If unspecified, the default FILE_LEVEL is used, which returns a not-mergeable result if the same file has differences in both branches. If LINE_LEVEL is specified, a conflict is considered not mergeable if the same file in both branches has differences on the same line.

Type: String

Valid Values: `FILE_LEVEL | LINE_LEVEL`

Required: No

conflictResolution

If AUTOMERGE is the conflict resolution strategy, a list of inputs to use when resolving conflicts during a merge.

Type: ConflictResolution object

Required: No

conflictResolutionStrategy

Specifies which branch to use when resolving conflicts, or whether to attempt automatically merging two versions of a file. The default is NONE, which requires any conflicts to be resolved manually before the merge operation is successful.

Type: String

Valid Values: `NONE | ACCEPT_SOURCE | ACCEPT_DESTINATION | AUTOMERGE`

Required: No

destinationCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

email

The email address of the person merging the branches. This information is used in the commit information for the merge.

Type: String

Required: No

keepEmptyFolders

If the commit contains deletions, whether to keep a folder or folder structure if the changes leave the folders empty. If true, a .gitkeep file is created for empty folders. The default is false.

Type: Boolean

Required: No

repositoryName

The name of the repository where you want to merge two branches.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

sourceCommitSpecifier

The branch, tag, HEAD, or other fully qualified reference used to identify a commit (for example, a branch name or a full commit ID).

Type: String

Required: Yes

targetBranch

The branch where the merge is applied.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

Response Syntax

```
{
   "commitId": "string",
   "treeId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

commitId

The commit ID of the merge in the destination or target branch.

Type: String

treeId

The tree ID of the merge in the destination or target branch.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

BranchDoesNotExistException

The specified branch does not exist.

HTTP Status Code: 400

BranchNameIsTagNameException

The specified branch name is not valid because it is a tag name. Enter the name of a branch in the repository. For a list of valid branch names, use ListBranches.

HTTP Status Code: 400

BranchNameRequiredException

A branch name is required, but was not specified.

HTTP Status Code: 400

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

CommitMessageLengthExceededException

The commit message is too long. Provide a shorter string.

HTTP Status Code: 400

CommitRequiredException

A commit was not specified.

HTTP Status Code: 400

ConcurrentReferenceUpdateException

The merge cannot be completed because the target branch has been modified. Another user might have modified the target branch while the merge was in progress. Wait a few minutes, and then try again.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

FileContentSizeLimitExceededException

The file cannot be added because it is too large. The maximum file size is 6 MB, and the combined file content change size is 7 MB. Consider making these changes using a Git client.

HTTP Status Code: 400

FileModeRequiredException

The commit cannot be created because no file mode has been specified. A file mode is required to update mode permissions for a file.

HTTP Status Code: 400

FolderContentSizeLimitExceededException

The commit cannot be created because at least one of the overall changes in the commit results in a folder whose contents exceed the limit of 6 MB. Either reduce the number and size of your changes, or split the changes across multiple folders.

HTTP Status Code: 400

InvalidBranchNameException

The specified reference name is not valid.

HTTP Status Code: 400

InvalidCommitException

The specified commit is not valid.

HTTP Status Code: 400

InvalidConflictDetailLevelException

The specified conflict detail level is not valid.

HTTP Status Code: 400

InvalidConflictResolutionException

The specified conflict resolution list is not valid.

HTTP Status Code: 400

InvalidConflictResolutionStrategyException

The specified conflict resolution strategy is not valid.

HTTP Status Code: 400

InvalidEmailException

The specified email address either contains one or more characters that are not allowed, or it exceeds the maximum number of characters allowed for an email address.

HTTP Status Code: 400

InvalidFileModeException

The specified file mode permission is not valid. For a list of valid file mode permissions, see PutFile.

HTTP Status Code: 400

InvalidPathException

The specified path is not valid.

HTTP Status Code: 400

InvalidReplacementContentException

Automerge was specified for resolving the conflict, but the replacement type is not valid or content is missing.

HTTP Status Code: 400

InvalidReplacementTypeException

Automerge was specified for resolving the conflict, but the specified replacement type is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

InvalidTargetBranchException

The specified target branch is not valid.

HTTP Status Code: 400

ManualMergeRequiredException

The pull request cannot be merged automatically into the destination branch. You must manually merge the branches and resolve any conflicts.

HTTP Status Code: 400

MaximumConflictResolutionEntriesExceededException

The number of allowed conflict resolution entries was exceeded.

HTTP Status Code: 400

MaximumFileContentToLoadExceededException

The number of files to load exceeds the allowed limit.

HTTP Status Code: 400

MaximumItemsToCompareExceededException

The number of items to compare between the source or destination branches and the merge base has exceeded the maximum allowed.

HTTP Status Code: 400

MultipleConflictResolutionEntriesException

More than one conflict resolution entries exists for the conflict. A conflict can have only one conflict resolution entry.

HTTP Status Code: 400

NameLengthExceededException

The user name is not valid because it has exceeded the character limit for author names.

HTTP Status Code: 400

PathRequiredException

The folderPath for a location cannot be null.

HTTP Status Code: 400

ReplacementContentRequiredException

USE_NEW_CONTENT was specified, but no replacement content has been provided.

HTTP Status Code: 400

ReplacementTypeRequiredException

A replacement type is required.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

TipsDivergenceExceededException

The divergence between the tips of the provided commit specifiers is too great to determine whether there might be any merge conflicts. Locally compare the specifiers using `git diff` or a diff tool.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of MergeBranchesByThreeWay.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 226
X-Amz-Target: CodeCommit_20150413.MergeBranchesByThreeWay
```

```
X-Amz-Date: 20190428T213222Z
User-Agent: aws-cli/1.16.137 Python/3.6.0 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "destinationCommitSpecifier": "bugfix-bug1234",
  "sourceCommitSpecifier": "main",
  "authorName": "Jorge Souza",
  "email": "jorge_souza@example.com",
  "commitMessage": "Merging changes from main before additional testing"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 1681
Date: Sun, 28 Apr 2019 22:43:13 GMT

{
    "commitId": "4f178133EXAMPLE",
    "treeId": "389765daEXAMPLE"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3

- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

MergePullRequestByFastForward

Attempts to merge the source commit of a pull request into the specified destination branch for that pull request at the specified commit using the fast-forward merge strategy. If the merge is successful, it closes the pull request.

Request Syntax

```
{
   "pullRequestId": "string",
   "repositoryName": "string",
   "sourceCommitId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

pullRequestId

The system-generated ID of the pull request. To get this ID, use ListPullRequests.

Type: String

Required: Yes

repositoryName

The name of the repository where the pull request was created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

sourceCommitId

The full commit ID of the original or updated commit in the pull request source branch. Pass this value if you want an exception thrown if the current commit ID of the tip of the source branch does not match this commit ID.

Type: String

Required: No

Response Syntax

```
{
   "pullRequest": {
      "approvalRules": [
         {
            "approvalRuleContent": "string",
            "approvalRuleId": "string",
            "approvalRuleName": "string",
            "creationDate": number,
            "lastModifiedDate": number,
            "lastModifiedUser": "string",
            "originApprovalRuleTemplate": {
               "approvalRuleTemplateId": "string",
               "approvalRuleTemplateName": "string"
            },
            "ruleContentSha256": "string"
         }
      ],
      "authorArn": "string",
      "clientRequestToken": "string",
      "creationDate": number,
      "description": "string",
      "lastActivityDate": number,
      "pullRequestId": "string",
      "pullRequestStatus": "string",
      "pullRequestTargets": [
         {
            "destinationCommit": "string",
            "destinationReference": "string",
            "mergeBase": "string",
            "mergeMetadata": {
               "isMerged": boolean,
               "mergeCommitId": "string",
               "mergedBy": "string",
               "mergeOption": "string"
            },
            "repositoryName": "string",
            "sourceCommit": "string",
            "sourceReference": "string"
         }
      ],
      "revisionId": "string",
      "title": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

pullRequest

Information about the specified pull request, including the merge.

Type: PullRequest object

Errors

For information about the errors that are common to all actions, see Common Errors.

ConcurrentReferenceUpdateException

The merge cannot be completed because the target branch has been modified. Another user might have modified the target branch while the merge was in progress. Wait a few minutes, and then try again.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidCommitIdException

The specified commit ID is not valid.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

ManualMergeRequiredException

The pull request cannot be merged automatically into the destination branch. You must manually merge the branches and resolve any conflicts.

HTTP Status Code: 400

PullRequestAlreadyClosedException

The pull request status cannot be updated because it is already closed.

HTTP Status Code: 400

PullRequestApprovalRulesNotSatisfiedException

The pull request cannot be merged because one or more approval rules applied to the pull request have conditions that have not been met.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

ReferenceDoesNotExistException

The specified reference does not exist. You must provide a full commit ID.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

RepositoryNotAssociatedWithPullRequestException

The repository does not contain any pull requests with that pull request ID. Use GetPullRequest to verify the correct repository name for the pull request ID.

HTTP Status Code: 400

TipOfSourceReferenceIsDifferentException

The tip of the source branch in the destination repository does not match the tip of the source branch specified in your request. The pull request might have been updated. Make sure that you have the latest changes.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of MergePullRequestByFastForward.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.MergePullRequestByFastForward
X-Amz-Date: 20171025T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "pullRequestId": "47",
   "repositoryName": "MyDemoRepo",
   "sourceCommitId": "99132ab0EXAMPLE"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Wed, 25 Oct 2017 20:20:13 GMT

{
    "pullRequest": {
        "approvalRules": [
            {
                "approvalRuleContent": "{\"Version\": \"2018-11-08\",\"Statements\":
 [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 1,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
                "approvalRuleId": "dd8b17fe-EXAMPLE",
                "approvalRuleName": "I want one approver for this pull request",
                "creationDate": 1571356106.936,
                "lastModifiedDate": 571356106.936,
                "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
                "ruleContentSha256": "4711b576EXAMPLE"
            }
        ],
        "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
        "clientRequestToken": "",
        "creationDate": 1508530823.142,
        "description": "Review the latest changes and updates to the global variables",
        "lastActivityDate": 1508887223.155,
        "pullRequestId": "47",
        "pullRequestStatus": "CLOSED",
        "pullRequestTargets": [
            {
                "destinationCommit": "9f31c968EXAMPLE",
                "destinationReference": "refs/heads/main",
                "mergeMetadata": {
                    "isMerged": true,
                    "mergedBy": "arn:aws:iam::123456789012:user/Mary_Major"
                },
                "repositoryName": "MyDemoRepo",
                "sourceCommit": "99132ab0EXAMPLE",
                "sourceReference": "refs/heads/variables-branch"
            }
        ],
        "title": "Consolidation of global variables"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

MergePullRequestBySquash

Attempts to merge the source commit of a pull request into the specified destination branch for that pull request at the specified commit using the squash merge strategy. If the merge is successful, it closes the pull request.

Request Syntax

```
{
   "authorName": "string",
   "commitMessage": "string",
   "conflictDetailLevel": "string",
   "conflictResolution": {
      "deleteFiles": [
         {
            "filePath": "string"
         }
      ],
      "replaceContents": [
         {
            "content": blob,
            "fileMode": "string",
            "filePath": "string",
            "replacementType": "string"
         }
      ],
      "setFileModes": [
         {
            "fileMode": "string",
            "filePath": "string"
         }
      ]
   },
   "conflictResolutionStrategy": "string",
   "email": "string",
   "keepEmptyFolders": boolean,
   "pullRequestId": "string",
   "repositoryName": "string",
   "sourceCommitId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

authorName

The name of the author who created the commit. This information is used as both the author and committer for the commit.

Type: String

Required: No

commitMessage

The commit message to include in the commit information for the merge.

Type: String

Required: No

conflictDetailLevel

The level of conflict detail to use. If unspecified, the default FILE_LEVEL is used, which returns a not-mergeable result if the same file has differences in both branches. If LINE_LEVEL is specified, a conflict is considered not mergeable if the same file in both branches has differences on the same line.

Type: String

Valid Values: `FILE_LEVEL | LINE_LEVEL`

Required: No

conflictResolution

If AUTOMERGE is the conflict resolution strategy, a list of inputs to use when resolving conflicts during a merge.

Type: ConflictResolution object

Required: No

conflictResolutionStrategy

Specifies which branch to use when resolving conflicts, or whether to attempt automatically merging two versions of a file. The default is NONE, which requires any conflicts to be resolved manually before the merge operation is successful.

Type: String

Valid Values: `NONE | ACCEPT_SOURCE | ACCEPT_DESTINATION | AUTOMERGE`

Required: No

email

The email address of the person merging the branches. This information is used in the commit information for the merge.

Type: String

Required: No

keepEmptyFolders

If the commit contains deletions, whether to keep a folder or folder structure if the changes leave the folders empty. If true, a .gitkeep file is created for empty folders. The default is false.

Type: Boolean

Required: No

pullRequestId

The system-generated ID of the pull request. To get this ID, use ListPullRequests.

Type: String

Required: Yes

repositoryName

The name of the repository where the pull request was created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

sourceCommitId

The full commit ID of the original or updated commit in the pull request source branch. Pass this value if you want an exception thrown if the current commit ID of the tip of the source branch does not match this commit ID.

Type: String

Required: No

Response Syntax

```
{
   "pullRequest": {
      "approvalRules": [
         {
            "approvalRuleContent": "string",
            "approvalRuleId": "string",
            "approvalRuleName": "string",
            "creationDate": number,
            "lastModifiedDate": number,
            "lastModifiedUser": "string",
            "originApprovalRuleTemplate": {
               "approvalRuleTemplateId": "string",
               "approvalRuleTemplateName": "string"
            },
            "ruleContentSha256": "string"
         }
      ],
      "authorArn": "string",
      "clientRequestToken": "string",
      "creationDate": number,
      "description": "string",
      "lastActivityDate": number,
      "pullRequestId": "string",
      "pullRequestStatus": "string",
      "pullRequestTargets": [
         {
            "destinationCommit": "string",
            "destinationReference": "string",
            "mergeBase": "string",
            "mergeMetadata": {
               "isMerged": boolean,
               "mergeCommitId": "string",
               "mergedBy": "string",
               "mergeOption": "string"
            },
            "repositoryName": "string",
            "sourceCommit": "string",
            "sourceReference": "string"
         }
      ],
      "revisionId": "string",
      "title": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

pullRequest

Returns information about a pull request.

Type: PullRequest object

Errors

For information about the errors that are common to all actions, see Common Errors.

CommitMessageLengthExceededException

The commit message is too long. Provide a shorter string.

HTTP Status Code: 400

ConcurrentReferenceUpdateException

The merge cannot be completed because the target branch has been modified. Another user might have modified the target branch while the merge was in progress. Wait a few minutes, and then try again.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

FileContentSizeLimitExceededException

The file cannot be added because it is too large. The maximum file size is 6 MB, and the combined file content change size is 7 MB. Consider making these changes using a Git client.

HTTP Status Code: 400

FolderContentSizeLimitExceededException

The commit cannot be created because at least one of the overall changes in the commit results in a folder whose contents exceed the limit of 6 MB. Either reduce the number and size of your changes, or split the changes across multiple folders.

HTTP Status Code: 400

InvalidCommitIdException

The specified commit ID is not valid.

HTTP Status Code: 400

InvalidConflictDetailLevelException

The specified conflict detail level is not valid.

HTTP Status Code: 400

InvalidConflictResolutionException

The specified conflict resolution list is not valid.

HTTP Status Code: 400

InvalidConflictResolutionStrategyException

The specified conflict resolution strategy is not valid.

HTTP Status Code: 400

InvalidEmailException

The specified email address either contains one or more characters that are not allowed, or it exceeds the maximum number of characters allowed for an email address.

HTTP Status Code: 400

InvalidFileModeException

The specified file mode permission is not valid. For a list of valid file mode permissions, see PutFile.

HTTP Status Code: 400

InvalidPathException

The specified path is not valid.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

InvalidReplacementContentException

Automerge was specified for resolving the conflict, but the replacement type is not valid or content is missing.

HTTP Status Code: 400

InvalidReplacementTypeException

Automerge was specified for resolving the conflict, but the specified replacement type is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

ManualMergeRequiredException

The pull request cannot be merged automatically into the destination branch. You must manually merge the branches and resolve any conflicts.

HTTP Status Code: 400

MaximumConflictResolutionEntriesExceededException

The number of allowed conflict resolution entries was exceeded.

HTTP Status Code: 400

MaximumFileContentToLoadExceededException

The number of files to load exceeds the allowed limit.

HTTP Status Code: 400

MaximumItemsToCompareExceededException

The number of items to compare between the source or destination branches and the merge base has exceeded the maximum allowed.

HTTP Status Code: 400

MultipleConflictResolutionEntriesException

More than one conflict resolution entries exists for the conflict. A conflict can have only one conflict resolution entry.

HTTP Status Code: 400

NameLengthExceededException

The user name is not valid because it has exceeded the character limit for author names.

HTTP Status Code: 400

PathRequiredException

The folderPath for a location cannot be null.

HTTP Status Code: 400

PullRequestAlreadyClosedException

The pull request status cannot be updated because it is already closed.

HTTP Status Code: 400

PullRequestApprovalRulesNotSatisfiedException

The pull request cannot be merged because one or more approval rules applied to the pull request have conditions that have not been met.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

ReplacementContentRequiredException

USE_NEW_CONTENT was specified, but no replacement content has been provided.

HTTP Status Code: 400

ReplacementTypeRequiredException

A replacement type is required.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

RepositoryNotAssociatedWithPullRequestException

The repository does not contain any pull requests with that pull request ID. Use GetPullRequest to verify the correct repository name for the pull request ID.

HTTP Status Code: 400

TipOfSourceReferenceIsDifferentException

The tip of the source branch in the destination repository does not match the tip of the source branch specified in your request. The pull request might have been updated. Make sure that you have the latest changes.

HTTP Status Code: 400

TipsDivergenceExceededException

The divergence between the tips of the provided commit specifiers is too great to determine whether there might be any merge conflicts. Locally compare the specifiers using `git diff` or a diff tool.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of MergePullRequestBySquash.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.MergePullRequestBySquash
X-Amz-Date: 20190428T213222Z
User-Agent: aws-cli/1.16.137 Python/3.6.0 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "pullRequestId": "47",
   "repositoryName": "MyDemoRepo",
   "sourceCommitId": "99132ab0EXAMPLE",
   "authorName": "Mary Major"
   "email": "mary_major@example.com",
   "commitMessage": "Merging this pull request after code review and acceptance tests
 passed."
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Sun, 28 Apr 2019 22:43:13 GMT

{
    "pullRequest": {
        "approvalRules": [
            {
                "approvalRuleContent": "{\"Version\": \"2018-11-08\",
\"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
                "approvalRuleId": "dd8b17fe-EXAMPLE",
                "approvalRuleName": "2-approver-rule-for-main",
                "creationDate": 1571356106.936,
                "lastModifiedDate": 571356106.936,
                "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
                "originApprovalRuleTemplate": {
                    "approvalRuleTemplateId": "dd8b17fe-EXAMPLE",
                    "approvalRuleTemplateName": "2-approver-rule-for-main"
                },
                "ruleContentSha256": "4711b576EXAMPLE"
            }
        ],
        "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
        "clientRequestToken": "",
        "creationDate": 1508530823.142,
        "description": "Review the latest changes and updates to the global variables",
        "lastActivityDate": 1508887223.155,
        "pullRequestId": "47",
        "pullRequestStatus": "CLOSED",
        "pullRequestTargets": [
            {
                "destinationCommit": "9f31c968EXAMPLE",
                "destinationReference": "refs/heads/main",
                "mergeMetadata": {
                    "isMerged": true,
                    "mergedBy": "arn:aws:iam::123456789012:user/Mary_Major"
                },
                "repositoryName": "MyDemoRepo",
                "sourceCommit": "99132ab0EXAMPLE",
                "sourceReference": "refs/heads/variables-branch"
            }
        ],
        "title": "Consolidation of global variables"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

MergePullRequestByThreeWay

Attempts to merge the source commit of a pull request into the specified destination branch for that pull request at the specified commit using the three-way merge strategy. If the merge is successful, it closes the pull request.

Request Syntax

```
{
   "authorName": "string",
   "commitMessage": "string",
   "conflictDetailLevel": "string",
   "conflictResolution": {
      "deleteFiles": [
         {
            "filePath": "string"
         }
      ],
      "replaceContents": [
         {
            "content": blob,
            "fileMode": "string",
            "filePath": "string",
            "replacementType": "string"
         }
      ],
      "setFileModes": [
         {
            "fileMode": "string",
            "filePath": "string"
         }
      ]
   },
   "conflictResolutionStrategy": "string",
   "email": "string",
   "keepEmptyFolders": boolean,
   "pullRequestId": "string",
   "repositoryName": "string",
   "sourceCommitId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

authorName

The name of the author who created the commit. This information is used as both the author and committer for the commit.

Type: String

Required: No

commitMessage

The commit message to include in the commit information for the merge.

Type: String

Required: No

conflictDetailLevel

The level of conflict detail to use. If unspecified, the default FILE_LEVEL is used, which returns a not-mergeable result if the same file has differences in both branches. If LINE_LEVEL is specified, a conflict is considered not mergeable if the same file in both branches has differences on the same line.

Type: String

Valid Values: `FILE_LEVEL | LINE_LEVEL`

Required: No

conflictResolution

If AUTOMERGE is the conflict resolution strategy, a list of inputs to use when resolving conflicts during a merge.

Type: ConflictResolution object

Required: No

conflictResolutionStrategy

Specifies which branch to use when resolving conflicts, or whether to attempt automatically merging two versions of a file. The default is NONE, which requires any conflicts to be resolved manually before the merge operation is successful.

Type: String

Valid Values: `NONE | ACCEPT_SOURCE | ACCEPT_DESTINATION | AUTOMERGE`

Required: No

email

The email address of the person merging the branches. This information is used in the commit information for the merge.

Type: String

Required: No

keepEmptyFolders

If the commit contains deletions, whether to keep a folder or folder structure if the changes leave the folders empty. If true, a .gitkeep file is created for empty folders. The default is false.

Type: Boolean

Required: No

pullRequestId

The system-generated ID of the pull request. To get this ID, use ListPullRequests.

Type: String

Required: Yes

repositoryName

The name of the repository where the pull request was created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

sourceCommitId

The full commit ID of the original or updated commit in the pull request source branch. Pass this value if you want an exception thrown if the current commit ID of the tip of the source branch does not match this commit ID.

Type: String

Required: No

Response Syntax

```
{
   "pullRequest": {
      "approvalRules": [
         {
            "approvalRuleContent": "string",
            "approvalRuleId": "string",
            "approvalRuleName": "string",
            "creationDate": number,
            "lastModifiedDate": number,
            "lastModifiedUser": "string",
            "originApprovalRuleTemplate": {
               "approvalRuleTemplateId": "string",
               "approvalRuleTemplateName": "string"
            },
            "ruleContentSha256": "string"
         }
      ],
      "authorArn": "string",
      "clientRequestToken": "string",
      "creationDate": number,
      "description": "string",
      "lastActivityDate": number,
      "pullRequestId": "string",
      "pullRequestStatus": "string",
      "pullRequestTargets": [
         {
            "destinationCommit": "string",
            "destinationReference": "string",
            "mergeBase": "string",
            "mergeMetadata": { 
               "isMerged": boolean,
               "mergeCommitId": "string",
               "mergedBy": "string",
               "mergeOption": "string"
            },
            "repositoryName": "string",
            "sourceCommit": "string",
            "sourceReference": "string"
         }
      ],
      "revisionId": "string",
      "title": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

pullRequest

Returns information about a pull request.

Type: PullRequest object

Errors

For information about the errors that are common to all actions, see Common Errors.

CommitMessageLengthExceededException

The commit message is too long. Provide a shorter string.

HTTP Status Code: 400

ConcurrentReferenceUpdateException

The merge cannot be completed because the target branch has been modified. Another user might have modified the target branch while the merge was in progress. Wait a few minutes, and then try again.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

FileContentSizeLimitExceededException

The file cannot be added because it is too large. The maximum file size is 6 MB, and the combined file content change size is 7 MB. Consider making these changes using a Git client.

HTTP Status Code: 400

FolderContentSizeLimitExceededException

The commit cannot be created because at least one of the overall changes in the commit results in a folder whose contents exceed the limit of 6 MB. Either reduce the number and size of your changes, or split the changes across multiple folders.

HTTP Status Code: 400

InvalidCommitIdException

The specified commit ID is not valid.

HTTP Status Code: 400

InvalidConflictDetailLevelException

The specified conflict detail level is not valid.

HTTP Status Code: 400

InvalidConflictResolutionException

The specified conflict resolution list is not valid.

HTTP Status Code: 400

InvalidConflictResolutionStrategyException

The specified conflict resolution strategy is not valid.

HTTP Status Code: 400

InvalidEmailException

The specified email address either contains one or more characters that are not allowed, or it exceeds the maximum number of characters allowed for an email address.

HTTP Status Code: 400

InvalidFileModeException

The specified file mode permission is not valid. For a list of valid file mode permissions, see PutFile.

HTTP Status Code: 400

InvalidPathException

The specified path is not valid.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

InvalidReplacementContentException

Automerge was specified for resolving the conflict, but the replacement type is not valid or content is missing.

HTTP Status Code: 400

InvalidReplacementTypeException

Automerge was specified for resolving the conflict, but the specified replacement type is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

ManualMergeRequiredException

The pull request cannot be merged automatically into the destination branch. You must manually merge the branches and resolve any conflicts.

HTTP Status Code: 400

MaximumConflictResolutionEntriesExceededException

The number of allowed conflict resolution entries was exceeded.

HTTP Status Code: 400

MaximumFileContentToLoadExceededException

The number of files to load exceeds the allowed limit.

HTTP Status Code: 400

MaximumItemsToCompareExceededException

The number of items to compare between the source or destination branches and the merge base has exceeded the maximum allowed.

HTTP Status Code: 400

MultipleConflictResolutionEntriesException

More than one conflict resolution entries exists for the conflict. A conflict can have only one conflict resolution entry.

HTTP Status Code: 400

NameLengthExceededException

The user name is not valid because it has exceeded the character limit for author names.

HTTP Status Code: 400

PathRequiredException

The folderPath for a location cannot be null.

HTTP Status Code: 400

PullRequestAlreadyClosedException

The pull request status cannot be updated because it is already closed.

HTTP Status Code: 400

PullRequestApprovalRulesNotSatisfiedException

The pull request cannot be merged because one or more approval rules applied to the pull request have conditions that have not been met.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

ReplacementContentRequiredException

USE_NEW_CONTENT was specified, but no replacement content has been provided.

HTTP Status Code: 400

ReplacementTypeRequiredException

A replacement type is required.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

RepositoryNotAssociatedWithPullRequestException

The repository does not contain any pull requests with that pull request ID. Use GetPullRequest to verify the correct repository name for the pull request ID.

HTTP Status Code: 400

TipOfSourceReferenceIsDifferentException

The tip of the source branch in the destination repository does not match the tip of the source branch specified in your request. The pull request might have been updated. Make sure that you have the latest changes.

HTTP Status Code: 400

TipsDivergenceExceededException

The divergence between the tips of the provided commit specifiers is too great to determine whether there might be any merge conflicts. Locally compare the specifiers using `git diff` or a diff tool.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of MergePullRequestByThreeWay.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.MergePullRequestByThreeWay
X-Amz-Date: 20190428T213222Z
User-Agent: aws-cli/1.16.137 Python/3.6.0 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "pullRequestId": "47",
   "repositoryName": "MyDemoRepo",
   "sourceCommitId": "99132ab0EXAMPLE",
   "authorName": "Mary Major"
   "email": "mary_major@example.com",
   "commitMessage": "Merging this pull request after code review and acceptance tests
 passed."
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Sun, 28 Apr 2019 22:43:13 GMT

{
    "pullRequest": {
        "approvalRules": [
            {
                "approvalRuleContent": "{\"Version\": \"2018-11-08\",
\"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
                "approvalRuleId": "dd8b17fe-EXAMPLE",
                "approvalRuleName": "2-approver-rule-for-main",
                "creationDate": 1571356106.936,
                "lastModifiedDate": 571356106.936,
                "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
                "originApprovalRuleTemplate": {
                    "approvalRuleTemplateId": "dd8b17fe-EXAMPLE",
                    "approvalRuleTemplateName": "2-approver-rule-for-main"
                },
                "ruleContentSha256": "4711b576EXAMPLE"
            }
        ],
        "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
        "clientRequestToken": "",
        "creationDate": 1508530823.142,
        "description": "Review the latest changes and updates to the global variables",
        "lastActivityDate": 1508887223.155,
        "pullRequestId": "47",
        "pullRequestStatus": "CLOSED",
        "pullRequestTargets": [
            {
                "destinationCommit": "9f31c968EXAMPLE",
                "destinationReference": "refs/heads/main",
                "mergeMetadata": {
                    "isMerged": true,
                    "mergedBy": "arn:aws:iam::123456789012:user/Mary_Major"
                },
                "repositoryName": "MyDemoRepo",
                "sourceCommit": "99132ab0EXAMPLE",
                "sourceReference": "refs/heads/variables-branch"
            }
        ],
        "title": "Consolidation of global variables"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

OverridePullRequestApprovalRules

Sets aside (overrides) all approval rule requirements for a specified pull request.

Request Syntax

```
{
   "overrideStatus": "string",
   "pullRequestId": "string",
   "revisionId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

overrideStatus

Whether you want to set aside approval rule requirements for the pull request (OVERRIDE) or revoke a previous override and apply approval rule requirements (REVOKE). REVOKE status is not stored.

Type: String

Valid Values: `OVERRIDE | REVOKE`

Required: Yes

pullRequestId

The system-generated ID of the pull request for which you want to override all approval rule requirements. To get this information, use GetPullRequest.

Type: String

Required: Yes

revisionId

The system-generated ID of the most recent revision of the pull request. You cannot override approval rules for anything but the most recent revision of a pull request. To get the revision ID, use GetPullRequest.

Type: String

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidOverrideStatusException

The override status is not valid. Valid statuses are OVERRIDE and REVOKE.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

InvalidRevisionIdException

The revision ID is not valid. Use GetPullRequest to determine the value.

HTTP Status Code: 400

OverrideAlreadySetException

The pull request has already had its approval rules set to override.

HTTP Status Code: 400

OverrideStatusRequiredException

An override status is required, but no value was provided. Valid values include OVERRIDE and REVOKE.

HTTP Status Code: 400

PullRequestAlreadyClosedException

The pull request status cannot be updated because it is already closed.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

RevisionIdRequiredException

A revision ID is required, but was not provided.

HTTP Status Code: 400

RevisionNotCurrentException

The revision ID provided in the request does not match the current revision ID. Use GetPullRequest to retrieve the current revision ID.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of OverridePullRequestApprovalRules.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.OverridePullRequestApprovalRules
X-Amz-Date: 20191021T213222Z
User-Agent: aws-cli/1.16.137 Python/3.6.0 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "pullRequestId": "34",
   "revisionId": "927df8d8dEXAMPLE",
   "overrideStatus": OVERRIDE
}
```

Sample Response

```
HTTP/1.1 200 OK
```

```
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Mon, 21 Oct 2019 22:43:13 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

PostCommentForComparedCommit

Posts a comment on the comparison between two commits.

Request Syntax

```
{
   "afterCommitId": "string",
   "beforeCommitId": "string",
   "clientRequestToken": "string",
   "content": "string",
   "location": {
      "filePath": "string",
      "filePosition": number,
      "relativeFileVersion": "string"
   },
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

afterCommitId

To establish the directionality of the comparison, the full commit ID of the after commit.

Type: String

Required: Yes

beforeCommitId

To establish the directionality of the comparison, the full commit ID of the before commit. Required for commenting on any commit unless that commit is the initial commit.

Type: String

Required: No

clientRequestToken

A unique, client-generated idempotency token that, when provided in a request, ensures the request cannot be repeated with a changed parameter. If a request is received with the same parameters and a token is included, the request returns information about the initial request that used that token.

Type: String

Required: No

content

The content of the comment you want to make.

Type: String

Required: Yes

location

The location of the comparison where you want to comment.

Type: Location object

Required: No

repositoryName

The name of the repository where you want to post a comment on the comparison between commits.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "afterBlobId": "string",
   "afterCommitId": "string",
   "beforeBlobId": "string",
   "beforeCommitId": "string",
   "comment": { 
      "authorArn": "string",
      "callerReactions": [ "string" ],
      "clientRequestToken": "string",
      "commentId": "string",
      "content": "string",
      "creationDate": number,
      "deleted": boolean,
      "inReplyTo": "string",
      "lastModifiedDate": number,
      "reactionCounts": { 
         "string" : number 
      }
   },
   "location": { 
      "filePath": "string",
      "filePosition": number,
      "relativeFileVersion": "string"
   },
   "repositoryName": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

afterBlobId

In the directionality you established, the blob ID of the after blob.

Type: String

afterCommitId

In the directionality you established, the full commit ID of the after commit.

Type: String

beforeBlobId

In the directionality you established, the blob ID of the before blob.

Type: String

beforeCommitId

In the directionality you established, the full commit ID of the before commit.

Type: String

comment

The content of the comment you posted.

Type: Comment object

location

The location of the comment in the comparison between the two commits.

Type: Location object

repositoryName

The name of the repository where you posted a comment on the comparison between commits.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Errors

For information about the errors that are common to all actions, see Common Errors.

BeforeCommitIdAndAfterCommitIdAreSameException

The before commit ID and the after commit ID are the same, which is not valid. The before commit ID and the after commit ID must be different commit IDs.

HTTP Status Code: 400

ClientRequestTokenRequiredException

A client request token is required. A client request token is an unique, client-generated idempotency token that, when provided in a request, ensures the request cannot be repeated with a changed parameter. If a request is received with the same parameters and a token is included, the request returns information about the initial request that used that token.

HTTP Status Code: 400

CommentContentRequiredException

The comment is empty. You must provide some content for a comment. The content cannot be null.

HTTP Status Code: 400

CommentContentSizeLimitExceededException

The comment is too large. Comments are limited to 10,240 characters.

HTTP Status Code: 400

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

CommitIdRequiredException

A commit ID was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

IdempotencyParameterMismatchException

The client request token is not valid. Either the token is not in a valid format, or the token has been used in a previous request and cannot be reused.

HTTP Status Code: 400

InvalidClientRequestTokenException

The client request token is not valid.

HTTP Status Code: 400

InvalidCommitIdException

The specified commit ID is not valid.

HTTP Status Code: 400

InvalidFileLocationException

The location of the file is not valid. Make sure that you include the file name and extension.

HTTP Status Code: 400

InvalidFilePositionException

The position is not valid. Make sure that the line number exists in the version of the file you want to comment on.

HTTP Status Code: 400

InvalidPathException

The specified path is not valid.

HTTP Status Code: 400

InvalidRelativeFileVersionEnumException

Either the enum is not in a valid format, or the specified file version enum is not valid in respect to the current file version.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

PathDoesNotExistException

The specified path does not exist.

HTTP Status Code: 400

PathRequiredException

The folderPath for a location cannot be null.

HTTP Status Code: 400

PathRequiredException

The folderPath for a location cannot be null.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of PostCommentForComparedCommit.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.PostCommentForComparedCommit
X-Amz-Date: 20171025T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "afterCommitId": "317f8570EXAMPLE",
   "beforeCommitId": "6e147360EXAMPLE",
   "clientRequestToken": "123Example",
   "content": "Can you add a test case for this?",
   "location": {
            "filePath": "cl_sample.js",
            "filePosition": 1232,
            "relativeFileVersion": "AFTER"
         },
   "repositoryName": "MyDemoRepo"
}
```

Sample Response

```
{
        "afterBlobId": "1f330709EXAMPLE",
        "afterCommitId": "317f8570EXAMPLE",
        "beforeBlobId": "80906a4cEXAMPLE",
        "beforeCommitId": "6e147360EXAMPLE",
        "comment": {
              "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
              "clientRequestToken": "",
              "commentId": "553b509bEXAMPLE56198325",
              "content": "Can you add a test case for this?",
              "creationDate": 1508369612.203,
              "deleted": false,
              "commentId": "abc123-EXAMPLE",
              "lastModifiedDate": 1508369612.203,
              "callerReactions": [],
              "reactionCounts": []
            },
            "location": {
              "filePath": "cl_sample.js",
              "filePosition": 1232,
              "relativeFileVersion": "AFTER"
            },
        "repositoryName": "MyDemoRepo"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

PostCommentForPullRequest

Posts a comment on a pull request.

Request Syntax

```
{
   "afterCommitId": "string",
   "beforeCommitId": "string",
   "clientRequestToken": "string",
   "content": "string",
   "location": {
      "filePath": "string",
      "filePosition": number,
      "relativeFileVersion": "string"
   },
   "pullRequestId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

afterCommitId

The full commit ID of the commit in the source branch that is the current tip of the branch for the pull request when you post the comment.

Type: String

Required: Yes

beforeCommitId

The full commit ID of the commit in the destination branch that was the tip of the branch at the time the pull request was created.

Type: String

Required: Yes

clientRequestToken

A unique, client-generated idempotency token that, when provided in a request, ensures the request cannot be repeated with a changed parameter. If a request is received with the same parameters and a token is included, the request returns information about the initial request that used that token.

Type: String

Required: No

content

The content of your comment on the change.

Type: String

Required: Yes

location

The location of the change where you want to post your comment. If no location is provided, the comment is posted as a general comment on the pull request difference between the before commit ID and the after commit ID.

Type: Location object

Required: No

pullRequestId

The system-generated ID of the pull request. To get this ID, use ListPullRequests.

Type: String

Required: Yes

repositoryName

The name of the repository where you want to post a comment on a pull request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "afterBlobId": "string",
   "afterCommitId": "string",
   "beforeBlobId": "string",
   "beforeCommitId": "string",
   "comment": {
      "authorArn": "string",
      "callerReactions": [ "string" ],
      "clientRequestToken": "string",
      "commentId": "string",
      "content": "string",
      "creationDate": number,
      "deleted": boolean,
      "inReplyTo": "string",
      "lastModifiedDate": number,
      "reactionCounts": {
         "string" : number
      }
   },
   "location": {
      "filePath": "string",
      "filePosition": number,
      "relativeFileVersion": "string"
   },
   "pullRequestId": "string",
   "repositoryName": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

afterBlobId

In the directionality of the pull request, the blob ID of the after blob.

Type: String

afterCommitId

The full commit ID of the commit in the destination branch where the pull request is merged.

Type: String

beforeBlobId

In the directionality of the pull request, the blob ID of the before blob.

Type: String

beforeCommitId

The full commit ID of the commit in the source branch used to create the pull request, or in the case of an updated pull request, the full commit ID of the commit used to update the pull request.

Type: String

comment

The content of the comment you posted.

Type: Comment object

location

The location of the change where you posted your comment.

Type: Location object

pullRequestId

The system-generated ID of the pull request.

Type: String

repositoryName

The name of the repository where you posted a comment on a pull request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Errors

For information about the errors that are common to all actions, see Common Errors.

BeforeCommitIdAndAfterCommitIdAreSameException

The before commit ID and the after commit ID are the same, which is not valid. The before commit ID and the after commit ID must be different commit IDs.

HTTP Status Code: 400

ClientRequestTokenRequiredException

A client request token is required. A client request token is an unique, client-generated idempotency token that, when provided in a request, ensures the request cannot be repeated with a changed parameter. If a request is received with the same parameters and a token is included, the request returns information about the initial request that used that token.

HTTP Status Code: 400

CommentContentRequiredException

The comment is empty. You must provide some content for a comment. The content cannot be null.

HTTP Status Code: 400

CommentContentSizeLimitExceededException

The comment is too large. Comments are limited to 10,240 characters.

HTTP Status Code: 400

CommitDoesNotExistException

The specified commit does not exist or no commit was specified, and the specified repository has no default branch.

HTTP Status Code: 400

CommitIdRequiredException

A commit ID was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

IdempotencyParameterMismatchException

The client request token is not valid. Either the token is not in a valid format, or the token has been used in a previous request and cannot be reused.

HTTP Status Code: 400

InvalidClientRequestTokenException

The client request token is not valid.

HTTP Status Code: 400

InvalidCommitIdException

The specified commit ID is not valid.

HTTP Status Code: 400

InvalidFileLocationException

The location of the file is not valid. Make sure that you include the file name and extension.

HTTP Status Code: 400

InvalidFilePositionException

The position is not valid. Make sure that the line number exists in the version of the file you want to comment on.

HTTP Status Code: 400

InvalidPathException

The specified path is not valid.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

InvalidRelativeFileVersionEnumException

Either the enum is not in a valid format, or the specified file version enum is not valid in respect to the current file version.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

PathDoesNotExistException

The specified path does not exist.

HTTP Status Code: 400

PathRequiredException

The folderPath for a location cannot be null.

HTTP Status Code: 400

PathRequiredException

The folderPath for a location cannot be null.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

RepositoryNotAssociatedWithPullRequestException

The repository does not contain any pull requests with that pull request ID. Use GetPullRequest to verify the correct repository name for the pull request ID.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of PostCommentForPullRequest.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.PostCommentForPullRequest
X-Amz-Date: 20171025T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "afterCommitId": "5d036259EXAMPLE",
   "beforeCommitId": "317f8570EXAMPLE",
   "clientRequestToken": "123Example",
   "content": "These don't appear to be used anywhere. Can we remove them?",
   "location": {
            "filePath": "ahs_count.py",
            "filePosition": 367,
            "relativeFileVersion": "AFTER"
         },
   "pullRequestId": 47,
   "repositoryName": "MyDemoRepo"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Wed, 25 Oct 2017 20:20:13 GMT

{
         "afterBlobId": "1f330709EXAMPLE",
         "afterCommitId": "5d036259EXAMPLE",
         "beforeBlobId": "80906a4cEXAMPLE",
         "beforeCommitId": "317f8570EXAMPLE",
         "comment": {
               "authorArn": "arn:aws:iam::123456789012:user/Saanvi_Sarkar",
               "clientRequestToken": "",
               "commentId": "abcd1234EXAMPLEb5678efgh",
               "content": "These don't appear to be used anywhere. Can we remove
 them?",
               "creationDate": 1508369622.123,
               "deleted": false,
               "lastModifiedDate": 1508369622.123,
                "callerReactions": [],
                "reactionCounts": []
            }
             "location": {
               "filePath": "ahs_count.py",
               "filePosition": 367,
               "relativeFileVersion": "AFTER"
             },
         "repositoryName": "MyDemoRepo",
         "pullRequestId": "47"
 }
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

PostCommentReply

Posts a comment in reply to an existing comment on a comparison between commits or a pull request.

Request Syntax

```
{
   "clientRequestToken": "string",
   "content": "string",
   "inReplyTo": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

clientRequestToken

A unique, client-generated idempotency token that, when provided in a request, ensures the request cannot be repeated with a changed parameter. If a request is received with the same parameters and a token is included, the request returns information about the initial request that used that token.

Type: String

Required: No

content

The contents of your reply to a comment.

Type: String

Required: Yes

inReplyTo

The system-generated ID of the comment to which you want to reply. To get this ID, use GetCommentsForComparedCommit or GetCommentsForPullRequest.

Type: String

Required: Yes

Response Syntax

```
{
   "comment": {
      "authorArn": "string",
      "callerReactions": [ "string" ],
      "clientRequestToken": "string",
      "commentId": "string",
      "content": "string",
      "creationDate": number,
      "deleted": boolean,
      "inReplyTo": "string",
      "lastModifiedDate": number,
      "reactionCounts": {
         "string" : number
      }
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

comment

Information about the reply to a comment.

Type: Comment object

Errors

For information about the errors that are common to all actions, see Common Errors.

ClientRequestTokenRequiredException

A client request token is required. A client request token is an unique, client-generated idempotency token that, when provided in a request, ensures the request cannot be repeated with a changed parameter. If a request is received with the same parameters and a token is included, the request returns information about the initial request that used that token.

HTTP Status Code: 400

CommentContentRequiredException

The comment is empty. You must provide some content for a comment. The content cannot be null.

HTTP Status Code: 400

CommentContentSizeLimitExceededException

The comment is too large. Comments are limited to 10,240 characters.

HTTP Status Code: 400

CommentDoesNotExistException

No comment exists with the provided ID. Verify that you have used the correct ID, and then try again.

HTTP Status Code: 400

CommentIdRequiredException

The comment ID is missing or null. A comment ID is required.

HTTP Status Code: 400

IdempotencyParameterMismatchException

The client request token is not valid. Either the token is not in a valid format, or the token has been used in a previous request and cannot be reused.

HTTP Status Code: 400

InvalidClientRequestTokenException

The client request token is not valid.

HTTP Status Code: 400

InvalidCommentIdException

The comment ID is not in a valid format. Make sure that you have provided the full comment ID.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of PostCommentReply.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.PostCommentReply
X-Amz-Date: 20171025T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "clientRequestToken": "123Example",
   "content": "Good catch. I'll remove them.",
   "commentId": "abcd1234EXAMPLEb5678efgh"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Wed, 25 Oct 2017 20:20:13 GMT

{
    "comment": {
        "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
        "clientRequestToken": "123Example",
        "commentId": "442b498bEXAMPLE5756813",
        "content": "Good catch. I'll remove them.",
        "creationDate": 1508369829.136,
        "deleted": false,
        "lastModifiedDate": 150836912.221,
        "callerReactions": [],
        "reactionCounts": []
    }
 }
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

PutCommentReaction

Adds or updates a reaction to a specified comment for the user whose identity is used to make the request. You can only add or update a reaction for yourself. You cannot add, modify, or delete a reaction for another user.

Request Syntax

```
{
   "commentId": "string",
   "reactionValue": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

commentId

The ID of the comment to which you want to add or update a reaction.

Type: String

Required: Yes

reactionValue

The emoji reaction you want to add or update. To remove a reaction, provide a value of blank or null. You can also provide the value of none. For information about emoji reaction values supported in AWS CodeCommit, see the AWS CodeCommit User Guide.

Type: String

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors.

CommentDeletedException

This comment has already been deleted. You cannot edit or delete a deleted comment.

HTTP Status Code: 400

CommentDoesNotExistException

No comment exists with the provided ID. Verify that you have used the correct ID, and then try again.

HTTP Status Code: 400

CommentIdRequiredException

The comment ID is missing or null. A comment ID is required.

HTTP Status Code: 400

InvalidCommentIdException

The comment ID is not in a valid format. Make sure that you have provided the full comment ID.

HTTP Status Code: 400

InvalidReactionValueException

The value of the reaction is not valid. For more information, see the AWS CodeCommit User Guide.

HTTP Status Code: 400

ReactionLimitExceededException

The number of reactions has been exceeded. Reactions are limited to one reaction per user for each individual comment ID.

HTTP Status Code: 400

ReactionValueRequiredException

A reaction value is required.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of PutCommentReaction.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.PutCommentReaction
X-Amz-Date: 20200406T224019Z
User-Agent: aws-cli/1.18.36 Python/3.6.10 Linux/4.9.184-0.1.ac.235.83.329.metal1.x86_64
 botocore/1.15.36
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "commentId": "ff30b348EXAMPLE1EXAMPLE",
   "reaction:: ":thumbsup:"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 0
Date: Mon, 06 Apr 2020 22:40:19 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

PutFile

Adds or updates a file in a branch in an AWS CodeCommit repository, and generates a commit for the addition in the specified branch.

Request Syntax

```
{
   "branchName": "string",
   "commitMessage": "string",
   "email": "string",
   "fileContent": blob,
   "fileMode": "string",
   "filePath": "string",
   "name": "string",
   "parentCommitId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

branchName

The name of the branch where you want to add or update the file. If this is an empty repository, this branch is created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: Yes

commitMessage

A message about why this file was added or updated. Although it is optional, a message makes the commit history for your repository more useful.

Type: String

Required: No

email

An email address for the person adding or updating the file.

Type: String

Required: No

fileContent

The content of the file, in binary object format.

Type: Base64-encoded binary data object

Length Constraints: Maximum length of 6291456.

Required: Yes

fileMode

The file mode permissions of the blob. Valid file mode permissions are listed here.

Type: String

Valid Values: `EXECUTABLE | NORMAL | SYMLINK`

Required: No

filePath

The name of the file you want to add or update, including the relative path to the file in the repository.

> **Note**
>
> If the path does not currently exist in the repository, the path is created as part of adding the file.

Type: String

Required: Yes

name

The name of the person adding or updating the file. Although it is optional, a name makes the commit history for your repository more useful.

Type: String

Required: No

parentCommitId

The full commit ID of the head commit in the branch where you want to add or update the file. If this is an empty repository, no commit ID is required. If this is not an empty repository, a commit ID is required.

The commit ID must match the ID of the head commit at the time of the operation. Otherwise, an error occurs, and the file is not added or updated.

Type: String

Required: No

repositoryName

The name of the repository where you want to add or update the file.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "blobId": "string",
   "commitId": "string",
   "treeId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

blobId

The ID of the blob, which is its SHA-1 pointer.

Type: String

commitId

The full SHA ID of the commit that contains this file change.

Type: String

treeId

The full SHA-1 pointer of the tree information for the commit that contains this file change.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

BranchDoesNotExistException

The specified branch does not exist.

HTTP Status Code: 400

BranchNameIsTagNameException

The specified branch name is not valid because it is a tag name. Enter the name of a branch in the repository. For a list of valid branch names, use ListBranches.

HTTP Status Code: 400

BranchNameRequiredException

A branch name is required, but was not specified.

HTTP Status Code: 400

CommitMessageLengthExceededException

The commit message is too long. Provide a shorter string.

HTTP Status Code: 400

DirectoryNameConflictsWithFileNameException

A file cannot be added to the repository because the specified path name has the same name as a file that already exists in this repository. Either provide a different name for the file, or specify a different path for the file.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

FileContentRequiredException

The file cannot be added because it is empty. Empty files cannot be added to the repository with this API.

HTTP Status Code: 400

FileContentSizeLimitExceededException

The file cannot be added because it is too large. The maximum file size is 6 MB, and the combined file content change size is 7 MB. Consider making these changes using a Git client.

HTTP Status Code: 400

FileNameConflictsWithDirectoryNameException

A file cannot be added to the repository because the specified file name has the same name as a directory in this repository. Either provide another name for the file, or add the file in a directory that does not match the file name.

HTTP Status Code: 400

FilePathConflictsWithSubmodulePathException

The commit cannot be created because a specified file path points to a submodule. Verify that the destination files have valid file paths that do not point to a submodule.

HTTP Status Code: 400

FolderContentSizeLimitExceededException

The commit cannot be created because at least one of the overall changes in the commit results in a folder whose contents exceed the limit of 6 MB. Either reduce the number and size of your changes, or split the changes across multiple folders.

HTTP Status Code: 400

InvalidBranchNameException

The specified reference name is not valid.

HTTP Status Code: 400

InvalidDeletionParameterException

The specified deletion parameter is not valid.

HTTP Status Code: 400

InvalidEmailException

The specified email address either contains one or more characters that are not allowed, or it exceeds the maximum number of characters allowed for an email address.

HTTP Status Code: 400

InvalidFileModeException

The specified file mode permission is not valid. For a list of valid file mode permissions, see PutFile.

HTTP Status Code: 400

InvalidParentCommitIdException

The parent commit ID is not valid. The commit ID cannot be empty, and must match the head commit ID for the branch of the repository where you want to add or update a file.

HTTP Status Code: 400

InvalidPathException

The specified path is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

NameLengthExceededException

The user name is not valid because it has exceeded the character limit for author names.

HTTP Status Code: 400

ParentCommitDoesNotExistException

The parent commit ID is not valid because it does not exist. The specified parent commit ID does not exist in the specified branch of the repository.

HTTP Status Code: 400

ParentCommitIdOutdatedException

The file could not be added because the provided parent commit ID is not the current tip of the specified branch. To view the full commit ID of the current head of the branch, use GetBranch.

HTTP Status Code: 400

ParentCommitIdRequiredException

A parent commit ID is required. To view the full commit ID of a branch in a repository, use GetBranch or a Git command (for example, git pull or git log).

HTTP Status Code: 400

PathRequiredException

The folderPath for a location cannot be null.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

SameFileContentException

The file was not added or updated because the content of the file is exactly the same as the content of that file in the repository and branch that you specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of PutFile.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.PutFile
X-Amz-Date: 20180126T224019Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "branchName": "MyFeatureBranch",
   "commitMessage": "I'm adding this file to fix a bug Mary found.",
   "email": "maria_garcia@example.com",
   "fileContent": QSBCaW5hcnkgTGFyToEXAMPLE=,
   "fileMode": "NORMAL",
   "filePath": "ExampleSolution.py",
   "name": "Maria Garcia",
   "parentCommitId": "4c925148EXAMPLE",
   "repositoryName": "MyDemoRepo"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Fri, 26 Jan 2018 22:40:19 GMT

{
   "blobId": "2eb4af3bEXAMPLE",
   "commitId": "317f8570EXAMPLE",
    "treeId": "347a3408EXAMPLE"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

PutRepositoryTriggers

Replaces all triggers for a repository. Used to create or delete triggers.

Request Syntax

```
{
   "repositoryName": "string",
   "triggers": [
      {
         "branches": [ "string" ],
         "customData": "string",
         "destinationArn": "string",
         "events": [ "string" ],
         "name": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

repositoryName

The name of the repository where you want to create or update the trigger.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

triggers

The JSON block of configuration information for each trigger.

Type: Array of RepositoryTrigger objects

Required: Yes

Response Syntax

```
{
   "configurationId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

configurationId

The system-generated unique ID for the create or update operation.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

InvalidRepositoryTriggerBranchNameException

One or more branch names specified for the trigger is not valid.

HTTP Status Code: 400

InvalidRepositoryTriggerCustomDataException

The custom data provided for the trigger is not valid.

HTTP Status Code: 400

InvalidRepositoryTriggerDestinationArnException

The Amazon Resource Name (ARN) for the trigger is not valid for the specified destination. The most common reason for this error is that the ARN does not meet the requirements for the service type.

HTTP Status Code: 400

InvalidRepositoryTriggerEventsException

One or more events specified for the trigger is not valid. Check to make sure that all events specified match the requirements for allowed events.

HTTP Status Code: 400

InvalidRepositoryTriggerNameException

The name of the trigger is not valid.

HTTP Status Code: 400

InvalidRepositoryTriggerRegionException

The AWS Region for the trigger target does not match the AWS Region for the repository. Triggers must be created in the same AWS Region as the target for the trigger.

HTTP Status Code: 400

MaximumBranchesExceededException

The number of branches for the trigger was exceeded.

HTTP Status Code: 400

MaximumRepositoryTriggersExceededException

The number of triggers allowed for the repository was exceeded.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

RepositoryTriggerBranchNameListRequiredException

At least one branch name is required, but was not specified in the trigger configuration.

HTTP Status Code: 400

RepositoryTriggerDestinationArnRequiredException

A destination ARN for the target service for the trigger is required, but was not specified.

HTTP Status Code: 400

RepositoryTriggerEventsListRequiredException

At least one event for the trigger is required, but was not specified.

HTTP Status Code: 400

RepositoryTriggerNameRequiredException

A name for the trigger is required, but was not specified.

HTTP Status Code: 400

RepositoryTriggersListRequiredException

The list of triggers for the repository is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of PutRepositoryTriggers.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 468
X-Amz-Target: CodeCommit_20150413.PutRepositoryTriggers
X-Amz-Date: 20151028T232919Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "ThisisaDemo",
  "triggers": [
    {
   "destinationArn": "arn:aws:sns:us-east-1:123456789012EXAMPLE:MyCodeCommitTopic",
   "branches": [
     "main",
     "anotherbranch"
     ],
   "events": ["all"],
   "name": "MyFirstTrigger",
   "customData": ""
 },
 {
   "destinationArn": "arn:aws:sns:us-
east-1:123456789012EXAMPLE:MyOtherCodeCommitTopic",
   "branches": [],
   "events": ["all"],
   "name": "MySecondTrigger",
   "customData": "Use the example-example IRC channel to discuss development on this
 repo"
 }
  ]
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 58
Date: Wed, 28 Oct 2015 23:29:21 GMT

{
  "configurationId":"6fa51cd8-35c1-EXAMPLE"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2

- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

TagResource

Adds or updates tags for a resource in AWS CodeCommit. For a list of valid resources in AWS CodeCommit, see CodeCommit Resources and Operations in the *AWS CodeCommit User Guide*.

Request Syntax

```
{
   "resourceArn": "string",
   "tags": {
      "string" : "string"
   }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

resourceArn

The Amazon Resource Name (ARN) of the resource to which you want to add or update tags.

Type: String

Required: Yes

tags

The key-value pair to use when tagging this repository.

Type: String to string map

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors.

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

InvalidResourceArnException

The value for the resource ARN is not valid. For more information about resources in AWS CodeCommit, see CodeCommit Resources and Operations in the AWS CodeCommit User Guide.

HTTP Status Code: 400

InvalidSystemTagUsageException

The specified tag is not valid. Key names cannot be prefixed with aws:.

HTTP Status Code: 400

InvalidTagsMapException

The map of tags is not valid.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

ResourceArnRequiredException

A valid Amazon Resource Name (ARN) for an AWS CodeCommit resource is required. For a list of valid resources in AWS CodeCommit, see CodeCommit Resources and Operations in the AWS CodeCommit User Guide.

HTTP Status Code: 400

TagPolicyException

The tag policy is not valid.

HTTP Status Code: 400

TagsMapRequiredException

A map of tags is required.

HTTP Status Code: 400

TooManyTagsException

The maximum number of tags for an AWS CodeCommit resource has been exceeded.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of TagResource.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 2
X-Amz-Target: CodeCommit_20150413.TagResource
X-Amz-Date: 20190228T224019Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE
```

```
{
  "resourceArn": "arn:aws:codecommit:us-east-2:123456789012:MySecretProjectRepo"
  "tags": {
       "Status": "Secret",
       "Team": "Saanvi"
   }
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 186
Date: Thu, 28 Feb 2019 22:40:19 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

TestRepositoryTriggers

Tests the functionality of repository triggers by sending information to the trigger target. If real data is available in the repository, the test sends data from the last commit. If no data is available, sample data is generated.

Request Syntax

```
{
   "repositoryName": "string",
   "triggers": [
      {
         "branches": [ "string" ],
         "customData": "string",
         "destinationArn": "string",
         "events": [ "string" ],
         "name": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

repositoryName

The name of the repository in which to test the triggers.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

triggers

The list of triggers to test.

Type: Array of RepositoryTrigger objects

Required: Yes

Response Syntax

```
{
   "failedExecutions": [
      {
         "failureMessage": "string",
         "trigger": "string"
      }
   ],
   "successfulExecutions": [ "string" ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

failedExecutions

The list of triggers that were not tested. This list provides the names of the triggers that could not be tested, separated by commas.

Type: Array of RepositoryTriggerExecutionFailure objects

successfulExecutions

The list of triggers that were successfully tested. This list provides the names of the triggers that were successfully tested, separated by commas.

Type: Array of strings

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

InvalidRepositoryTriggerBranchNameException

One or more branch names specified for the trigger is not valid.

HTTP Status Code: 400

InvalidRepositoryTriggerCustomDataException

The custom data provided for the trigger is not valid.

HTTP Status Code: 400

InvalidRepositoryTriggerDestinationArnException

The Amazon Resource Name (ARN) for the trigger is not valid for the specified destination. The most common reason for this error is that the ARN does not meet the requirements for the service type.

HTTP Status Code: 400

InvalidRepositoryTriggerEventsException

One or more events specified for the trigger is not valid. Check to make sure that all events specified match the requirements for allowed events.

HTTP Status Code: 400

InvalidRepositoryTriggerNameException

The name of the trigger is not valid.

HTTP Status Code: 400

InvalidRepositoryTriggerRegionException

The AWS Region for the trigger target does not match the AWS Region for the repository. Triggers must be created in the same AWS Region as the target for the trigger.

HTTP Status Code: 400

MaximumBranchesExceededException

The number of branches for the trigger was exceeded.

HTTP Status Code: 400

MaximumRepositoryTriggersExceededException

The number of triggers allowed for the repository was exceeded.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

RepositoryTriggerBranchNameListRequiredException

At least one branch name is required, but was not specified in the trigger configuration.

HTTP Status Code: 400

RepositoryTriggerDestinationArnRequiredException

A destination ARN for the target service for the trigger is required, but was not specified.

HTTP Status Code: 400

RepositoryTriggerEventsListRequiredException

At least one event for the trigger is required, but was not specified.

HTTP Status Code: 400

RepositoryTriggerNameRequiredException

A name for the trigger is required, but was not specified.

HTTP Status Code: 400

RepositoryTriggersListRequiredException

The list of triggers for the repository is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of TestRepositoryTriggers.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 33
X-Amz-Target: CodeCommit_20150413.TestRepositoryTriggers
X-Amz-Date: 20151028T230050Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
      "repositoryName": "MyDemoRepo",
      "triggers": [
        {
          "name": "MyFirstTrigger",
          "destinationArn": "arn:aws:sns:us-east-1:123456789012:MyCodeCommitTopic",
          "branches": [
            "main",
            "preprod"
          ],
          "events": [
            "all"
          ]
        }
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 107
Date: Wed, 28 Oct 2015 23:00:52 GMT

{
      "successfulExecutions": [
        "MyFirstTrigger"
      ],
      "failedExecutions": []
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UntagResource

Removes tags for a resource in AWS CodeCommit. For a list of valid resources in AWS CodeCommit, see CodeCommit Resources and Operations in the *AWS CodeCommit User Guide*.

Request Syntax

```
{
   "resourceArn": "string",
   "tagKeys": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

resourceArn

The Amazon Resource Name (ARN) of the resource to which you want to remove tags.

Type: String

Required: Yes

tagKeys

The tag key for each tag that you want to remove from the resource.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 128.

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors.

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

InvalidResourceArnException

The value for the resource ARN is not valid. For more information about resources in AWS CodeCommit, see CodeCommit Resources and Operations in the AWS CodeCommit User Guide.

HTTP Status Code: 400

InvalidSystemTagUsageException

The specified tag is not valid. Key names cannot be prefixed with aws:.

HTTP Status Code: 400

InvalidTagKeysListException

The list of tags is not valid.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

ResourceArnRequiredException

A valid Amazon Resource Name (ARN) for an AWS CodeCommit resource is required. For a list of valid resources in AWS CodeCommit, see CodeCommit Resources and Operations in the AWS CodeCommit User Guide.

HTTP Status Code: 400

TagKeysListRequiredException

A list of tag keys is required. The list cannot be empty or null.

HTTP Status Code: 400

TagPolicyException

The tag policy is not valid.

HTTP Status Code: 400

TooManyTagsException

The maximum number of tags for an AWS CodeCommit resource has been exceeded.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of UntagResource.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 2
X-Amz-Target: CodeCommit_20150413.UntagResource
X-Amz-Date: 20190228T224019Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151028/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "resourceArn": "arn:aws:codecommit:us-east-2:123456789012:MySecretProjectRepo"
  "tagKeys": [
    "Status"
  ]
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 188
Date: Thu, 28 Feb 2019 22:40:19 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateApprovalRuleTemplateContent

Updates the content of an approval rule template. You can change the number of required approvals, the membership of the approval rule, and whether an approval pool is defined.

Request Syntax

```
{
   "approvalRuleTemplateName": "string",
   "existingRuleContentSha256": "string",
   "newRuleContent": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

approvalRuleTemplateName

The name of the approval rule template where you want to update the content of the rule.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: Yes

existingRuleContentSha256

The SHA-256 hash signature for the content of the approval rule. You can retrieve this information by using GetPullRequest.

Type: String

Required: No

newRuleContent

The content that replaces the existing content of the rule. Content statements must be complete. You cannot provide only the changes.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 3000.

Required: Yes

Response Syntax

```
{
   "approvalRuleTemplate": {
      "approvalRuleTemplateContent": "string",
      "approvalRuleTemplateDescription": "string",
      "approvalRuleTemplateId": "string",
      "approvalRuleTemplateName": "string",
      "creationDate": number,
      "lastModifiedDate": number,
      "lastModifiedUser": "string",
      "ruleContentSha256": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

approvalRuleTemplate

Returns information about an approval rule template.

Type: ApprovalRuleTemplate object

Errors

For information about the errors that are common to all actions, see Common Errors.

ApprovalRuleTemplateContentRequiredException

The content for the approval rule template is empty. You must provide some content for an approval rule template. The content cannot be null.

HTTP Status Code: 400

ApprovalRuleTemplateDoesNotExistException

The specified approval rule template does not exist. Verify that the name is correct and that you are signed in to the AWS Region where the template was created, and then try again.

HTTP Status Code: 400

ApprovalRuleTemplateNameRequiredException

An approval rule template name is required, but was not specified.

HTTP Status Code: 400

InvalidApprovalRuleTemplateContentException

The content of the approval rule template is not valid.

HTTP Status Code: 400

InvalidApprovalRuleTemplateNameException

The name of the approval rule template is not valid. Template names must be between 1 and 100 valid characters in length. For more information about limits in AWS CodeCommit, see Quotas in the *AWS CodeCommit User Guide*.

HTTP Status Code: 400

InvalidRuleContentSha256Exception

The SHA-256 hash signature for the rule content is not valid.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of UpdateApprovalRuleTemplateContent.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.UpdateApprovalRuleTemplateContent
```

```
X-Amz-Date: 20191021T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
    "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
    "newRuleContent": "{\"Version\": \"2018-11-08\",\"DestinationReferences\": [\"refs/
heads/main\"],\"Statements\": [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,
\"ApprovalPoolMembers\": [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*
\"]}]}"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Mon, 21 Oct 2019 20:20:13 GMT

{
    "approvalRuleTemplate": {
        "creationDate": 1571352720.773,
        "approvalRuleTemplateDescription": "Requires 1 approval for all pull requests
 from the CodeCommitReview pool",
        "lastModifiedDate": 1571358728.41,
        "approvalRuleTemplateId": "41de97b7-EXAMPLE",
        "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\",\"Statements\":
 [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 1,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
        "ruleContentSha256": "2f6c21a5EXAMPLE",
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Li_Juan"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateApprovalRuleTemplateDescription

Updates the description for a specified approval rule template.

Request Syntax

```
{
   "approvalRuleTemplateDescription": "string",
   "approvalRuleTemplateName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

approvalRuleTemplateDescription

The updated description of the approval rule template.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1000.

Required: Yes

approvalRuleTemplateName

The name of the template for which you want to update the description.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: Yes

Response Syntax

```
{
   "approvalRuleTemplate": {
      "approvalRuleTemplateContent": "string",
      "approvalRuleTemplateDescription": "string",
      "approvalRuleTemplateId": "string",
      "approvalRuleTemplateName": "string",
      "creationDate": number,
      "lastModifiedDate": number,
      "lastModifiedUser": "string",
      "ruleContentSha256": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

approvalRuleTemplate

The structure and content of the updated approval rule template.

Type: ApprovalRuleTemplate object

Errors

For information about the errors that are common to all actions, see Common Errors.

ApprovalRuleTemplateDoesNotExistException

The specified approval rule template does not exist. Verify that the name is correct and that you are signed in to the AWS Region where the template was created, and then try again.

HTTP Status Code: 400

ApprovalRuleTemplateNameRequiredException

An approval rule template name is required, but was not specified.

HTTP Status Code: 400

InvalidApprovalRuleTemplateDescriptionException

The description for the approval rule template is not valid because it exceeds the maximum characters allowed for a description. For more information about limits in AWS CodeCommit, see Quotas in the *AWS CodeCommit User Guide*.

HTTP Status Code: 400

InvalidApprovalRuleTemplateNameException

The name of the approval rule template is not valid. Template names must be between 1 and 100 valid characters in length. For more information about limits in AWS CodeCommit, see Quotas in the *AWS CodeCommit User Guide*.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of UpdateApprovalRuleTemplateDescription.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.UpdateApprovalRuleTemplateDescription
X-Amz-Date: 20191021T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
   "approvalRuleTemplateDescription": "Requires 1 approval for all pull requests from
 the CodeCommitReview pool"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
```

```
Date: Mon, 21 Oct 2019 20:20:13 GMT

{
    "approvalRuleTemplate": {
        "creationDate": 1571352720.773,
        "approvalRuleTemplateDescription": "Requires 1 approval for all pull requests
 from the CodeCommitReview pool",
        "lastModifiedDate": 1571358728.41,
        "approvalRuleTemplateId": "41de97b7-EXAMPLE",
        "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\",\"Statements\":
 [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 1,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
        "ruleContentSha256": "2f6c21a5EXAMPLE",
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Li_Juan"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateApprovalRuleTemplateName

Updates the name of a specified approval rule template.

Request Syntax

```
{
   "newApprovalRuleTemplateName": "string",
   "oldApprovalRuleTemplateName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

newApprovalRuleTemplateName

The new name you want to apply to the approval rule template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: Yes

oldApprovalRuleTemplateName

The current name of the approval rule template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: Yes

Response Syntax

```
{
   "approvalRuleTemplate": { 
      "approvalRuleTemplateContent": "string",
      "approvalRuleTemplateDescription": "string",
      "approvalRuleTemplateId": "string",
      "approvalRuleTemplateName": "string",
      "creationDate": number,
      "lastModifiedDate": number,
      "lastModifiedUser": "string",
      "ruleContentSha256": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

approvalRuleTemplate

The structure and content of the updated approval rule template.

Type: ApprovalRuleTemplate object

Errors

For information about the errors that are common to all actions, see Common Errors.

ApprovalRuleTemplateDoesNotExistException

The specified approval rule template does not exist. Verify that the name is correct and that you are signed in to the AWS Region where the template was created, and then try again.

HTTP Status Code: 400

ApprovalRuleTemplateNameAlreadyExistsException

You cannot create an approval rule template with that name because a template with that name already exists in this AWS Region for your AWS account. Approval rule template names must be unique.

HTTP Status Code: 400

ApprovalRuleTemplateNameRequiredException

An approval rule template name is required, but was not specified.

HTTP Status Code: 400

InvalidApprovalRuleTemplateNameException

The name of the approval rule template is not valid. Template names must be between 1 and 100 valid characters in length. For more information about limits in AWS CodeCommit, see Quotas in the *AWS CodeCommit User Guide*.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of UpdateApprovalRuleTemplateName.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.UpdateApprovalRuleTemplateName
X-Amz-Date: 20191021T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/10
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "oldApprovalRuleTemplateName": "1-approver-rule",
   "newApprovalRuleTemplateName": "1-approver-rule-for-all-pull-requests"
}
```

Sample Response

```
HTTP/1.1 200 OK
```

```
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Mon, 21 Oct 2019 20:20:13 GMT

{
    "approvalRuleTemplate": {
        "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
        "lastModifiedDate": 1571358241.619,
        "approvalRuleTemplateId": "41de97b7-EXAMPLE",
        "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\",\"Statements\":
 [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 1,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "creationDate": 1571352720.773,
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "approvalRuleTemplateDescription": "All pull requests must be approved by one
 developer on the team.",
        "ruleContentSha256": "2f6c21a5cEXAMPLE"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateComment

Replaces the contents of a comment.

Request Syntax

```
{
   "commentId": "string",
   "content": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

commentId

The system-generated ID of the comment you want to update. To get this ID, use GetCommentsForComparedCommit or GetCommentsForPullRequest.

Type: String

Required: Yes

content

The updated content to replace the existing content of the comment.

Type: String

Required: Yes

Response Syntax

```
{
   "comment": {
      "authorArn": "string",
      "callerReactions": [ "string" ],
      "clientRequestToken": "string",
      "commentId": "string",
      "content": "string",
      "creationDate": number,
      "deleted": boolean,
      "inReplyTo": "string",
      "lastModifiedDate": number,
      "reactionCounts": { 
         "string" : number 
      }
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

comment

Information about the updated comment.

Type: Comment object

Errors

For information about the errors that are common to all actions, see Common Errors.

CommentContentRequiredException

The comment is empty. You must provide some content for a comment. The content cannot be null.

HTTP Status Code: 400

CommentContentSizeLimitExceededException

The comment is too large. Comments are limited to 10,240 characters.

HTTP Status Code: 400

CommentDeletedException

This comment has already been deleted. You cannot edit or delete a deleted comment.

HTTP Status Code: 400

CommentDoesNotExistException

No comment exists with the provided ID. Verify that you have used the correct ID, and then try again.

HTTP Status Code: 400

CommentIdRequiredException

The comment ID is missing or null. A comment ID is required.

HTTP Status Code: 400

CommentNotCreatedByCallerException

You cannot modify or delete this comment. Only comment authors can modify or delete their comments.

HTTP Status Code: 400

InvalidCommentIdException

The comment ID is not in a valid format. Make sure that you have provided the full comment ID.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of UpdateComment.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.UpdateComment
X-Amz-Date: 20171025T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
```

```
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "commentId": "442b498bEXAMPLE5756813",
   "content": "Fixed as requested. I'll update the pull request."
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Wed, 25 Oct 2017 20:20:13 GMT

{
    "comment": {
        "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
        "clientRequestToken": "",
        "commentId": "442b498bEXAMPLE5756813",
        "content": "Fixed as requested. I'll update the pull request.",
        "creationDate": 1508369929.783,
        "deleted": false,
        "lastModifiedDate": 1508369929.287,
        "callerReactions": [],
        "reactionCounts":
          {
            "THUMBSUP" : 2
          }
    }
 }
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4

- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateDefaultBranch

Sets or changes the default branch name for the specified repository.

> **Note**
>
> If you use this operation to change the default branch name to the current default branch name, a success message is returned even though the default branch did not change.

Request Syntax

```
{
   "defaultBranchName": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

defaultBranchName

The name of the branch to set as the default branch.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: Yes

repositoryName

The name of the repository for which you want to set or change the default branch.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors.

BranchDoesNotExistException

The specified branch does not exist.

HTTP Status Code: 400

BranchNameRequiredException

A branch name is required, but was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidBranchNameException

The specified reference name is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of UpdateDefaultBranch.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 71
X-Amz-Target: CodeCommit_20150413.UpdateDefaultBranch
X-Amz-Date: 20151029T151143Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151029/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "defaultBranchName": "MyNewBranch",
  "repositoryName": "MyDemoRepo"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 0
Date: Thu, 29 Oct 2015 15:11:44 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin

- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdatePullRequestApprovalRuleContent

Updates the structure of an approval rule created specifically for a pull request. For example, you can change the number of required approvers and the approval pool for approvers.

Request Syntax

```
{
   "approvalRuleName": "string",
   "existingRuleContentSha256": "string",
   "newRuleContent": "string",
   "pullRequestId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

approvalRuleName

The name of the approval rule you want to update.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: Yes

existingRuleContentSha256

The SHA-256 hash signature for the content of the approval rule. You can retrieve this information by using GetPullRequest.

Type: String

Required: No

newRuleContent

The updated content for the approval rule.

Note

When you update the content of the approval rule, you can specify approvers in an approval pool in one of two ways:

- **CodeCommitApprovers**: This option only requires an AWS account and a resource. It can be used for both IAM users and federated access users whose name matches the provided resource name. This is a very powerful option that offers a great deal of flexibility. For example, if you specify the AWS account *123456789012* and *Mary_Major*, all of the following are counted as approvals coming from that user:
  - An IAM user in the account (arn:aws:iam::*123456789012*:user/*Mary_Major*)
  - A federated user identified in IAM as Mary_Major (arn:aws:sts::*123456789012*:federated-user/*Mary_Major*)

  This option does not recognize an active session of someone assuming the role of CodeCommitReview with a role session name of *Mary_Major* (arn:aws:sts::*123456789012*:assumed-role/CodeCommitReview/*Mary_Major*) unless you include a wildcard (*Mary_Major).
- **Fully qualified ARN**: This option allows you to specify the fully qualified Amazon Resource Name (ARN) of the IAM user or role.

For more information about IAM ARNs, wildcards, and formats, see IAM Identifiers in the *IAM User Guide*.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 3000.

Required: Yes

pullRequestId

The system-generated ID of the pull request.

Type: String

Required: Yes

Response Syntax

```
{
   "approvalRule": {
      "approvalRuleContent": "string",
      "approvalRuleId": "string",
      "approvalRuleName": "string",
      "creationDate": number,
      "lastModifiedDate": number,
      "lastModifiedUser": "string",
      "originApprovalRuleTemplate": {
         "approvalRuleTemplateId": "string",
         "approvalRuleTemplateName": "string"
      },
      "ruleContentSha256": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

approvalRule

Information about the updated approval rule.

Type: ApprovalRule object

Errors

For information about the errors that are common to all actions, see Common Errors.

ApprovalRuleContentRequiredException

The content for the approval rule is empty. You must provide some content for an approval rule. The content cannot be null.

HTTP Status Code: 400

ApprovalRuleDoesNotExistException

The specified approval rule does not exist.

HTTP Status Code: 400

ApprovalRuleNameRequiredException

An approval rule name is required, but was not specified.

HTTP Status Code: 400

CannotModifyApprovalRuleFromTemplateException

The approval rule cannot be modified for the pull request because it was created by an approval rule template and applied to the pull request automatically.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidApprovalRuleContentException

The content for the approval rule is not valid.

HTTP Status Code: 400

InvalidApprovalRuleNameException

The name for the approval rule is not valid.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

InvalidRuleContentSha256Exception

The SHA-256 hash signature for the rule content is not valid.

HTTP Status Code: 400

PullRequestAlreadyClosedException

The pull request status cannot be updated because it is already closed.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of UpdatePullRequestApprovalRuleContent.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.UpdatePullRequestApprovalRuleContent
X-Amz-Date: 20191025T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "pullRequestId": "27",
   "approvalRuleName": "Require two approvers from approval rule",
   "approvalRuleContent": "{Version: 2018-11-08, Statements: [{Type:
 \"Approvers\", NumberOfApprovalsNeeded: 2, ApprovalPoolMembers:
[\"CodeCommitApprovers:123456789012:user/Jorge_Souza",
 \"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*"]}]}}"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Fri, 25 Oct 2019 20:20:13 GMT

{
    "approvalRule": {
        "approvalRuleName": "Require two approvers from approval pool",
        "lastModifiedDate": 1570752871.932,
        "ruleContentSha256": "7c44e6ebEXAMPLE",
        "creationDate": 1570752871.932,
         "approvalRuleId": "bal37823-EXAMPLE",
         "originApprovalRuleTemplate": {},
         "approvalRuleContent": "{Version: 2018-11-08, Statements:
 [{Type: \"Approvers\", NumberOfApprovalsNeeded: 2, ApprovalPoolMembers:
[\"CodeCommitApprovers:123456789012:user/Jorge_Souza",
 \"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*"]}]}}",
         "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major"
     }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdatePullRequestApprovalState

Updates the state of a user's approval on a pull request. The user is derived from the signed-in account when the request is made.

Request Syntax

```
{
   "approvalState": "string",
   "pullRequestId": "string",
   "revisionId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

approvalState

The approval state to associate with the user on the pull request.

Type: String

Valid Values: APPROVE | REVOKE

Required: Yes

pullRequestId

The system-generated ID of the pull request.

Type: String

Required: Yes

revisionId

The system-generated ID of the revision.

Type: String

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors.

ApprovalStateRequiredException

An approval state is required, but was not specified.

HTTP Status Code: 400

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidApprovalStateException

The state for the approval is not valid. Valid values include APPROVE and REVOKE.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

InvalidRevisionIdException

The revision ID is not valid. Use GetPullRequest to determine the value.

HTTP Status Code: 400

MaximumNumberOfApprovalsExceededException

The number of approvals required for the approval rule exceeds the maximum number allowed.

HTTP Status Code: 400

PullRequestAlreadyClosedException

The pull request status cannot be updated because it is already closed.

HTTP Status Code: 400

PullRequestCannotBeApprovedByAuthorException

The approval cannot be applied because the user approving the pull request matches the user who created the pull request. You cannot approve a pull request that you created.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

RevisionIdRequiredException

A revision ID is required, but was not provided.

HTTP Status Code: 400

RevisionNotCurrentException

The revision ID provided in the request does not match the current revision ID. Use GetPullRequest to retrieve the current revision ID.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of UpdatePullRequestApprovalState.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.UpdatePullRequestApprovalRuleState
X-Amz-Date: 20191025T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "pullRequestId": "27",
   "revisionId": "9f29d167EXAMPLE",
   "approvalState": "APPROVE"
}
```

Sample Response

```
HTTP/1.1 200 OK
```

```
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Fri, 25 Oct 2019 20:20:13 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdatePullRequestDescription

Replaces the contents of the description of a pull request.

Request Syntax

```
{
   "description": "string",
   "pullRequestId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

description

The updated content of the description for the pull request. This content replaces the existing description.

Type: String

Length Constraints: Maximum length of 10240.

Required: Yes

pullRequestId

The system-generated ID of the pull request. To get this ID, use ListPullRequests.

Type: String

Required: Yes

Response Syntax

```
{
   "pullRequest": {
      "approvalRules": [
         {
            "approvalRuleContent": "string",
            "approvalRuleId": "string",
            "approvalRuleName": "string",
            "creationDate": number,
            "lastModifiedDate": number,
            "lastModifiedUser": "string",
            "originApprovalRuleTemplate": {
               "approvalRuleTemplateId": "string",
               "approvalRuleTemplateName": "string"
            },
            "ruleContentSha256": "string"
         }
      ],
      "authorArn": "string",
      "clientRequestToken": "string",
      "creationDate": number,
      "description": "string",
      "lastActivityDate": number,
      "pullRequestId": "string",
      "pullRequestStatus": "string",
      "pullRequestTargets": [
         {
            "destinationCommit": "string",
            "destinationReference": "string",
            "mergeBase": "string",
            "mergeMetadata": {
               "isMerged": boolean,
               "mergeCommitId": "string",
               "mergedBy": "string",
               "mergeOption": "string"
            },
            "repositoryName": "string",
            "sourceCommit": "string",
            "sourceReference": "string"
         }
      ],
      "revisionId": "string",
      "title": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

pullRequest

Information about the updated pull request.

Type: PullRequest object

Errors

For information about the errors that are common to all actions, see Common Errors.

InvalidDescriptionException

The pull request description is not valid. Descriptions cannot be more than 1,000 characters.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

PullRequestAlreadyClosedException

The pull request status cannot be updated because it is already closed.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of UpdatePullRequestDescription.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.UpdatePullRequestDescription
X-Amz-Date: 20171025T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "description": "Updated the pull request to remove unused global variable.",
   "pullRequestId": "47"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Wed, 25 Oct 2017 20:20:13 GMT

{
    "pullRequest": {
        "approvalRules": [
            {
                "approvalRuleContent": "{\"Version\": \"2018-11-08\",
\"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
                "approvalRuleId": "dd8b17fe-EXAMPLE",
                "approvalRuleName": "2-approver-rule-for-main",
                "creationDate": 1571356106.936,
                "lastModifiedDate": 571356106.936,
                "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
                "originApprovalRuleTemplate": {
                    "approvalRuleTemplateId": "dd8b17fe-OTHEREXAMPLE",
                    "approvalRuleTemplateName": "2-approver-rule-for-main"
                },
                "ruleContentSha256": "4711b576EXAMPLE"
            }
        ],
        "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
        "clientRequestToken": "",
        "creationDate": 1508530823.155,
        "description": "Updated the pull request to remove unused global variable.",
        "lastActivityDate": 1508372423.204,
        "pullRequestId": "47",
        "pullRequestStatus": "OPEN",
        "pullRequestTargets": [
            {
                "destinationCommit": "9f31c968EXAMPLE",
                "destinationReference": "refs/heads/main",
                "mergeMetadata": {
                    "isMerged": false,
                },
                "repositoryName": "MyDemoRepo",
                "sourceCommit": "99132ab0EXAMPLE",
                "sourceReference": "refs/heads/variables-branch"
            }
        ],
        "title": "Consolidation of global variables"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2

- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdatePullRequestStatus

Updates the status of a pull request.

Request Syntax

```
{
   "pullRequestId": "string",
   "pullRequestStatus": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

pullRequestId

The system-generated ID of the pull request. To get this ID, use ListPullRequests.

Type: String

Required: Yes

pullRequestStatus

The status of the pull request. The only valid operations are to update the status from `OPEN` to `OPEN`, `OPEN` to `CLOSED` or from `CLOSED` to `CLOSED`.

Type: String

Valid Values: `OPEN | CLOSED`

Required: Yes

Response Syntax

```
{
   "pullRequest": {
      "approvalRules": [
         {
            "approvalRuleContent": "string",
            "approvalRuleId": "string",
            "approvalRuleName": "string",
            "creationDate": number,
            "lastModifiedDate": number,
            "lastModifiedUser": "string",
            "originApprovalRuleTemplate": {
               "approvalRuleTemplateId": "string",
               "approvalRuleTemplateName": "string"
            },
            "ruleContentSha256": "string"
         }
      ],
      "authorArn": "string",
      "clientRequestToken": "string",
      "creationDate": number,
      "description": "string",
      "lastActivityDate": number,
      "pullRequestId": "string",
      "pullRequestStatus": "string",
      "pullRequestTargets": [
         {
            "destinationCommit": "string",
            "destinationReference": "string",
            "mergeBase": "string",
            "mergeMetadata": {
               "isMerged": boolean,
               "mergeCommitId": "string",
               "mergedBy": "string",
               "mergeOption": "string"
            },
            "repositoryName": "string",
            "sourceCommit": "string",
            "sourceReference": "string"
         }
      ],
      "revisionId": "string",
      "title": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

pullRequest

Information about the pull request.

Type: PullRequest object

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

InvalidPullRequestStatusException

The pull request status is not valid. The only valid values are `OPEN` and `CLOSED`.

HTTP Status Code: 400

InvalidPullRequestStatusUpdateException

The pull request status update is not valid. The only valid update is from `OPEN` to `CLOSED`.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

PullRequestStatusRequiredException

A pull request status is required, but none was provided.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of UpdatePullRequestStatus.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
```

```
X-Amz-Target: CodeCommit_20150413.UpdatePullRequestStatus
X-Amz-Date: 20171025T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "pullRequestStatus": "CLOSED",
   "pullRequestId": "47"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Wed, 25 Oct 2017 20:20:13 GMT

{
    "pullRequest": {
        "approvalRules": [
            {
                "approvalRuleContent": "{\"Version\": \"2018-11-08\",\"Statements\":
 [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
                "approvalRuleId": "dd8b17fe-EXAMPLE",
                "approvalRuleName": "2-approvers-needed-for-this-change",
                "creationDate": 1571356106.936,
                "lastModifiedDate": 571356106.936,
                "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
                "ruleContentSha256": "4711b576EXAMPLE"
            }
        ],
        "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
        "clientRequestToken": "",
        "creationDate": 1508530823.165,
        "description": "Updated the pull request to remove unused global variable.",
        "lastActivityDate": 1508372423.12,
        "pullRequestId": "47",
        "pullRequestStatus": "CLOSED",
        "pullRequestTargets": [
            {
                "destinationCommit": "9f31c968EXAMPLE",
                "destinationReference": "refs/heads/main",
                "mergeMetadata": {
                    "isMerged": false,
                },
                "repositoryName": "MyDemoRepo",
                "sourceCommit": "99132ab0EXAMPLE",
                "sourceReference": "refs/heads/variables-branch"
            }
        ],
        "title": "Consolidation of global variables"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdatePullRequestTitle

Replaces the title of a pull request.

Request Syntax

```
{
   "pullRequestId": "string",
   "title": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

pullRequestId

The system-generated ID of the pull request. To get this ID, use ListPullRequests.

Type: String

Required: Yes

title

The updated title of the pull request. This replaces the existing title.

Type: String

Length Constraints: Maximum length of 150.

Required: Yes

Response Syntax

```
{
   "pullRequest": {
      "approvalRules": [
         {
            "approvalRuleContent": "string",
            "approvalRuleId": "string",
            "approvalRuleName": "string",
            "creationDate": number,
            "lastModifiedDate": number,
            "lastModifiedUser": "string",
            "originApprovalRuleTemplate": {
               "approvalRuleTemplateId": "string",
               "approvalRuleTemplateName": "string"
            },
            "ruleContentSha256": "string"
         }
      ],
      "authorArn": "string",
      "clientRequestToken": "string",
      "creationDate": number,
      "description": "string",
      "lastActivityDate": number,
      "pullRequestId": "string",
      "pullRequestStatus": "string",
      "pullRequestTargets": [
         {
            "destinationCommit": "string",
            "destinationReference": "string",
            "mergeBase": "string",
            "mergeMetadata": {
               "isMerged": boolean,
               "mergeCommitId": "string",
               "mergedBy": "string",
               "mergeOption": "string"
            },
            "repositoryName": "string",
            "sourceCommit": "string",
            "sourceReference": "string"
         }
      ],
      "revisionId": "string",
      "title": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

pullRequest

Information about the updated pull request.

Type: PullRequest object

Errors

For information about the errors that are common to all actions, see Common Errors.

InvalidPullRequestIdException

The pull request ID is not valid. Make sure that you have provided the full ID and that the pull request is in the specified repository, and then try again.

HTTP Status Code: 400

InvalidTitleException

The title of the pull request is not valid. Pull request titles cannot exceed 100 characters in length.

HTTP Status Code: 400

PullRequestAlreadyClosedException

The pull request status cannot be updated because it is already closed.

HTTP Status Code: 400

PullRequestDoesNotExistException

The pull request ID could not be found. Make sure that you have specified the correct repository name and pull request ID, and then try again.

HTTP Status Code: 400

PullRequestIdRequiredException

A pull request ID is required, but none was provided.

HTTP Status Code: 400

TitleRequiredException

A pull request title is required. It cannot be empty or null.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of UpdatePullRequestTitle.

Sample Request

```
>POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 350
X-Amz-Target: CodeCommit_20150413.UpdatePullRequestTitle
X-Amz-Date: 20171025T132023Z
User-Agent: aws-cli/1.11.187 Python/2.7.9 Windows/8
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20171025/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
   "description": "Consolidation of global variables - updated review",
   "pullRequestId": "47"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
Date: Wed, 25 Oct 2017 20:20:13 GMT

{
    "pullRequest": {
        "approvalRules": [
            {
                "approvalRuleContent": "{\"Version\": \"2018-11-08\",
\"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
                "approvalRuleId": "dd8b17fe-EXAMPLE",
                "approvalRuleName": "2-approver-rule-for-main",
                "creationDate": 1571356106.936,
                "lastModifiedDate": 571356106.936,
                "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
                "originApprovalRuleTemplate": {
                    "approvalRuleTemplateId": "dd8b17fe-OTHEREXAMPLE",
                    "approvalRuleTemplateName": "2-approver-rule-for-main"
                },
                "ruleContentSha256": "4711b576EXAMPLE"
            }
        ],
        "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
        "clientRequestToken": "",
        "creationDate": 1508530823.12,
        "description": "Review the latest changes and updates to the global variables.
 I have updated this request with some changes, including removing some unused
 variables.",
        "lastActivityDate": 1508372657.188,
        "pullRequestId": "47",
        "pullRequestStatus": "OPEN",
        "pullRequestTargets": [
            {
                "destinationCommit": "9f31c968EXAMPLE",
                "destinationReference": "refs/heads/main",
                "mergeMetadata": {
                    "isMerged": false,
                },
                "repositoryName": "MyDemoRepo",
                "sourceCommit": "99132ab0EXAMPLE",
                "sourceReference": "refs/heads/variables-branch"
            }
        ],
        "title": "Consolidation of global variables - updated review"
    }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateRepositoryDescription

Sets or changes the comment or description for a repository.

> **Note**
>
> The description field for a repository accepts all HTML characters and all valid Unicode characters. Applications that do not HTML-encode the description and display it in a webpage can expose users to potentially malicious code. Make sure that you HTML-encode the description field in any application that uses this API to display the repository description on a webpage.

Request Syntax

```
{
   "repositoryDescription": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

repositoryDescription

The new comment or description for the specified repository. Repository descriptions are limited to 1,000 characters.

Type: String

Length Constraints: Maximum length of 1000.

Required: No

repositoryName

The name of the repository to set or change the comment or description for.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidRepositoryDescriptionException

The specified repository description is not valid.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of UpdateRepositoryDescription.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 90
```

```
X-Amz-Target: CodeCommit_20150413.UpdateRepositoryDescription
X-Amz-Date: 20151029T153247Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151029/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "repositoryName": "MyDemoRepo",
  "repositoryDescription": "This description was changed"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 0
Date: Thu, 29 Oct 2015 15:32:49 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateRepositoryEncryptionKey

Updates the AWS Key Management Service encryption key used to encrypt and decrypt a CodeCommit repository.

Request Syntax

```
{
   "kmsKeyId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

kmsKeyId

The ID of the encryption key. You can view the ID of an encryption key in the AWS KMS console, or use the AWS KMS APIs to programmatically retrieve a key ID. For more information about acceptable values for keyID, see KeyId in the Decrypt API description in the *AWS Key Management Service API Reference*.

Type: String

Pattern: `^[a-zA-Z0-9:/_-]+$`

Required: Yes

repositoryName

The name of the repository for which you want to update the AWS KMS encryption key used to encrypt and decrypt the repository.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Syntax

```
{
   "kmsKeyId": "string",
   "originalKmsKeyId": "string",
   "repositoryId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

kmsKeyId

The ID of the encryption key.

Type: String

Pattern: `^[a-zA-Z0-9:/_-]+$`

originalKmsKeyId

The ID of the encryption key formerly used to encrypt and decrypt the repository.

Type: String

Pattern: `^[a-zA-Z0-9:/_-]+$`

repositoryId

The ID of the repository.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors.

EncryptionIntegrityChecksFailedException

An encryption integrity check failed.

HTTP Status Code: 500

EncryptionKeyAccessDeniedException

An encryption key could not be accessed.

HTTP Status Code: 400

EncryptionKeyDisabledException

The encryption key is disabled.

HTTP Status Code: 400

EncryptionKeyInvalidIdException

The AWS Key Management Service encryption key is not valid.

HTTP Status Code: 400

EncryptionKeyInvalidUsageException

A AWS KMS encryption key was used to try and encrypt or decrypt a repository, but either the repository or the key was not in a valid state to support the operation.

HTTP Status Code: 400

EncryptionKeyNotFoundException

No encryption key was found.

HTTP Status Code: 400

EncryptionKeyRequiredException

A AWS KMS encryption key ID is required but was not specified.

HTTP Status Code: 400

EncryptionKeyUnavailableException

The encryption key is not available.

HTTP Status Code: 400

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

The following example updates a repository named *MyDemoRepo* to use a AWS KMS key with the ID of *a1b2c3d4-5678-90ab-cdef-EXAMPLE11111*.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 79
X-Amz-Target: CodeCommit_20150413.UpdateRepositoryEncryptionKey
X-Amz-Date: 20231221T153512Z
User-Agent: aws-cli/1.32.0 Python/3.6.0 Windows/10
Content-Type: application/x-amz-json-1.1
```

```
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151029/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "kmsKeyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "repositoryName": "MyDemoRepo"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 0
Date: Thu, 21 Dec 2023 15:35:13 GMT

 {
  "kmsKeyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "originalKmsKeyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa",
  "repositoryName": "MyDemoRepo"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python

- AWS SDK for Ruby V3

UpdateRepositoryName

Renames a repository. The repository name must be unique across the calling AWS account. Repository names are limited to 100 alphanumeric, dash, and underscore characters, and cannot include certain characters. The suffix .git is prohibited. For more information about the limits on repository names, see Quotas in the AWS CodeCommit User Guide.

Request Syntax

```
{
   "newName": "string",
   "oldName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

newName

The new name for the repository.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

oldName

The current name of the repository.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors.

InvalidRepositoryNameException

A specified repository name is not valid.

> **Note**
>
> This exception occurs only when a specified repository name is not valid. Other exceptions occur when a required repository parameter is missing, or when a specified repository does not exist.

HTTP Status Code: 400

RepositoryDoesNotExistException

The specified repository does not exist.

HTTP Status Code: 400

RepositoryNameExistsException

The specified repository name already exists.

HTTP Status Code: 400

RepositoryNameRequiredException

A repository name is required, but was not specified.

HTTP Status Code: 400

Examples

Example

This example illustrates one usage of UpdateRepositoryName.

Sample Request

```
POST / HTTP/1.1
Host: codecommit.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 62
X-Amz-Target: CodeCommit_20150413.UpdateRepositoryName
X-Amz-Date: 20151029T153512Z
User-Agent: aws-cli/1.7.38 Python/2.7.9 Windows/7
Content-Type: application/x-amz-json-1.1
Authorization: AWS4-HMAC-SHA256 Credential=AKIAI44QH8DHBEXAMPLE/20151029/us-east-1/
codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-
target, Signature=8d9b5998EXAMPLE

{
  "newName": "MyRenamedDemoRepo",
  "oldName": "MyDemoRepo"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 0
Date: Thu, 29 Oct 2015 15:35:13 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin

- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Data Types

The AWS CodeCommit API contains several data types that various actions use. This section describes each data type in detail.

Note

The order of each element in a data type structure is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

- Approval
- ApprovalRule
- ApprovalRuleEventMetadata
- ApprovalRuleOverriddenEventMetadata
- ApprovalRuleTemplate
- ApprovalStateChangedEventMetadata
- BatchAssociateApprovalRuleTemplateWithRepositoriesError
- BatchDescribeMergeConflictsError
- BatchDisassociateApprovalRuleTemplateFromRepositoriesError
- BatchGetCommitsError
- BatchGetRepositoriesError
- BlobMetadata
- BranchInfo
- Comment
- CommentsForComparedCommit
- CommentsForPullRequest
- Commit
- Conflict
- ConflictMetadata
- ConflictResolution

- DeleteFileEntry
- Difference
- Evaluation
- File
- FileMetadata
- FileModes
- FileSizes
- FileVersion
- Folder
- IsBinaryFile
- Location
- MergeHunk
- MergeHunkDetail
- MergeMetadata
- MergeOperations
- ObjectTypes
- OriginApprovalRuleTemplate
- PullRequest
- PullRequestCreatedEventMetadata
- PullRequestEvent
- PullRequestMergedStateChangedEventMetadata
- PullRequestSourceReferenceUpdatedEventMetadata
- PullRequestStatusChangedEventMetadata
- PullRequestTarget
- PutFileEntry
- ReactionForComment
- ReactionValueFormats
- ReplaceContentEntry
- RepositoryMetadata
- RepositoryNameIdPair

- RepositoryTrigger
- RepositoryTriggerExecutionFailure
- SetFileModeEntry
- SourceFileSpecifier
- SubModule
- SymbolicLink
- Target
- UserInfo

Approval

Returns information about a specific approval on a pull request.

Contents

approvalState

The state of the approval, APPROVE or REVOKE. REVOKE states are not stored.

Type: String

Valid Values: `APPROVE | REVOKE`

Required: No

userArn

The Amazon Resource Name (ARN) of the user.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ApprovalRule

Returns information about an approval rule.

Contents

approvalRuleContent

The content of the approval rule.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 3000.

Required: No

approvalRuleId

The system-generated ID of the approval rule.

Type: String

Required: No

approvalRuleName

The name of the approval rule.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: No

creationDate

The date the approval rule was created, in timestamp format.

Type: Timestamp

Required: No

lastModifiedDate

The date the approval rule was most recently changed, in timestamp format.

Type: Timestamp

Required: No

lastModifiedUser

The Amazon Resource Name (ARN) of the user who made the most recent changes to the approval rule.

Type: String

Required: No

originApprovalRuleTemplate

The approval rule template used to create the rule.

Type: OriginApprovalRuleTemplate object

Required: No

ruleContentSha256

The SHA-256 hash signature for the content of the approval rule.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ApprovalRuleEventMetadata

Returns information about an event for an approval rule.

Contents

approvalRuleContent

The content of the approval rule.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 3000.

Required: No

approvalRuleId

The system-generated ID of the approval rule.

Type: String

Required: No

approvalRuleName

The name of the approval rule.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ApprovalRuleOverriddenEventMetadata

Returns information about an override event for approval rules for a pull request.

Contents

overrideStatus

The status of the override event.

Type: String

Valid Values: `OVERRIDE | REVOKE`

Required: No

revisionId

The revision ID of the pull request when the override event occurred.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ApprovalRuleTemplate

Returns information about an approval rule template.

Contents

approvalRuleTemplateContent

The content of the approval rule template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 3000.

Required: No

approvalRuleTemplateDescription

The description of the approval rule template.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1000.

Required: No

approvalRuleTemplateId

The system-generated ID of the approval rule template.

Type: String

Required: No

approvalRuleTemplateName

The name of the approval rule template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: No

creationDate

The date the approval rule template was created, in timestamp format.

Type: Timestamp

Required: No

lastModifiedDate

The date the approval rule template was most recently changed, in timestamp format.

Type: Timestamp

Required: No

lastModifiedUser

The Amazon Resource Name (ARN) of the user who made the most recent changes to the approval rule template.

Type: String

Required: No

ruleContentSha256

The SHA-256 hash signature for the content of the approval rule template.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ApprovalStateChangedEventMetadata

Returns information about a change in the approval state for a pull request.

Contents

approvalStatus

The approval status for the pull request.

Type: String

Valid Values: `APPROVE | REVOKE`

Required: No

revisionId

The revision ID of the pull request when the approval state changed.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

BatchAssociateApprovalRuleTemplateWithRepositoriesError

Returns information about errors in a BatchAssociateApprovalRuleTemplateWithRepositories operation.

Contents

errorCode

An error code that specifies whether the repository name was not valid or not found.

Type: String

Required: No

errorMessage

An error message that provides details about why the repository name was not found or not valid.

Type: String

Required: No

repositoryName

The name of the repository where the association was not made.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++

- AWS SDK for Java V2
- AWS SDK for Ruby V3

BatchDescribeMergeConflictsError

Returns information about errors in a BatchDescribeMergeConflicts operation.

Contents

exceptionName

The name of the exception.

Type: String

Required: Yes

filePath

The path to the file.

Type: String

Required: Yes

message

The message provided by the exception.

Type: String

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

BatchDisassociateApprovalRuleTemplateFromRepositoriesError

Returns information about errors in a BatchDisassociateApprovalRuleTemplateFromRepositories operation.

Contents

errorCode

An error code that specifies whether the repository name was not valid or not found.

Type: String

Required: No

errorMessage

An error message that provides details about why the repository name was either not found or not valid.

Type: String

Required: No

repositoryName

The name of the repository where the association with the template was not able to be removed.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++

- AWS SDK for Java V2
- AWS SDK for Ruby V3

BatchGetCommitsError

Returns information about errors in a BatchGetCommits operation.

Contents

commitId

A commit ID that either could not be found or was not in a valid format.

Type: String

Required: No

errorCode

An error code that specifies whether the commit ID was not valid or not found.

Type: String

Required: No

errorMessage

An error message that provides detail about why the commit ID either was not found or was not valid.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

BatchGetRepositoriesError

Returns information about errors in a BatchGetRepositories operation.

Contents

errorCode

An error code that specifies the type of failure.

Type: String

Valid Values: `EncryptionIntegrityChecksFailedException | EncryptionKeyAccessDeniedException | EncryptionKeyDisabledException | EncryptionKeyNotFoundException | EncryptionKeyUnavailableException | RepositoryDoesNotExistException`

Required: No

errorMessage

An error message that provides detail about why the repository either was not found or was not in a valid state.

Type: String

Required: No

repositoryId

The ID of a repository that either could not be found or was not in a valid state.

Type: String

Required: No

repositoryName

The name of a repository that either could not be found or was not in a valid state.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

BlobMetadata

Returns information about a specific Git blob object.

Contents

blobId

The full ID of the blob.

Type: String

Required: No

mode

The file mode permissions of the blob. File mode permission codes include:

- `100644` indicates read/write
- `100755` indicates read/write/execute
- `160000` indicates a submodule
- `120000` indicates a symlink

Type: String

Required: No

path

The path to the blob and associated file name, if any.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2

- AWS SDK for Ruby V3

BranchInfo

Returns information about a branch.

Contents

branchName

The name of the branch.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

commitId

The ID of the last commit made to the branch.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Comment

Returns information about a specific comment.

Contents

authorArn

The Amazon Resource Name (ARN) of the person who posted the comment.

Type: String

Required: No

callerReactions

The emoji reactions to a comment, if any, submitted by the user whose credentials are associated with the call to the API.

Type: Array of strings

Required: No

clientRequestToken

A unique, client-generated idempotency token that, when provided in a request, ensures the request cannot be repeated with a changed parameter. If a request is received with the same parameters and a token is included, the request returns information about the initial request that used that token.

Type: String

Required: No

commentId

The system-generated comment ID.

Type: String

Required: No

content

The content of the comment.

Type: String

Required: No

creationDate

The date and time the comment was created, in timestamp format.

Type: Timestamp

Required: No

deleted

A Boolean value indicating whether the comment has been deleted.

Type: Boolean

Required: No

inReplyTo

The ID of the comment for which this comment is a reply, if any.

Type: String

Required: No

lastModifiedDate

The date and time the comment was most recently modified, in timestamp format.

Type: Timestamp

Required: No

reactionCounts

A string to integer map that represents the number of individual users who have responded to a comment with the specified reactions.

Type: String to integer map

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

CommentsForComparedCommit

Returns information about comments on the comparison between two commits.

Contents

afterBlobId

The full blob ID of the commit used to establish the after of the comparison.

Type: String

Required: No

afterCommitId

The full commit ID of the commit used to establish the after of the comparison.

Type: String

Required: No

beforeBlobId

The full blob ID of the commit used to establish the before of the comparison.

Type: String

Required: No

beforeCommitId

The full commit ID of the commit used to establish the before of the comparison.

Type: String

Required: No

comments

An array of comment objects. Each comment object contains information about a comment on the comparison between commits.

Type: Array of Comment objects

Required: No

location

Location information about the comment on the comparison, including the file name, line number, and whether the version of the file where the comment was made is BEFORE or AFTER.

Type: Location object

Required: No

repositoryName

The name of the repository that contains the compared commits.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

CommentsForPullRequest

Returns information about comments on a pull request.

Contents

afterBlobId

The full blob ID of the file on which you want to comment on the source commit.

Type: String

Required: No

afterCommitId

The full commit ID of the commit that was the tip of the source branch at the time the comment was made.

Type: String

Required: No

beforeBlobId

The full blob ID of the file on which you want to comment on the destination commit.

Type: String

Required: No

beforeCommitId

The full commit ID of the commit that was the tip of the destination branch when the pull request was created. This commit is superceded by the after commit in the source branch when and if you merge the source branch into the destination branch.

Type: String

Required: No

comments

An array of comment objects. Each comment object contains information about a comment on the pull request.

Type: Array of Comment objects

Required: No

location

Location information about the comment on the pull request, including the file name, line number, and whether the version of the file where the comment was made is BEFORE (destination branch) or AFTER (source branch).

Type: Location object

Required: No

pullRequestId

The system-generated ID of the pull request.

Type: String

Required: No

repositoryName

The name of the repository that contains the pull request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Commit

Returns information about a specific commit.

Contents

additionalData

Any other data associated with the specified commit.

Type: String

Required: No

author

Information about the author of the specified commit. Information includes the date in timestamp format with GMT offset, the name of the author, and the email address for the author, as configured in Git.

Type: UserInfo object

Required: No

commitId

The full SHA ID of the specified commit.

Type: String

Required: No

committer

Information about the person who committed the specified commit, also known as the committer. Information includes the date in timestamp format with GMT offset, the name of the committer, and the email address for the committer, as configured in Git.

For more information about the difference between an author and a committer in Git, see Viewing the Commit History in Pro Git by Scott Chacon and Ben Straub.

Type: UserInfo object

Required: No

message

The commit message associated with the specified commit.

Type: String

Required: No

parents

A list of parent commits for the specified commit. Each parent commit ID is the full commit ID.

Type: Array of strings

Required: No

treeId

Tree information for the specified commit.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Conflict

Information about conflicts in a merge operation.

Contents

conflictMetadata

Metadata about a conflict in a merge operation.

Type: ConflictMetadata object

Required: No

mergeHunks

A list of hunks that contain the differences between files or lines causing the conflict.

Type: Array of MergeHunk objects

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ConflictMetadata

Information about the metadata for a conflict in a merge operation.

Contents

contentConflict

A boolean value indicating whether there are conflicts in the content of a file.

Type: Boolean

Required: No

fileModeConflict

A boolean value indicating whether there are conflicts in the file mode of a file.

Type: Boolean

Required: No

fileModes

The file modes of the file in the source, destination, and base of the merge.

Type: FileModes object

Required: No

filePath

The path of the file that contains conflicts.

Type: String

Required: No

fileSizes

The file sizes of the file in the source, destination, and base of the merge.

Type: FileSizes object

Required: No

isBinaryFile

A boolean value (true or false) indicating whether the file is binary or textual in the source, destination, and base of the merge.

Type: IsBinaryFile object

Required: No

mergeOperations

Whether an add, modify, or delete operation caused the conflict between the source and destination of the merge.

Type: MergeOperations object

Required: No

numberOfConflicts

The number of conflicts, including both hunk conflicts and metadata conflicts.

Type: Integer

Required: No

objectTypeConflict

A boolean value (true or false) indicating whether there are conflicts between the branches in the object type of a file, folder, or submodule.

Type: Boolean

Required: No

objectTypes

Information about any object type conflicts in a merge operation.

Type: ObjectTypes object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ConflictResolution

If AUTOMERGE is the conflict resolution strategy, a list of inputs to use when resolving conflicts during a merge.

Contents

deleteFiles

Files to be deleted as part of the merge conflict resolution.

Type: Array of DeleteFileEntry objects

Required: No

replaceContents

Files to have content replaced as part of the merge conflict resolution.

Type: Array of ReplaceContentEntry objects

Required: No

setFileModes

File modes that are set as part of the merge conflict resolution.

Type: Array of SetFileModeEntry objects

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DeleteFileEntry

A file that is deleted as part of a commit.

Contents

filePath

The full path of the file to be deleted, including the name of the file.

Type: String

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Difference

Returns information about a set of differences for a commit specifier.

Contents

afterBlob

Information about an `afterBlob` data type object, including the ID, the file mode permission code, and the path.

Type: BlobMetadata object

Required: No

beforeBlob

Information about a `beforeBlob` data type object, including the ID, the file mode permission code, and the path.

Type: BlobMetadata object

Required: No

changeType

Whether the change type of the difference is an addition (A), deletion (D), or modification (M).

Type: String

Valid Values: `A | M | D`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Evaluation

Returns information about the approval rules applied to a pull request and whether conditions have been met.

Contents

approvalRulesNotSatisfied

The names of the approval rules that have not had their conditions met.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: No

approvalRulesSatisfied

The names of the approval rules that have had their conditions met.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: No

approved

Whether the state of the pull request is approved.

Type: Boolean

Required: No

overridden

Whether the approval rule requirements for the pull request have been overridden and no longer need to be met.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

File

Returns information about a file in a repository.

Contents

absolutePath

The fully qualified path to the file in the repository.

Type: String

Required: No

blobId

The blob ID that contains the file information.

Type: String

Required: No

fileMode

The extrapolated file mode permissions for the file. Valid values include EXECUTABLE and NORMAL.

Type: String

Valid Values: `EXECUTABLE | NORMAL | SYMLINK`

Required: No

relativePath

The relative path of the file from the folder where the query originated.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

FileMetadata

A file to be added, updated, or deleted as part of a commit.

Contents

absolutePath

The full path to the file to be added or updated, including the name of the file.

Type: String

Required: No

blobId

The blob ID that contains the file information.

Type: String

Required: No

fileMode

The extrapolated file mode permissions for the file. Valid values include EXECUTABLE and NORMAL.

Type: String

Valid Values: `EXECUTABLE | NORMAL | SYMLINK`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

FileModes

Information about file modes in a merge or pull request.

Contents

base

The file mode of a file in the base of a merge or pull request.

Type: String

Valid Values: `EXECUTABLE | NORMAL | SYMLINK`

Required: No

destination

The file mode of a file in the destination of a merge or pull request.

Type: String

Valid Values: `EXECUTABLE | NORMAL | SYMLINK`

Required: No

source

The file mode of a file in the source of a merge or pull request.

Type: String

Valid Values: `EXECUTABLE | NORMAL | SYMLINK`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2

- AWS SDK for Ruby V3

FileSizes

Information about the size of files in a merge or pull request.

Contents

base

The size of a file in the base of a merge or pull request.

Type: Long

Required: No

destination

The size of a file in the destination of a merge or pull request.

Type: Long

Required: No

source

The size of a file in the source of a merge or pull request.

Type: Long

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

FileVersion

Information about a version of a file.

Contents

blobId

The blob ID of the object that represents the content of the file in this version.

Type: String

Required: No

commit

Returns information about a specific commit.

Type: Commit object

Required: No

path

The name and path of the file at which this blob is indexed which contains the data for this version of the file. This value will vary between file versions if a file is renamed or if its path changes.

Type: String

Required: No

revisionChildren

An array of commit IDs that contain more recent versions of this file. If there are no additional versions of the file, this array will be empty.

Type: Array of strings

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Folder

Returns information about a folder in a repository.

Contents

absolutePath

The fully qualified path of the folder in the repository.

Type: String

Required: No

relativePath

The relative path of the specified folder from the folder where the query originated.

Type: String

Required: No

treeId

The full SHA-1 pointer of the tree information for the commit that contains the folder.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

IsBinaryFile

Information about whether a file is binary or textual in a merge or pull request operation.

Contents

base

The binary or non-binary status of a file in the base of a merge or pull request.

Type: Boolean

Required: No

destination

The binary or non-binary status of a file in the destination of a merge or pull request.

Type: Boolean

Required: No

source

The binary or non-binary status of file in the source of a merge or pull request.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Location

Returns information about the location of a change or comment in the comparison between two commits or a pull request.

Contents

filePath

The name of the file being compared, including its extension and subdirectory, if any.

Type: String

Required: No

filePosition

The position of a change in a compared file, in line number format.

Type: Long

Required: No

relativeFileVersion

In a comparison of commits or a pull request, whether the change is in the before or after of that comparison.

Type: String

Valid Values: `BEFORE | AFTER`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

MergeHunk

Information about merge hunks in a merge or pull request operation.

Contents

base

Information about the merge hunk in the base of a merge or pull request.

Type: MergeHunkDetail object

Required: No

destination

Information about the merge hunk in the destination of a merge or pull request.

Type: MergeHunkDetail object

Required: No

isConflict

A Boolean value indicating whether a combination of hunks contains a conflict. Conflicts occur when the same file or the same lines in a file were modified in both the source and destination of a merge or pull request. Valid values include true, false, and null. True when the hunk represents a conflict and one or more files contains a line conflict. File mode conflicts in a merge do not set this to true.

Type: Boolean

Required: No

source

Information about the merge hunk in the source of a merge or pull request.

Type: MergeHunkDetail object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

MergeHunkDetail

Information about the details of a merge hunk that contains a conflict in a merge or pull request operation.

Contents

endLine

The end position of the hunk in the merge result.

Type: Integer

Required: No

hunkContent

The base-64 encoded content of the hunk merged region that might contain a conflict.

Type: String

Required: No

startLine

The start position of the hunk in the merge result.

Type: Integer

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

MergeMetadata

Returns information about a merge or potential merge between a source reference and a destination reference in a pull request.

Contents

isMerged

A Boolean value indicating whether the merge has been made.

Type: Boolean

Required: No

mergeCommitId

The commit ID for the merge commit, if any.

Type: String

Required: No

mergedBy

The Amazon Resource Name (ARN) of the user who merged the branches.

Type: String

Required: No

mergeOption

The merge strategy used in the merge.

Type: String

Valid Values: `FAST_FORWARD_MERGE | SQUASH_MERGE | THREE_WAY_MERGE`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

MergeOperations

Information about the file operation conflicts in a merge operation.

Contents

destination

The operation on a file in the destination of a merge or pull request.

Type: String

Valid Values: `A | M | D`

Required: No

source

The operation (add, modify, or delete) on a file in the source of a merge or pull request.

Type: String

Valid Values: `A | M | D`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ObjectTypes

Information about the type of an object in a merge operation.

Contents

base

The type of the object in the base commit of the merge.

Type: String

Valid Values: `FILE | DIRECTORY | GIT_LINK | SYMBOLIC_LINK`

Required: No

destination

The type of the object in the destination branch.

Type: String

Valid Values: `FILE | DIRECTORY | GIT_LINK | SYMBOLIC_LINK`

Required: No

source

The type of the object in the source branch.

Type: String

Valid Values: `FILE | DIRECTORY | GIT_LINK | SYMBOLIC_LINK`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2

- AWS SDK for Ruby V3

OriginApprovalRuleTemplate

Returns information about the template that created the approval rule for a pull request.

Contents

approvalRuleTemplateId

The ID of the template that created the approval rule.

Type: String

Required: No

approvalRuleTemplateName

The name of the template that created the approval rule.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

PullRequest

Returns information about a pull request.

Contents

approvalRules

The approval rules applied to the pull request.

Type: Array of ApprovalRule objects

Required: No

authorArn

The Amazon Resource Name (ARN) of the user who created the pull request.

Type: String

Required: No

clientRequestToken

A unique, client-generated idempotency token that, when provided in a request, ensures the request cannot be repeated with a changed parameter. If a request is received with the same parameters and a token is included, the request returns information about the initial request that used that token.

Type: String

Required: No

creationDate

The date and time the pull request was originally created, in timestamp format.

Type: Timestamp

Required: No

description

The user-defined description of the pull request. This description can be used to clarify what should be reviewed and other details of the request.

Type: String

Length Constraints: Maximum length of 10240.

Required: No

lastActivityDate

The day and time of the last user or system activity on the pull request, in timestamp format.

Type: Timestamp

Required: No

pullRequestId

The system-generated ID of the pull request.

Type: String

Required: No

pullRequestStatus

The status of the pull request. Pull request status can only change from `OPEN` to `CLOSED`.

Type: String

Valid Values: `OPEN | CLOSED`

Required: No

pullRequestTargets

The targets of the pull request, including the source branch and destination branch for the pull request.

Type: Array of PullRequestTarget objects

Required: No

revisionId

The system-generated revision ID for the pull request.

Type: String

Required: No

title

The user-defined title of the pull request. This title is displayed in the list of pull requests to other repository users.

Type: String

Length Constraints: Maximum length of 150.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

PullRequestCreatedEventMetadata

Metadata about the pull request that is used when comparing the pull request source with its destination.

Contents

destinationCommitId

The commit ID of the tip of the branch specified as the destination branch when the pull request was created.

Type: String

Required: No

mergeBase

The commit ID of the most recent commit that the source branch and the destination branch have in common.

Type: String

Required: No

repositoryName

The name of the repository where the pull request was created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: No

sourceCommitId

The commit ID on the source branch used when the pull request was created.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

PullRequestEvent

Returns information about a pull request event.

Contents

actorArn

The Amazon Resource Name (ARN) of the user whose actions resulted in the event. Examples include updating the pull request with more commits or changing the status of a pull request.

Type: String

Required: No

approvalRuleEventMetadata

Information about a pull request event.

Type: ApprovalRuleEventMetadata object

Required: No

approvalRuleOverriddenEventMetadata

Information about an approval rule override event for a pull request.

Type: ApprovalRuleOverriddenEventMetadata object

Required: No

approvalStateChangedEventMetadata

Information about an approval state change for a pull request.

Type: ApprovalStateChangedEventMetadata object

Required: No

eventDate

The day and time of the pull request event, in timestamp format.

Type: Timestamp

Required: No

pullRequestCreatedEventMetadata

Information about the source and destination branches for the pull request.

Type: PullRequestCreatedEventMetadata object

Required: No

pullRequestEventType

The type of the pull request event (for example, a status change event (PULL_REQUEST_STATUS_CHANGED) or update event (PULL_REQUEST_SOURCE_REFERENCE_UPDATED)).

Type: String

Valid Values: `PULL_REQUEST_CREATED | PULL_REQUEST_STATUS_CHANGED | PULL_REQUEST_SOURCE_REFERENCE_UPDATED | PULL_REQUEST_MERGE_STATE_CHANGED | PULL_REQUEST_APPROVAL_RULE_CREATED | PULL_REQUEST_APPROVAL_RULE_UPDATED | PULL_REQUEST_APPROVAL_RULE_DELETED | PULL_REQUEST_APPROVAL_RULE_OVERRIDDEN | PULL_REQUEST_APPROVAL_STATE_CHANGED`

Required: No

pullRequestId

The system-generated ID of the pull request.

Type: String

Required: No

pullRequestMergedStateChangedEventMetadata

Information about the change in mergability state for the pull request event.

Type: PullRequestMergedStateChangedEventMetadata object

Required: No

pullRequestSourceReferenceUpdatedEventMetadata

Information about the updated source branch for the pull request event.

Type: PullRequestSourceReferenceUpdatedEventMetadata object

Required: No

pullRequestStatusChangedEventMetadata

Information about the change in status for the pull request event.

Type: PullRequestStatusChangedEventMetadata object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

PullRequestMergedStateChangedEventMetadata

Returns information about the change in the merge state for a pull request event.

Contents

destinationReference

The name of the branch that the pull request is merged into.

Type: String

Required: No

mergeMetadata

Information about the merge state change event.

Type: MergeMetadata object

Required: No

repositoryName

The name of the repository where the pull request was created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

PullRequestSourceReferenceUpdatedEventMetadata

Information about an update to the source branch of a pull request.

Contents

afterCommitId

The full commit ID of the commit in the source branch that was the tip of the branch at the time the pull request was updated.

Type: String

Required: No

beforeCommitId

The full commit ID of the commit in the destination branch that was the tip of the branch at the time the pull request was updated.

Type: String

Required: No

mergeBase

The commit ID of the most recent commit that the source branch and the destination branch have in common.

Type: String

Required: No

repositoryName

The name of the repository where the pull request was updated.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

PullRequestStatusChangedEventMetadata

Information about a change to the status of a pull request.

Contents

pullRequestStatus

The changed status of the pull request.

Type: String

Valid Values: `OPEN | CLOSED`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

PullRequestTarget

Returns information about a pull request target.

Contents

destinationCommit

The full commit ID that is the tip of the destination branch. This is the commit where the pull request was or will be merged.

Type: String

Required: No

destinationReference

The branch of the repository where the pull request changes are merged. Also known as the destination branch.

Type: String

Required: No

mergeBase

The commit ID of the most recent commit that the source branch and the destination branch have in common.

Type: String

Required: No

mergeMetadata

Returns metadata about the state of the merge, including whether the merge has been made.

Type: MergeMetadata object

Required: No

repositoryName

The name of the repository that contains the pull request source and destination branches.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: No

sourceCommit

The full commit ID of the tip of the source branch used to create the pull request. If the pull request branch is updated by a push while the pull request is open, the commit ID changes to reflect the new tip of the branch.

Type: String

Required: No

sourceReference

The branch of the repository that contains the changes for the pull request. Also known as the source branch.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

PutFileEntry

Information about a file added or updated as part of a commit.

Contents

filePath

The full path to the file in the repository, including the name of the file.

Type: String

Required: Yes

fileContent

The content of the file, if a source file is not specified.

Type: Base64-encoded binary data object

Length Constraints: Maximum length of 6291456.

Required: No

fileMode

The extrapolated file mode permissions for the file. Valid values include EXECUTABLE and NORMAL.

Type: String

Valid Values: `EXECUTABLE | NORMAL | SYMLINK`

Required: No

sourceFile

The name and full path of the file that contains the changes you want to make as part of the commit, if you are not providing the file content directly.

Type: SourceFileSpecifier object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ReactionForComment

Information about the reaction values provided by users on a comment.

Contents

reaction

The reaction for a specified comment.

Type: ReactionValueFormats object

Required: No

reactionsFromDeletedUsersCount

A numerical count of users who reacted with the specified emoji whose identities have been subsequently deleted from IAM. While these IAM users or roles no longer exist, the reactions might still appear in total reaction counts.

Type: Integer

Required: No

reactionUsers

The Amazon Resource Names (ARNs) of users who have provided reactions to the comment.

Type: Array of strings

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ReactionValueFormats

Information about the values for reactions to a comment. AWS CodeCommit supports a limited set of reactions.

Contents

emoji

The Emoji Version 1.0 graphic of the reaction. These graphics are interpreted slightly differently on different operating systems.

Type: String

Required: No

shortCode

The emoji short code for the reaction. Short codes are interpreted slightly differently on different operating systems.

Type: String

Required: No

unicode

The Unicode codepoint for the reaction.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ReplaceContentEntry

Information about a replacement content entry in the conflict of a merge or pull request operation.

Contents

filePath

The path of the conflicting file.

Type: String

Required: Yes

replacementType

The replacement type to use when determining how to resolve the conflict.

Type: String

Valid Values: `KEEP_BASE | KEEP_SOURCE | KEEP_DESTINATION | USE_NEW_CONTENT`

Required: Yes

content

The base-64 encoded content to use when the replacement type is USE_NEW_CONTENT.

Type: Base64-encoded binary data object

Length Constraints: Maximum length of 6291456.

Required: No

fileMode

The file mode to apply during conflict resoltion.

Type: String

Valid Values: `EXECUTABLE | NORMAL | SYMLINK`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

RepositoryMetadata

Information about a repository.

Contents

accountId

The ID of the AWS account associated with the repository.

Type: String

Required: No

Arn

The Amazon Resource Name (ARN) of the repository.

Type: String

Required: No

cloneUrlHttp

The URL to use for cloning the repository over HTTPS.

Type: String

Required: No

cloneUrlSsh

The URL to use for cloning the repository over SSH.

Type: String

Required: No

creationDate

The date and time the repository was created, in timestamp format.

Type: Timestamp

Required: No

defaultBranch

The repository's default branch name.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

kmsKeyId

The ID of the AWS Key Management Service encryption key used to encrypt and decrypt the repository.

Type: String

Pattern: `^[a-zA-Z0-9:/_-]+$`

Required: No

lastModifiedDate

The date and time the repository was last modified, in timestamp format.

Type: Timestamp

Required: No

repositoryDescription

A comment or description about the repository.

Type: String

Length Constraints: Maximum length of 1000.

Required: No

repositoryId

The ID of the repository.

Type: String

Required: No

repositoryName

The repository's name.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

RepositoryNameIdPair

Information about a repository name and ID.

Contents

repositoryId

The ID associated with the repository.

Type: String

Required: No

repositoryName

The name associated with the repository.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

RepositoryTrigger

Information about a trigger for a repository.

> **Note**
>
> If you want to receive notifications about repository events, consider using notifications instead of triggers. For more information, see [Configuring notifications for repository events](#).

Contents

destinationArn

The ARN of the resource that is the target for a trigger (for example, the ARN of a topic in Amazon SNS).

Type: String

Required: Yes

events

The repository events that cause the trigger to run actions in another service, such as sending a notification through Amazon SNS.

> **Note**
>
> The valid value "all" cannot be used with any other values.

Type: Array of strings

Valid Values: `all | updateReference | createReference | deleteReference`

Required: Yes

name

The name of the trigger.

Type: String

Required: Yes

branches

The branches to be included in the trigger configuration. If you specify an empty array, the trigger applies to all branches.

> **Note**
>
> Although no content is required in the array, you must include the array itself.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

customData

Any custom data associated with the trigger to be included in the information sent to the target of the trigger.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

RepositoryTriggerExecutionFailure

A trigger failed to run.

Contents

failureMessage

Message information about the trigger that did not run.

Type: String

Required: No

trigger

The name of the trigger that did not run.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

SetFileModeEntry

Information about the file mode changes.

Contents

fileMode

The file mode for the file.

Type: String

Valid Values: `EXECUTABLE | NORMAL | SYMLINK`

Required: Yes

filePath

The full path to the file, including the name of the file.

Type: String

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

SourceFileSpecifier

Information about a source file that is part of changes made in a commit.

Contents

filePath

The full path to the file, including the name of the file.

Type: String

Required: Yes

isMove

Whether to remove the source file from the parent commit.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

SubModule

Returns information about a submodule reference in a repository folder.

Contents

absolutePath

The fully qualified path to the folder that contains the reference to the submodule.

Type: String

Required: No

commitId

The commit ID that contains the reference to the submodule.

Type: String

Required: No

relativePath

The relative path of the submodule from the folder where the query originated.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

SymbolicLink

Returns information about a symbolic link in a repository folder.

Contents

absolutePath

The fully qualified path to the folder that contains the symbolic link.

Type: String

Required: No

blobId

The blob ID that contains the information about the symbolic link.

Type: String

Required: No

fileMode

The file mode permissions of the blob that cotains information about the symbolic link.

Type: String

Valid Values: `EXECUTABLE | NORMAL | SYMLINK`

Required: No

relativePath

The relative path of the symbolic link from the folder where the query originated.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Target

Returns information about a target for a pull request.

Contents

repositoryName

The name of the repository that contains the pull request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `[\w\.-]+`

Required: Yes

sourceReference

The branch of the repository that contains the changes for the pull request. Also known as the source branch.

Type: String

Required: Yes

destinationReference

The branch of the repository where the pull request changes are merged. Also known as the destination branch.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++

- AWS SDK for Java V2
- AWS SDK for Ruby V3

UserInfo

Information about the user who made a specified commit.

Contents

date

The date when the specified commit was commited, in timestamp format with GMT offset.

Type: String

Required: No

email

The email address associated with the user who made the commit, if any.

Type: String

Required: No

name

The name of the user who made the specified commit.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see Signing AWS API requests in the *IAM User Guide*.

Action

The action to be performed.

Type: string

Required: Yes

Version

The API version that the request is written for, expressed in the format YYYY-MM-DD.

Type: string

Required: Yes

X-Amz-Algorithm

The hash algorithm that you used to create the request signature.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Valid Values: `AWS4-HMAC-SHA256`

Required: Conditional

X-Amz-Credential

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4_request"). The value is expressed in the following format: *access_key*/*YYYYMMDD*/*region*/*service*/ aws4_request.

For more information, see Create a signed AWS API request in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

X-Amz-Date

The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: `20120325T120000Z`.

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see Elements of an AWS API request signature in the *IAM User Guide*.

Type: string

Required: Conditional

X-Amz-Security-Token

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS STS, see AWS services that work with IAM in the *IAM User Guide*.

Condition: If you're using temporary security credentials from AWS STS, you must include the security token.

Type: string

Required: Conditional

X-Amz-Signature

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

X-Amz-SignedHeaders

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see Create a signed AWS API request in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

Common Errors

This section lists the errors common to the API actions of all AWS services. For errors specific to an API action for this service, see the topic for that API action.

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

IncompleteSignature

The request signature does not conform to AWS standards.

HTTP Status Code: 400

InternalFailure

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

InvalidAction

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400

InvalidClientTokenId

The X.509 certificate or AWS access key ID provided does not exist in our records.

HTTP Status Code: 403

NotAuthorized

You do not have permission to perform this action.

HTTP Status Code: 400

OptInRequired

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

RequestExpired

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

ServiceUnavailable

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 400

ValidationError

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400