See Also ....................................................................................................................................... 61
GetDifferences .......................................................................................................................... 62
    Request Syntax .................................................................................................................. 62
    Request Parameters ......................................................................................................... 62
    Response Syntax ............................................................................................................... 63
    Response Elements ........................................................................................................... 63
    Errors ............................................................................................................................... 64
    Example ............................................................................................................................. 65
    See Also ............................................................................................................................. 66
GetMergeConflicts .................................................................................................................. 67
    Request Syntax .................................................................................................................. 67
    Request Parameters ......................................................................................................... 67
    Response Syntax ............................................................................................................... 68
    Response Elements ........................................................................................................... 68
    Errors ............................................................................................................................... 68
    Example ............................................................................................................................. 70
    See Also ............................................................................................................................. 70
GetPullRequest ......................................................................................................................... 72
    Request Syntax .................................................................................................................. 72
    Request Parameters ......................................................................................................... 72
    Response Syntax ............................................................................................................... 72
    Response Elements ........................................................................................................... 72
    Errors ............................................................................................................................... 73
    Example ............................................................................................................................. 74
    See Also ............................................................................................................................. 74
GetRepository .......................................................................................................................... 76
    Request Syntax .................................................................................................................. 76
    Request Parameters ......................................................................................................... 76
    Response Syntax ............................................................................................................... 76
    Response Elements ........................................................................................................... 76
    Errors ............................................................................................................................... 77
    Example ............................................................................................................................. 78
    See Also ............................................................................................................................. 78
GetRepositoryTriggers ............................................................................................................. 80
    Request Syntax .................................................................................................................. 80
    Request Parameters ......................................................................................................... 80
    Response Syntax ............................................................................................................... 80
    Response Elements ........................................................................................................... 80
    Errors ............................................................................................................................... 81
    Example ............................................................................................................................. 82
    See Also ............................................................................................................................. 82
ListBranches .......................................................................................................................... 84
    Request Syntax .................................................................................................................. 84
    Request Parameters ......................................................................................................... 84
    Response Syntax ............................................................................................................... 84
    Response Elements ........................................................................................................... 84
    Errors ............................................................................................................................... 85
    Example ............................................................................................................................. 86
    See Also ............................................................................................................................. 86
ListPullRequests ....................................................................................................................... 88
    Request Syntax .................................................................................................................. 88
    Request Parameters ......................................................................................................... 88
    Response Syntax ............................................................................................................... 89
    Response Elements ........................................................................................................... 89
    Errors ............................................................................................................................... 89
    Example ............................................................................................................................. 90
    See Also ............................................................................................................................. 91
<table>
<thead>
<tr>
<th>Data Types</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>BlobMetadata</td>
<td>153</td>
</tr>
<tr>
<td>UpdateRepositoryName</td>
<td>150</td>
</tr>
<tr>
<td>UpdateRepositoryDescription</td>
<td>147</td>
</tr>
<tr>
<td>UpdatePullRequestTitle</td>
<td>144</td>
</tr>
<tr>
<td>UpdatePullRequestStatus</td>
<td>141</td>
</tr>
<tr>
<td>UpdatePullRequestDescription</td>
<td>138</td>
</tr>
<tr>
<td>UpdateDefaultBranch</td>
<td>135</td>
</tr>
<tr>
<td>Data Types</td>
<td>152</td>
</tr>
<tr>
<td>BlobMetadata</td>
<td>153</td>
</tr>
<tr>
<td>Contents</td>
<td>153</td>
</tr>
<tr>
<td>See Also</td>
<td>153</td>
</tr>
<tr>
<td>BranchInfo</td>
<td>154</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>128</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>128</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>128</td>
</tr>
<tr>
<td>Response Elements</td>
<td>128</td>
</tr>
<tr>
<td>Errors</td>
<td>129</td>
</tr>
<tr>
<td>Example</td>
<td>129</td>
</tr>
<tr>
<td>See Also</td>
<td>130</td>
</tr>
<tr>
<td>UpdateDefaultBranch</td>
<td>131</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>131</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>131</td>
</tr>
<tr>
<td>Response Elements</td>
<td>131</td>
</tr>
<tr>
<td>Errors</td>
<td>131</td>
</tr>
<tr>
<td>Example</td>
<td>133</td>
</tr>
<tr>
<td>See Also</td>
<td>133</td>
</tr>
<tr>
<td>UpdatePullRequestDescription</td>
<td>134</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>134</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>134</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>134</td>
</tr>
<tr>
<td>Response Elements</td>
<td>135</td>
</tr>
<tr>
<td>Errors</td>
<td>135</td>
</tr>
<tr>
<td>Example</td>
<td>135</td>
</tr>
<tr>
<td>See Also</td>
<td>136</td>
</tr>
<tr>
<td>UpdatePullRequestStatus</td>
<td>138</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>138</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>138</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>138</td>
</tr>
<tr>
<td>Response Elements</td>
<td>139</td>
</tr>
<tr>
<td>Errors</td>
<td>139</td>
</tr>
<tr>
<td>Example</td>
<td>140</td>
</tr>
<tr>
<td>See Also</td>
<td>141</td>
</tr>
<tr>
<td>UpdatePullRequestTitle</td>
<td>142</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>142</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>142</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>142</td>
</tr>
<tr>
<td>Response Elements</td>
<td>143</td>
</tr>
<tr>
<td>Errors</td>
<td>143</td>
</tr>
<tr>
<td>Example</td>
<td>144</td>
</tr>
<tr>
<td>See Also</td>
<td>144</td>
</tr>
<tr>
<td>UpdateRepositoryDescription</td>
<td>146</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>146</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>146</td>
</tr>
<tr>
<td>Response Elements</td>
<td>146</td>
</tr>
<tr>
<td>Errors</td>
<td>146</td>
</tr>
<tr>
<td>Example</td>
<td>147</td>
</tr>
<tr>
<td>See Also</td>
<td>148</td>
</tr>
<tr>
<td>UpdateRepositoryName</td>
<td>149</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>149</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>149</td>
</tr>
<tr>
<td>Response Elements</td>
<td>149</td>
</tr>
<tr>
<td>Errors</td>
<td>149</td>
</tr>
<tr>
<td>Example</td>
<td>150</td>
</tr>
<tr>
<td>See Also</td>
<td>150</td>
</tr>
</tbody>
</table>
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 (p. 4)**, which returns information about one or more repositories associated with your AWS account.
- **CreateRepository (p. 18)**, which creates an AWS CodeCommit repository.
- **DeleteRepository (p. 29)**, which deletes an AWS CodeCommit repository.
- **GetRepository (p. 76)**, which returns information about a specified repository.
- **ListRepositories (p. 92)**, which lists all AWS CodeCommit repositories associated with your AWS account.
- **UpdateRepositoryDescription (p. 146)**, which sets or updates the description of the repository.
- **UpdateRepositoryName (p. 149)**, which changes the name of the repository. If you change the name of a repository, no other users of that repository will be able to access it until you send them the new HTTPS or SSH URL to use.

Branches, by calling the following:

- **CreateBranch (p. 8)**, which creates a new branch in a specified repository.
- **DeleteBranch (p. 22)**, which deletes the specified branch in a repository unless it is the default branch.
- **GetBranch (p. 40)**, which returns information about a specified branch.
- **ListBranches (p. 84)**, which lists all branches for a specified repository.
- **UpdateDefaultBranch (p. 131)**, which changes the default branch for a repository.

Information about committed code in a repository, by calling the following:

- **GetBlob (p. 37)**, which returns the base-64 encoded content of an individual Git blob object within a repository.
- **GetCommit (p. 58)**, which returns information about a commit, including commit messages and author and committer information.
- **GetDifferences (p. 62)**, which returns information about the differences in a valid commit specifier (such as a branch, tag, HEAD, commit ID or other fully qualified reference).

Pull requests, by calling the following:

- **CreatePullRequest (p. 12)**, which creates a pull request in a specified repository.
- **DescribePullRequestEvents (p. 32)**, which returns information about one or more pull request events.
- **GetCommentsForPullRequest (p. 52)**, which returns information about comments on a specified pull request.
- **GetMergeConflicts (p. 67)**, which returns information about merge conflicts between the source and destination branch in a pull request.
• **GetPullRequest (p. 72)**, which returns information about a specified pull request.
• **ListPullRequests (p. 88)**, which lists all pull requests for a repository.
• **MergePullRequestByFastForward (p. 95)**, 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.
• **PostCommentForPullRequest (p. 107)**, which posts a comment to a pull request at the specified line, file, or request.
• **UpdatePullRequestDescription (p. 134)**, which updates the description of a pull request.
• **UpdatePullRequestStatus (p. 138)**, which updates the status of a pull request.
• **UpdatePullRequestTitle (p. 142)**, which updates the title of a pull request.

Information about comments in a repository, by calling the following:

• **DeleteCommentContent (p. 26)**, which deletes the content of a comment on a commit in a repository.
• **GetComment (p. 44)**, which returns information about a comment on a commit.
• **GetCommentsForComparedCommit (p. 47)**, which returns information about comments on the comparison between two commit specifiers in a repository.
• **PostCommentForComparedCommit (p. 100)**, which creates a comment on the comparison between two commit specifiers in a repository.
• **PostCommentReply (p. 114)**, which creates a reply to a comment.
• **UpdateComment (p. 128)**, which updates the content of a comment on a commit in a repository.

Triggers, by calling the following:

• **GetRepositoryTriggers (p. 80)**, which returns information about triggers configured for a repository.
• **PutRepositoryTriggers (p. 118)**, which replaces all triggers for a repository and can be used to create or delete triggers.
• **TestRepositoryTriggers (p. 123)**, 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](https://aws.amazon.com/documentation/coderepos/).

This document was last published on February 7, 2018.
The following actions are supported:

- BatchGetRepositories (p. 4)
- CreateBranch (p. 8)
- CreatePullRequest (p. 12)
- CreateRepository (p. 18)
- DeleteBranch (p. 22)
- DeleteCommentContent (p. 26)
- DeleteRepository (p. 29)
- DescribePullRequestEvents (p. 32)
- GetBlob (p. 37)
- GetBranch (p. 40)
- GetComment (p. 44)
- GetCommentsForComparedCommit (p. 47)
- GetCommentsForPullRequest (p. 52)
- GetCommit (p. 58)
- GetDifferences (p. 62)
- GetMergeConflicts (p. 67)
- GetPullRequest (p. 72)
- GetRepository (p. 76)
- GetRepositoryTriggers (p. 80)
- ListBranches (p. 84)
- ListPullRequests (p. 88)
- ListRepositories (p. 92)
- MergePullRequestByFastForward (p. 95)
- PostCommentForComparedCommit (p. 100)
- PostCommentForPullRequest (p. 107)
- PostCommentReply (p. 114)
- PutRepositoryTriggers (p. 118)
- TestRepositoryTriggers (p. 123)
- UpdateComment (p. 128)
- UpdateDefaultBranch (p. 131)
- UpdatePullRequestDescription (p. 134)
- UpdatePullRequestStatus (p. 138)
- UpdatePullRequestTitle (p. 142)
- UpdateRepositoryDescription (p. 146)
- UpdateRepositoryName (p. 149)
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 web page could 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 web page.

**Request Syntax**

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

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

**repositoryNames (p. 4)**

The names of the repositories to get information about.

Type: Array of strings

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

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

Required: Yes

**Response Syntax**

```json
{
  "repositories": [
    {
      "accountId": "string",
      "Arn": "string",
      "cloneUrlHttp": "string",
      "cloneUrlSsh": "string",
      "creationDate": number,
      "defaultBranch": "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.

repositories (p. 4)
A list of repositories returned by the batch get repositories operation.
Type: Array of RepositoryMetadata (p. 175) objects

repositoriesNotFound (p. 4)
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 (p. 185).

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
At least one specified repository name is not valid.

Note
This exception only occurs 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 25.

HTTP Status Code: 400

**RepositoryNamesRequiredException**

A repository names object is required but was not specified.

HTTP Status Code: 400

Example

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=AKIAI44QH8DHEXAMPLE/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": "master",
            "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",
            "creationDate": 1441725693.583,
            "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
            "Arn": "arn:aws:codecommit:us-east-1:111111111111EXAMPLE:MyDemoRepo",
            "accountId": "111111111111"
        }
    ]
}
```
See Also

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

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

Creates a new 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**

```json
{
  "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 (p. 8)**

The name of the new branch to create.

Type: String

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

Required: Yes

**commitId (p. 8)**

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

Type: String

Required: Yes

**repositoryName (p. 8)**

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 (p. 185).

**BranchNameExistsException**

The specified branch name already exists.

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

At least one specified repository name is not valid.

Note
This exception only occurs 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

Example

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:
See Also

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

Creates a pull request in the specified repository.

Request Syntax

```json
{
  "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 (p. 183).

The request accepts the following data in JSON format.

clientRequestToken (p. 12)

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 will return information about the initial request that used that token.

**Note**
The AWS SDKs prepopulate client request tokens. If using an AWS SDK, you do not have to generate an idempotency token, as this will be done for you.

Type: String

Required: No

description (p. 12)

A description of the pull request.

Type: String

Length Constraints: Maximum length of 10240.

Required: No

targets (p. 12)

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 (p. 181) objects

Required: Yes
**Response Syntax**

```
{
    "pullRequest": {
        "authorArn": "string",
        "clientRequestToken": "string",
        "creationDate": number,
        "description": "string",
        "lastActivityDate": number,
        "pullRequestId": "string",
        "pullRequestStatus": "string",
        "pullRequestTargets": [
            {
                "destinationCommit": "string",
                "destinationReference": "string",
                "mergeMetadata": {
                    "isMerged": boolean,
                    "mergedBy": "string"
                },
                "repositoryName": "string",
                "sourceCommit": "string",
                "sourceReference": "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 (p. 13)**

Information about the newly created pull request.

Type: PullRequest (p. 166) object

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 185).

**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 will return 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 re-used.

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 are limited to 1,000 characters in length.

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/master. For more information, see [Git Internals - Git References](https://git-scm.com/book) or consult your Git documentation.

HTTP Status Code: 400

**InvalidRepositoryNameException**

At least one specified repository name is not valid.

**Note**

This exception only occurs 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 the 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

Example

Sample Request

```plaintext
>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=AKIAI44QH8DH8EXAMPLE/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/master",
      "repositoryName": "MyDemoRepo",
      "sourceReference": "refs/heads/jane-branch"
    }
  ],
  "title": "Pronunciation difficulty analyzer"
}
```

Sample Response

```plaintext
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847
```
See Also

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

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

Creates a new, empty repository.

**Request Syntax**

```json
{
  "repositoryDescription": "string",
  "repositoryName": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

**repositoryDescription (p. 18)**

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 web page could 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 web page.

Type: String

Length Constraints: Maximum length of 1000.

Required: No

**repositoryName (p. 18)**

The name of the new repository to be created.

*Note*

The repository name must be unique across the calling AWS account. In addition, repository names are limited to 100 alphanumeric, dash, and underscore characters, and cannot include certain characters. For a full description of the limits on repository names, see Limits 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

**Response Syntax**

```json
{
}
```
"repositoryMetadata": {
    "accountId": "string",
    "Arn": "string",
    "cloneUrlHttp": "string",
    "cloneUrlSsh": "string",
    "creationDate": number,
    "defaultBranch": "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 (p. 18)

Information about the newly created repository.

Type: RepositoryMetadata (p. 175) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

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

At least one specified repository name is not valid.

**Note**
This exception only occurs 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
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

Example

Sample Request

```plaintext
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"
}
```

Sample Response

```plaintext
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",
    "creationDate": 1446071622.494,
    "repositoryId": "f7579e13-b3b3-e4027-aaef-650c0EXAMPLE",
    "Arn": "arn:aws:codecommit:us-east-1:111111111111:MyDemoRepo",
    "accountId": "111111111111"
}

See Also

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

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

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

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

`branchName (p. 22)`

The name of the branch to delete.

- Type: String
- Required: Yes

`repositoryName (p. 22)`

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

- Type: String
- Pattern: `[\w\.-]+`
- Required: Yes

Response Syntax

```json
{
   "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 (p. 22)

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 (p. 154) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

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

At least one specified repository name is not valid.
Note
This exception only occurs 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

Example

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:
See Also

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

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

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

**commentId (p. 26)**

The unique, system-generated ID of the comment. To get this ID, use GetCommentsForComparedCommit (p. 47) or GetCommentsForPullRequest (p. 52).

Type: String

Required: Yes

Response Syntax

```json
{
   "comment": {
      "authorArn": "string",
      "clientRequestToken": "string",
      "commentId": "string",
      "content": "string",
      "creationDate": number,
      "deleted": boolean,
      "inReplyTo": "string",
      "lastModifiedDate": 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 (p. 26)**

Information about the comment you just deleted.

Type: Comment (p. 155) object
Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

**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 provided 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

Example

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

{
```
"comment": {
    "creationDate": 1508369768.142,
    "deleted": true,
    "lastModifiedDate": 1508369842.278,
    "clientRequestToken": "123Example",
    "commentId": "ff30b348EXAMPLEb9aa670f",
    "authorArn": "arn:aws:iam::111111111111: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
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DeleteRepository

Deletes a repository. If a specified repository was already deleted, a null repository ID will be 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 will fail.

**Request Syntax**

```json
{
    "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** *(p. 29)*

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**

```json
{
    "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** *(p. 29)*

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
At least one specified repository name is not valid.

Note
This exception only occurs 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

Example

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=AKIAI44QH8DHBFEXAMPLE/20151028/us-east-1/codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-target, Signature=8d9b5998EXAMPLE
```
See Also

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

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

Returns information about one or more pull request events.

Request Syntax

```json
{
  "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 (p. 183).

The request accepts the following data in JSON format.

actorArn (p. 32)

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

Type: String

Required: No

maxResults (p. 32)

A 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 (p. 32)

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

Type: String

Required: No

pullRequestEventType (p. 32)

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

Required: No

pullRequestId (p. 32)

The system-generated ID of the pull request. To get this ID, use ListPullRequests (p. 88).
Response Syntax

```json
{
    "nextToken": "string",
    "pullRequestEvents": [
    {
        "actorArn": "string",
        "eventDate": number,
        "pullRequestEventType": "string",
        "pullRequestId": "string",
        "pullRequestMergedStateChangedEventMetadata": {
            "destinationReference": "string",
            "mergeMetadata": {
                "isMerged": boolean,
                "mergedBy": "string"
            },
            "repositoryName": "string"
        },
        "pullRequestSourceReferenceUpdatedEventMetadata": {
            "afterCommitId": "string",
            "beforeCommitId": "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 (p. 33)**

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

Type: String

**pullRequestEvents (p. 33)**

Information about the pull request events.

Type: Array of PullRequestEvent (p. 168) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

**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.
**PullRequestIdRequiredException**

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

HTTP Status Code: 400

### Example

#### Sample Request

```plaintext
>>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=AKIAI44QH8DH9EXAMPLE/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::111111111111:user/Zhang_Wei"
        },
        {
            "pullRequestStatusChangedEventMetadata": {
                "pullRequestId": "8",
                "pullRequestEventType": "PULL_REQUEST_STATUS_CHANGED",
                "eventDate": 1510341930.72,
                "actor": "arn:aws:iam::111111111111: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
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
GetBlob

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

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

**blobId (p. 37)**

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

Type: String

Required: Yes

**repositoryName (p. 37)**

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

```json
{
    "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 (p. 37)**

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 (p. 185).

**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 AWS CodeCommit User Guide.

HTTP Status Code: 400

**InvalidBlobIdException**

The specified blob is not valid.

HTTP Status Code: 400

**InvalidRepositoryNameException**

At least one specified repository name is not valid.

*Note*

This exception only occurs 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
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
GetBranch

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

**Request Syntax**

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

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

**branchName (p. 40)**

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 (p. 40)**

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**

```json
{
  "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 (p. 40)

The name of the branch.

Type: BranchInfo (p. 154) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

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

At least one specified repository name is not valid.

Note

This exception only occurs 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

## Example

### Sample Request

```bash
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

```bash
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
- AWS SDK for .NET
- AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
GetComment

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

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

**commentId (p. 44)**

The unique, system-generated ID of the comment. To get this ID, use GetCommentsForComparedCommit (p. 47) or GetCommentsForPullRequest (p. 52).

Type: String

Required: Yes

Response Syntax

```json
{
   "comment": {
      "authorArn": "string",
      "clientRequestToken": "string",
      "commentId": "string",
      "content": "string",
      "creationDate": number,
      "deleted": boolean,
      "inReplyTo": "string",
      "lastModifiedDate": 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 (p. 44)**

The contents of the comment.

Type: Comment (p. 155) object
Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

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 provided 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

Example

Sample Request

```bash
>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

```bash
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847

{
    "comment": {
```
"authorArn": "arn:aws:iam::111111111111: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
}

See Also

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

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

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

**Request Syntax**

```json
{
    "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 (p. 183).

The request accepts the following data in JSON format.

- **afterCommitId (p. 47)**
  
  To establish the directionality of the comparison, the full commit ID of the 'after' commit.
  
  Type: String
  
  Required: Yes

- **beforeCommitId (p. 47)**
  
  To establish the directionality of the comparison, the full commit ID of the 'before' commit.
  
  Type: String
  
  Required: No

- **maxResults (p. 47)**
  
  A non-negative integer used to limit the number of returned results. The default is 100 comments, and is configurable up to 500.
  
  Type: Integer
  
  Required: No

- **nextToken (p. 47)**
  
  An enumeration token that when provided in a request, returns the next batch of the results.
  
  Type: String
  
  Required: No

- **repositoryName (p. 47)**
  
  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**

```json
{
    "commentsForComparedCommitData": [
        {
            "afterBlobId": "string",
            "afterCommitId": "string",
            "beforeBlobId": "string",
            "beforeCommitId": "string",
            "comments": [
                {
                    "authorArn": "string",
                    "clientRequestToken": "string",
                    "commentId": "string",
                    "content": "string",
                    "creationDate": number,
                    "deleted": boolean,
                    "inReplyTo": "string",
                    "lastModifiedDate": 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 (p. 48)**
  A list of comment objects on the compared commit.
  Type: Array of CommentsForComparedCommit (p. 157) objects

- **nextToken (p. 48)**
  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 (p. 185).
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
At least one specified repository name is not valid.

Note
This exception only occurs 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.
AWS CodeCommit API Reference

Example

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

Example

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=AKIAI44QH8DHEXAMPLE/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

{
  "commentsForComparedCommitData": [
  {
  "afterBlobId": "1f330709EXAMPLE",
  "afterCommitId": "317f8570EXAMPLE",
  "beforeBlobId": "80906a4cEXAMPLE",
  "beforeCommitId": "6e147360EXAMPLE",
  "comments": [
  {
  "authorArn": "arn:aws:iam::111111111111: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,
  }
  ]
  }
  ]
}

API Version 2015-04-13

50
See Also

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

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

Returns comments made on a pull request.

Request Syntax

```json
{
    "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 (p. 183).

The request accepts the following data in JSON format.

afterCommitId (p. 52)

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.

Type: String

Required: No

beforeCommitId (p. 52)

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: No

maxResults (p. 52)

A 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 (p. 52)

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

Type: String

Required: No

pullRequestId (p. 52)

The system-generated ID of the pull request. To get this ID, use ListPullRequests (p. 88).

Type: String
Required: Yes
repositoryName (p. 52)
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

Response Syntax

```json
{
    "commentsForPullRequestData": [
        {
            "afterBlobId": "string",
            "afterCommitId": "string",
            "beforeBlobId": "string",
            "beforeCommitId": "string",
            "comments": [
                {
                    "authorArn": "string",
                    "clientRequestToken": "string",
                    "commentId": "string",
                    "content": "string",
                    "creationDate": number,
                    "deleted": boolean,
                    "inReplyTo": "string",
                    "lastModifiedDate": 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 (p. 53)**
An array of comment objects on the pull request.

Type: Array of CommentsForPullRequest (p. 159) objects

**nextToken (p. 53)**
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 (p. 185).

**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

At least one specified repository name is not valid.

Note
This exception only occurs 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. Check to make sure you have provided the correct repository name for the pull request.

HTTP Status Code: 400

Example

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
```
Sample Response

HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 2101

{
    "commentsForPullRequestData": [
        {
            "afterBlobId": "1f330709EXAMPLE",
            "afterCommitId": "5d036259EXAMPLE",
            "beforeBlobId": "80906a4cEXAMPLE",
            "beforeCommitId": "317f8570EXAMPLE",
            "comments": [
                {
                    "authorArn": "arn:aws:iam::111111111111: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
                },
                {
                    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
                    "clientRequestToken": "",
                    "commentId": "442b498bEXAMPLE5756813",
                    "content": "Good catch. I'll remove them.",
                    "creationDate": 1508369829.104,
                    "deleted": false,
                    "commentId": "abcd1234EXAMPLEb5678efgh",
                    "lastModifiedDate": 150836912.273
                }
            ],
            "location": {
                "filePath": "ahs_count.py",
                "filePosition": 367,
                "relativeFileVersion": "AFTER"
            },
            "repositoryName": "MyDemoRepo",
            "pullRequestId": "42"
        },
        {
            "afterCommitId": "5d036259EXAMPLE",
            "beforeCommitId": "317f8570EXAMPLE",
            "maxResults": 250,
            "nextToken": "exampleToken",
            "pullRequestId": 42,
            "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
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
GetCommit

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

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

commitId (p. 58)

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

Type: String

Required: Yes

repositoryName (p. 58)

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

```json
{
   "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 (p. 58)**

A commit data type object that contains information about the specified commit.

Type: Commit (p. 161) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

**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**

At least one specified repository name is not valid.

**Note**
This exception only occurs 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

Example

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",
    }
  }
}
```
See Also

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

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
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 (p. 183).

The request accepts the following data in JSON format.

afterCommitSpecifier (p. 62)

The branch, tag, HEAD, or other fully qualified reference used to identify a commit.

Type: String

Required: Yes

afterPath (p. 62)

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 will be shown for all paths.

Type: String

Required: No

beforeCommitSpecifier (p. 62)

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 prior to the afterCommitSpecifier value will be shown. If you do not use beforeCommitSpecifier in your request, consider limiting the results with maxResults.

Type: String

Required: No

beforePath (p. 62)

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 will be shown for all paths.

Type: String
Required: No

**MaxResults (p. 62)**

A non-negative integer used to limit the number of returned results.

Type: Integer

Required: No

**NextToken (p. 62)**

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

Type: String

Required: No

**repositoryName (p. 62)**

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

```json
{
  "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 (p. 63)**

A differences data type object that contains information about the differences, including whether the difference is added, modified, or deleted (A, D, M).
Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

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
At least one specified repository name is not valid.

Note
This exception only occurs 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

Example

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
See Also

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

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

Returns information about merge conflicts between the before and after commit IDs for a pull request in a repository.

Request Syntax

```json
{
    "destinationCommitSpecifier": "string",
    "mergeOption": "string",
    "repositoryName": "string",
    "sourceCommitSpecifier": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

destinationCommitSpecifier (p. 67)

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

mergeOption (p. 67)

The merge option or strategy you want to use to merge the code. The only valid value is FAST_FORWARD_MERGE.

Type: String

Valid Values: FAST_FORWARD_MERGE

Required: Yes

repositoryName (p. 67)

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 (p. 67)

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

```json
{
  "destinationCommitId": "string",
  "mergeable": boolean,
  "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.

**destinationCommitId (p. 68)**

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

Type: String

**mergeable (p. 68)**

A Boolean value that indicates whether the code is mergable by the specified merge option.

Type: Boolean

**sourceCommitId (p. 68)**

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 (p. 185).

**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
**InvalidDestinationCommitSpecifierException**
The destination commit specifier is not valid. You must provide a valid branch name, tag, or full commit ID.

HTTP Status Code: 400
**InvalidMergeOptionException**
The specified merge option is not valid. The only valid value is FAST_FORWARD_MERGE.

HTTP Status Code: 400
**InvalidRepositoryNameException**
At least one specified repository name is not valid.

**Note**
This exception only occurs 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
**MergeOptionRequiredException**
A merge option or strategy is required, and none was provided.

HTTP Status Code: 400
**RepositoryDoesNotExistException**
The specified repository does not exist.
RepositoryNameRequiredException

A repository name is required but was not specified.

HTTP Status Code: 400

TipsDivergenceExceedededException

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

Example

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: 20171111T224311Z
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=AKIAI44QH8DHEXAMPLE/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": "master",
  "mergeOption": "FAST_FORWARD_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: Sat, 11 Nov 2017 22:59:13 GMT

{
  "destinationCommitId": "fac04518EXAMPLE",
  "mergeable": false,
  "sourceCommitId": "16d097f03EXAMPLE"
}
```

See Also

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

- AWS Command Line Interface
- AWS SDK for .NET
See Also

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

Gets information about a pull request in a specified repository.

Request Syntax

```json
{
    "pullRequestId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

pullRequestId (p. 72)

The system-generated ID of the pull request. To get this ID, use ListPullRequests (p. 88).

Type: String

Required: Yes

Response Syntax

```json
{
    "pullRequest": {
        "authorArn": "string",
        "clientRequestToken": "string",
        "creationDate": number,
        "description": "string",
        "lastActivityDate": number,
        "pullRequestId": "string",
        "pullRequestStatus": "string",
        "pullRequestTargets": [
            {
                "destinationCommit": "string",
                "destinationReference": "string",
                "isMerged": boolean,
                "mergedBy": "string"
            },
            {
                "repositoryName": "string",
                "sourceCommit": "string",
                "sourceReference": "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 (p. 72)

Information about the specified pull request.

Type: PullRequest (p. 166) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

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
Example

Sample Request

```plaintext
>>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: 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": "42"
}
```

Sample Response

```plaintext
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847

{
    "pullRequest": {
        "authorArn": "arn:aws:iam::111111111111:user/Jane_Doe",
        "title": "Pronunciation difficulty analyzer",
        "pullRequestTargets": [
            {
                "destinationReference": "refs/heads/master",
                "destinationCommit": "5d036259EXAMPLE",
                "sourceReference": "refs/heads/jane-branch",
                "sourceCommit": "317f8570EXAMPLE",
                "repositoryName": "MyDemoRepo",
                "mergeMetadata": {
                    "isMerged": false,
                }
            }
        ],
        "lastActivityDate": 1508442444.711,
        "pullRequestId": "42",
        "clientRequestToken": "123Example",
        "pullRequestStatus": "OPEN",
        "creationDate": 1508962823.522,
        "description": "A code review of the new feature I just added to the service."
    }
}
```

See Also

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

- AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
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 web page could 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 web page.

### Request Syntax

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

### Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

**repositoryName (p. 76)**

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

```json
{
    "repositoryMetadata": {
        "accountId": "string",
        "Arn": "string",
        "cloneUrlHttp": "string",
        "cloneUrlSsh": "string",
        "creationDate": number,
        "defaultBranch": "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 (p. 76)

Information about the repository.

Type: RepositoryMetadata (p. 175) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

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

At least one specified repository name is not valid.

Note
This exception only occurs 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
Example

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": "master",
    "repositoryName": "MyDemoRepo",
    "cloneUrlSsh": "ssh://git-codecommit.us-east-1.amazonaws.com/v1/repos/MyDemoRepo",
    "lastModifiedDate": 1430783812.0869999,
    "repositoryDescription": "My demonstration repository",
    "cloneUrlHttp": "https://codecommit.us-east-1.amazonaws.com/v1/repos/MyDemoRepo",
    "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
    "Arn": "arn:aws:codecommit:us-east-1:111111111111:MyDemoRepo",
    "accountId": "111111111111"
  }
}
```

See Also

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

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
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 (p. 183).

The request accepts the following data in JSON format.

**repositoryName (p. 80)**

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 (p. 80)**

The system-generated unique ID for the trigger.
Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

**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**

At least one specified repository name is not valid.

**Note**

This exception only occurs 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
Example

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": [
                "master",
                "MyNewBranch"
            ],
            "customData": "",
            "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
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
See Also

- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
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 (p. 183).

The request accepts the following data in JSON format.

`nextToken` (p. 84)

An enumeration token that allows the operation to batch the results.

Type: String

Required: No

`repositoryName` (p. 84)

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` (p. 84)

The list of branch names.
Type: Array of strings

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

nextToken (p. 84)

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 (p. 185).

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

At least one specified repository name is not valid.

Note
This exception only occurs 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

Example

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": [
        "master",
        "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
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
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**

```json
{
    "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](p. 183).

The request accepts the following data in JSON format.

**authorArn (p. 88)**

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 (p. 88)**

A non-negative integer used to limit the number of returned results.

Type: Integer

Required: No

**nextToken (p. 88)**

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

Type: String

Required: No

**pullRequestStatus (p. 88)**

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 (p. 88)**

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

```json
{
    "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 (p. 89)

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

Type: String

pullRequestIds (p. 89)

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 (p. 185).

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
At least one specified repository name is not valid.

Note
This exception only occurs 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

Example

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

```json
{
   "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
   "maxResults": 20,
   "nextToken": "",
   "pullRequestStatus": "CLOSED",
   "repositoryName": "MyDemoRepo"
}
```

**Sample Response**

```json
{
   "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
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

API Version 2015-04-13
ListRepositories

Gets information about one or more repositories.

Request Syntax

```json
{
  "nextToken": "string",
  "order": "string",
  "sortBy": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

**nextToken (p. 92)**

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 (p. 92)**

The order in which to sort the results of a list repositories operation.

Type: String

Valid Values: ascending | descending

Required: No

**sortBy (p. 92)**

The criteria used to sort the results of a list repositories operation.

Type: String

Valid Values: repositoryName | lastModifiedDate

Required: No

Response Syntax

```json
{
  "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 (p. 92)

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 (p. 92)

Lists the repositories called by the list repositories operation.

Type: Array of RepositoryNameIdPair (p. 177) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

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

Example

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=AKIAI44QH8DBBEXAMPLE/20151028/us-east-1/codecommit/aws4_request, SignedHeaders=content-type;host;user-agent;x-amz-date;x-amz-target, Signature=8d9b5998EXAMPLE

{}
MergePullRequestByFastForward

Closes a pull request and 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 option.

Request Syntax

```json
{
  "pullRequestId": "string",
  "repositoryName": "string",
  "sourceCommitId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

**pullRequestId (p. 95)**

The system-generated ID of the pull request. To get this ID, use ListPullRequests (p. 88).

Type: String

Required: Yes

**repositoryName (p. 95)**

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 (p. 95)**

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

```json
{
  "pullRequest": {
    "authorArn": "string",
    "clientRequestToken": "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 (p. 95)**

Information about the specified pull request, including information about the merge.

Type: PullRequest (p. 166) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

**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**

At least one specified repository name is not valid.

**Note**

This exception only occurs 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

**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

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

Example

Sample Request

```plaintext
>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

```plaintext
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847

{
    "pullRequest": {
        "authorArn": "arn:aws:iam::1111111111: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/master",
                "mergeMetadata": {
                    "isMerged": true,
                    "mergedBy": "arn:aws:iam::1111111111: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
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
PostCommentForComparedCommit

Posts a comment on the comparison between two commits.

Request Syntax

```json
{
   "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 (p. 183).

The request accepts the following data in JSON format.

afterCommitId (p. 100)

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

Type: String

Required: Yes

beforeCommitId (p. 100)

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

Type: String

Required: No

clientRequestToken (p. 100)

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 will return information about the initial request that used that token.

Type: String

Required: No

content (p. 100)

The content of the comment you want to make.

Type: String

Required: Yes
location (p. 100)

The location of the comparison where you want to comment.

Type: Location (p. 164) object

Required: No

repositoryName (p. 100)

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",
        "clientRequestToken": "string",
        "commentId": "string",
        "content": "string",
        "creationDate": number,
        "deleted": boolean,
        "inReplyTo": "string",
        "lastModifiedDate": 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 (p. 101)

In the directionality you established, the blob ID of the 'after' blob.

Type: String

afterCommitId (p. 101)

In the directionality you established, the full commit ID of the 'after' commit.
Type: String

**beforeBlobId (p. 101)**

In the directionality you established, the blob ID of the 'before' blob.

Type: String

**beforeCommitId (p. 101)**

In the directionality you established, the full commit ID of the 'before' commit.

Type: String

**comment (p. 101)**

The content of the comment you posted.

Type: Comment (p. 155) object

**location (p. 101)**

The location of the comment in the comparison between the two commits.

Type: Location (p. 164) object

**repositoryName (p. 101)**

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 (p. 185).

**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 will return 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 1,000 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 re-used.

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 extension of the file as well as the
file name.
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

At least one specified repository name is not valid.

Note

This exception only occurs 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 filePath for a location cannot be empty or null.

HTTP Status Code: 400

PathRequiredException

The filePath for a location cannot be empty or 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

Example

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::111111111111: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
    },
    "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
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
See Also

- AWS SDK for Python
- AWS SDK for Ruby V2
PostCommentForPullRequest

Posts a comment on a pull request.

Request Syntax

```json
{
  "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 (p. 183).

The request accepts the following data in JSON format.

**afterCommitId (p. 107)**

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 (p. 107)**

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 (p. 107)**

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 will return information about the initial request that used that token.

Type: String
Required: No

**content (p. 107)**

The content of your comment on the change.

Type: String
Required: Yes

`location` (p. 107)

The location of the change where you want to post your comment. If no location is provided, the comment will be posted as a general comment on the pull request difference between the before commit ID and the after commit ID.

Type: `Location` (p. 164) object

Required: No

`pullRequestId` (p. 107)

The system-generated ID of the pull request. To get this ID, use `ListPullRequests` (p. 88).

Type: String

Required: Yes

`repositoryName` (p. 107)

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**

```json
{
    "afterBlobId": "string",
    "afterCommitId": "string",
    "beforeBlobId": "string",
    "beforeCommitId": "string",
    "comment": {
        "authorArn": "string",
        "clientRequestToken": "string",
        "commentId": "string",
        "content": "string",
        "creationDate": number,
        "deleted": boolean,
        "inReplyTo": "string",
        "lastModifiedDate": 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 (p. 108)**

In the directionality of the pull request, the blob ID of the 'after' blob.

Type: String

**afterCommitId (p. 108)**

The full commit ID of the commit in the destination branch where the pull request will be merged.

Type: String

**beforeBlobId (p. 108)**

In the directionality of the pull request, the blob ID of the 'before' blob.

Type: String

**beforeCommitId (p. 108)**

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 (p. 108)**

The content of the comment you posted.

Type: Comment (p. 155) object

**location (p. 108)**

The location of the change where you posted your comment.

Type: Location (p. 164) object

**pullRequestId (p. 108)**

The system-generated ID of the pull request.

Type: String

**repositoryName (p. 108)**

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 (p. 185).

**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 will return 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 1,000 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 re-used.
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 extension of the file as well as the file name.

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
At least one specified repository name is not valid.

Note
This exception only occurs 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 filePath for a location cannot be empty or null.
HTTP Status Code: 400
PathRequiredException
The filePath for a location cannot be empty or 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. Check to make sure you have provided the correct repository name for the pull request.

HTTP Status Code: 400

Example

Sample Request

```plaintext
>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-cl1/1.11.187 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",
```
Sample Response

HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847

{
    "afterBlobId": "1f330709EXAMPLE",
    "afterCommitId": "5d036259EXAMPLE",
    "beforeBlobId": "80906a4cEXAMPLE",
    "beforeCommitId": "317f8570EXAMPLE",
    "comment": {
        "authorArn": "arn:aws:iam::11111111111: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
    },
    "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
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
PostCommentReply

Posts a comment in reply to an existing comment on a comparison between commits or a pull request.

Request Syntax

```json
{
    "clientRequestToken": "string",
    "content": "string",
    "inReplyTo": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

**clientRequestToken (p. 114)**

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 will return information about the initial request that used that token.

Type: String

Required: No

**content (p. 114)**

The contents of your reply to a comment.

Type: String

Required: Yes

**inReplyTo (p. 114)**

The system-generated ID of the comment to which you want to reply. To get this ID, use GetCommentsForComparedCommit (p. 47) or GetCommentsForPullRequest (p. 52).

Type: String

Required: Yes

Response Syntax

```json
{
    "comment": {
        "authorArn": "string",
        "clientRequestToken": "string",
        "commentId": "string",
        "content": "string",
        "creationDate": number,
        "deleted": boolean,
        "inReplyTo": "string",
        "lastModifiedDate": 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 (p. 114)

Information about the reply to a comment.

Type: Comment (p. 155) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

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 will return 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 1,000 characters.

HTTP Status Code: 400

CommentDoesNotExistException

No comment exists with the provided ID. Verify that you have provided 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 re-used.

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

### Example

#### 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=AKIAI44QH8DHBEEXAMPLE/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

{
    "comment": {
        "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
        "clientRequestToken": "123Example",
        "commentId": "442b498bEXAMPLE5756813",
        "content": "Good catch. I'll remove them.",
        "creationDate": 1508369829.136,
        "deleted": false,
        "lastModifiedDate": 150836912.221
    }
}
```

### See Also

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

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
PutRepositoryTriggers

Replaces all triggers for a repository. This can be used to create or delete triggers.

Request Syntax

```json
{
  "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 (p. 183).

The request accepts the following data in JSON format.

**repositoryName (p. 118)**

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 (p. 118)**

The JSON block of configuration information for each trigger.

Type: Array of RepositoryTrigger (p. 178) objects

Required: Yes

Response Syntax

```json
{
  "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 (p. 118)**

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 (p. 185).

**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**

At least one specified repository name is not valid.

**Note**

This exception only occurs 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 region for the trigger target does not match the region for the repository. Triggers must be created in the same 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

Example

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": [
    {
      "branches": [
        "mainline",
        "anotherbranch"
      ],
      "events": ["all"],
      "name": "MyFirstTrigger",
      "customData": ""
    },
    {
      "destinationArn": "arn:aws:sns:us-east-1:111111111111EXAMPLE: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
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
TestRepositoryTriggers

Tests the functionality of repository triggers by sending information to the trigger target. If real data is available in the repository, the test will send data from the last commit. If no data is available, sample data will be 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 (p. 183).

The request accepts the following data in JSON format.

**repositoryName (p. 123)**

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 (p. 123)**

The list of triggers to test.

Type: Array of RepositoryTrigger (p. 178) objects

Required: Yes

Response Syntax

```
{
    "failedExecutions": [
        {
            "failureMessage": "string",
            "trigger": "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 (p. 123)

The list of triggers that were not able to be tested. This list provides the names of the triggers that could not be tested, separated by commas.

Type: Array of RepositoryTriggerExecutionFailure (p. 180) objects

successfulExecutions (p. 123)

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 (p. 185).

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

At least one specified repository name is not valid.
Note
This exception only occurs 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 region for the trigger target does not match the region for the repository. Triggers must be created in the same 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

---

**Example**

**Sample Request**

```json
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",
           "branches": [
               "mainline",
               "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
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
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 (p. 183).

The request accepts the following data in JSON format.

**commentId (p. 128)**

The system-generated ID of the comment you want to update. To get this ID, use GetCommentsForComparedCommit (p. 47) or GetCommentsForPullRequest (p. 52).

Type: String

Required: Yes

**content (p. 128)**

The updated content with which you want to replace the existing content of the comment.

Type: String

Required: Yes

Response Syntax

```
{
  "comment": {
    "authorArn": "string",
    "clientRequestToken": "string",
    "commentId": "string",
    "content": "string",
    "creationDate": number,
    "deleted": boolean,
    "inReplyTo": "string",
    "lastModifiedDateTime": 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 (p. 128)

Information about the updated comment.

Type: Comment (p. 155) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

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 1,000 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 provided 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

Example

Sample Request

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

See Also

Sample Response

HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847

{
  "comment": {
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
    "clientRequestToken": "",
    "commentId": "442b498bEXAMPLE5756813",
    "content": "Fixed as requested. I'll update the pull request.",
    "creationDate": 1508369929.783,
    "deleted": false,
    "lastModifiedDate": 1508369929.287
  }
}

See Also

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

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

```json
{
  "defaultBranchName": "string",
  "repositoryName": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

- **defaultBranchName (p. 131)**
  - The name of the branch to set as the default.
  - Type: String
  - Required: Yes

- **repositoryName (p. 131)**
  - The name of the repository to set or change the default branch for.
  - Type: String
  - 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 (p. 185).

- **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**
At least one specified repository name is not valid.

**Note**
This exception only occurs 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.
Example

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

Replaces the contents of the description of a pull request.

Request Syntax

```json
{
  "description": "string",
  "pullRequestId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

**description (p. 134)**

The updated content of the description for the pull request. This content will replace the existing description.

Type: String

Length Constraints: Maximum length of 10240.

Required: Yes

**pullRequestId (p. 134)**

The system-generated ID of the pull request. To get this ID, use ListPullRequests (p. 88).

Type: String

Required: Yes

Response Syntax

```json
{
  "pullRequest": {
    "authorArn": "string",
    "clientRequestToken": "string",
    "creationDate": number,
    "description": "string",
    "lastActivityDate": number,
    "pullRequestId": "string",
    "pullRequestStatus": "string",
    "pullRequestTargets": [ 
      {
        "destinationCommit": "string",
        "destinationReference": "string",
        "mergeMetadata": {
          "isMerged": boolean,
          "mergedBy": "string"
        },
        "repositoryName": "string",
        "sourceCommit": "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 (p. 134)

Information about the updated pull request.

Type: PullRequest (p. 166) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

InvalidDescriptionException

The pull request description is not valid. Descriptions are limited to 1,000 characters in length.

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

Example

Sample Request

>POST / HTTP/1.1
**Sample Response**

```
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847

{
  "pullRequest": {
    "authorArn": "arn:aws:iam::111111111111: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/master",
        "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
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
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 (p. 183).

The request accepts the following data in JSON format.

**pullRequestId (p. 138)**

The system-generated ID of the pull request. To get this ID, use ListPullRequests (p. 88).

- Type: String
- Required: Yes

**pullRequestStatus (p. 138)**

The status of the pull request. The only valid operations are to update the status from OPEN to OPEN, OPEN to CLOSED or from from CLOSED to CLOSED.

- Type: String
- Valid Values: OPEN | CLOSED
- Required: Yes

Response Syntax

```
{
    "pullRequest": {
        "authorArn": "string",
        "clientRequestToken": "string",
        "creationDate": number,
        "description": "string",
        "lastActivityDate": number,
        "pullRequestId": "string",
        "pullRequestStatus": "string",
        "pullRequestTargets": [
            {
                "destinationCommit": "string",
                "destinationReference": "string",
                "mergeMetadata": {
                    "isMerged": boolean,
                    "mergedBy": "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.

pullRequest (p. 138)

Information about the pull request.

Type: PullRequest (p. 166) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

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.

Example

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=AKIAI44QH#DHSEXAMPLE/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

{
  "pullRequest": { 
    "authorArn": "arn:aws:iam::111111111111: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/master",
        "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
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
UpdatePullRequestTitle

Replaces the title of a pull request.

**Request Syntax**

```json
{
   "pullRequestId": "string",
   "title": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

**pullRequestId (p. 142)**

The system-generated ID of the pull request. To get this ID, use ListPullRequests (p. 88).

Type: String

Required: Yes

**title (p. 142)**

The updated title of the pull request. This will replace the existing title.

Type: String

Length Constraints: Maximum length of 150.

Required: Yes

**Response Syntax**

```json
{
   "pullRequest": {
      "authorArn": "string",
      "clientRequestToken": "string",
      "creationDate": number,
      "description": "string",
      "lastActivityDate": number,
      "pullRequestId": "string",
      "pullRequestStatus": "string",
      "pullRequestTargets": [
         {
            "destinationCommit": "string",
            "destinationReference": "string",
            "mergeMetadata": {
               "isMerged": boolean,
               "mergedBy": "string"
            },
            "repositoryName": "string"}]
```

API Version 2015-04-13

142
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 (p. 142)**

Information about the updated pull request.

Type: **PullRequest (p. 166) object**

Errors

For information about the errors that are common to all actions, see Common Errors (p. 185).

**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
Example

Sample Request

```plaintext
>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

```plaintext
HTTP/1.1 200 OK
x-amzn-RequestId: 0728aaa8-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 847

{
  "pullRequest": {
    "authorArn": "arn:aws:iam::111111111111: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/master",
      "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
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
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 web page could 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 web page.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

**repositoryDescription (p. 146)**

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 (p. 146)**

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 (p. 185).
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

At least one specified repository name is not valid.

Note
This exception only occurs 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

Example

Sample Request

POST / HTTP/1.1
```json
{
    "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
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
UpdateRepositoryName

Renames a repository. The repository name must be unique across the calling AWS account. In addition, repository names are limited to 100 alphanumeric, dash, and underscore characters, and cannot include certain characters. The suffix ".git" is prohibited. For a full description of the limits on repository names, see Limits in the AWS CodeCommit User Guide.

Request Syntax

```json
{
   "newName": "string",
   "oldName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 183).

The request accepts the following data in JSON format.

**newName (p. 149)**

The new name for the repository.

Type: String

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

Pattern: \[\w\.-]+

Required: Yes

**oldName (p. 149)**

The existing 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 (p. 185).

**InvalidRepositoryNameException**

At least one specified repository name is not valid.
Note
This exception only occurs 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.

Example

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
See Also

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

- BlobMetadata (p. 153)
- BranchInfo (p. 154)
- Comment (p. 155)
- CommentsForComparedCommit (p. 157)
- CommentsForPullRequest (p. 159)
- Commit (p. 161)
- Difference (p. 163)
- Location (p. 164)
- MergeMetadata (p. 165)
- PullRequest (p. 166)
- PullRequestEvent (p. 168)
- PullRequestMergedStateChangedEventMetadata (p. 170)
- PullRequestSourceReferenceUpdatedEventMetadata (p. 171)
- PullRequestStatusChangedEventMetadata (p. 172)
- PullRequestTarget (p. 173)
- RepositoryMetadata (p. 175)
- RepositoryNameIdPair (p. 177)
- RepositoryTrigger (p. 178)
- RepositoryTriggerExecutionFailure (p. 180)
- Target (p. 181)
- UserInfo (p. 182)
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 any 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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
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

**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 will return 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

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
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 (p. 155) 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 (p. 164) object
  - Required: No

- **repositoryName**
  - The name of the repository that contains the compared commits.
  - Type: String
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 Go  
  • AWS SDK for Java  
  • AWS SDK for Ruby V2
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 will be 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 (p. 155) 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 (p. 164) 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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
Commit

Returns information about a specific commit.

Contents

**additionalData**

Any additional 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 (p. 182)` object

Required: No

**commitId**

The full SHA 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 (p. 182)` object

Required: No

**message**

The commit message associated with the specified commit.

Type: String

Required: No

**parents**

The parent list for the specified commit.

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
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 (p. 153) object

Required: No

beforeBlob

Information about a beforeBlob data type object, including the ID, the file mode permission code, and the path.

Type: BlobMetadata (p. 153) 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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
**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 within 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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
**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

**mergedBy**

The Amazon Resource Name (ARN) of the user who merged the branches.

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
PullRequest

Returns information about a pull request.

Contents

**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 will return 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 (p. 173) objects

Required: No

title

The user-defined title of the pull request. This title is displayed in the list of pull requests to other users of the repository.

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
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 additional commits or changing the status of a pull request.

Type: String

Required: No

eventDate

The day and time of the pull request event, in timestamp format.

Type: Timestamp

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

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 (p. 170) object

Required: No

pullRequestSourceReferenceUpdatedEventMetadata

Information about the updated source branch for the pull request event.

Type: PullRequestSourceReferenceUpdatedEventMetadata (p. 171) object

Required: No

pullRequestStatusChangedEventMetadata

Information about the change in status for the pull request event.
Type: `PullRequestStatusChangedEventMetadata (p. 172)` 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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
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 will be merged into.

Type: String
Length Constraints: Minimum length of 1. Maximum length of 256.
Required: No

mergeMetadata

Information about the merge state change event.

Type: MergeMetadata (p. 165) 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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
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

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
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 will be merged into. Also known as the destination branch.

Type: String
Required: No

mergeMetadata

Returns metadata about the state of the merge, including whether the merge has been made.

Type: MergeMetadata (p. 165) 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 will change 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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
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

**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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
RepositoryTrigger

Information about a trigger for a repository.

Contents

branches

The branches that will be included in the trigger configuration. If you specify an empty array, the trigger will apply to all branches.

Note
While 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 that will be included in the information sent to the target of the trigger.

Type: String
Required: No

destinationArn

The ARN of the resource that is the target for a trigger. For example, the ARN of a topic in Amazon Simple Notification Service (SNS).

Type: String
Required: No

events

The repository events that will cause the trigger to run actions in another service, such as sending a notification through Amazon Simple Notification Service (SNS).

Note
The valid value “all” cannot be used with any other values.

Type: Array of strings
Valid Values: all | updateReference | createReference | deleteReference
Required: No

name

The name 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:
See Also

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
RepositoryTriggerExecutionFailure

A trigger failed to run.

Contents

failureMessage

Additional 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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
Target

Returns information about a target for a pull request.

Contents

destinationReference

The branch of the repository where the pull request changes will be merged into. Also known as the destination branch.

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: 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

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
UserInfo

Information about the user who made a specified commit.

Contents

date
The date when the specified commit was pushed to the repository.
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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
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 Signature Version 4 Signing Process in the Amazon Web Services General Reference.

**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 Task 2: Create a String to Sign for Signature Version 4 in the Amazon Web Services General Reference.

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 Handling Dates in Signature Version 4 in the Amazon Web Services General Reference.

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 Security Token Service, go to AWS Services That Work with IAM in the IAM User Guide.

Condition: If you're using temporary security credentials from the AWS Security Token Service, 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 Task 1: Create a Canonical Request For Signature Version 4 in the Amazon Web Services General Reference.

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

InvalidParameterCombination

Parameters that must not be used together were used together.

HTTP Status Code: 400

InvalidParameterValue

An invalid or out-of-range value was supplied for the input parameter.

HTTP Status Code: 400

InvalidQueryParameter

The AWS query string is malformed or does not adhere to AWS standards.

HTTP Status Code: 400

MalformedQueryString

The query string contains a syntax error.

HTTP Status Code: 404

MissingAction

The request is missing an action or a required parameter.

HTTP Status Code: 400
**MissingAuthenticationToken**

The request must contain either a valid (registered) AWS access key ID or X.509 certificate.

HTTP Status Code: 403

**MissingParameter**

A required parameter for the specified action is not supplied.

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