
Amazon Honeycode

API Reference

API Version 2020-03-01



Amazon Honeycode: API Reference

Copyright © Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Welcome	1
Actions	2
BatchCreateTableRows	3
Request Syntax	3
URI Request Parameters	3
Request Body	4
Response Syntax	4
Response Elements	4
Errors	5
See Also	6
BatchDeleteTableRows	7
Request Syntax	7
URI Request Parameters	7
Request Body	7
Response Syntax	8
Response Elements	8
Errors	9
See Also	9
BatchUpdateTableRows	11
Request Syntax	11
URI Request Parameters	11
Request Body	11
Response Syntax	12
Response Elements	12
Errors	13
See Also	13
BatchUpsertTableRows	15
Request Syntax	15
URI Request Parameters	15
Request Body	16
Response Syntax	16
Response Elements	17
Errors	17
See Also	18
DescribeTableDataImportJob	19
Request Syntax	19
URI Request Parameters	19
Request Body	19
Response Syntax	19
Response Elements	20
Errors	20
See Also	21
GetScreenData	22
Request Syntax	22
URI Request Parameters	22
Request Body	22
Response Syntax	23
Response Elements	24
Errors	24
See Also	25
InvokeScreenAutomation	26
Request Syntax	26
URI Request Parameters	26
Request Body	27

Response Syntax	27
Response Elements	28
Errors	28
See Also	29
ListTableColumns	30
Request Syntax	30
URI Request Parameters	30
Request Body	30
Response Syntax	30
Response Elements	31
Errors	31
See Also	32
ListTableRows	33
Request Syntax	33
URI Request Parameters	33
Request Body	33
Response Syntax	34
Response Elements	34
Errors	35
See Also	36
ListTables	37
Request Syntax	37
URI Request Parameters	37
Request Body	37
Response Syntax	37
Response Elements	38
Errors	38
See Also	39
QueryTableRows	40
Request Syntax	40
URI Request Parameters	40
Request Body	40
Response Syntax	41
Response Elements	41
Errors	42
See Also	43
StartTableDataImportJob	44
Request Syntax	44
URI Request Parameters	44
Request Body	45
Response Syntax	45
Response Elements	46
Errors	46
See Also	47
Data Types	48
Cell	49
Contents	49
See Also	50
CellInput	51
Contents	51
See Also	51
ColumnMetadata	52
Contents	52
See Also	52
CreateRowData	53
Contents	53
See Also	53

Dataltem	54
Contents	54
See Also	54
DelimitedTextImportOptions	55
Contents	55
See Also	55
DestinationOptions	56
Contents	56
See Also	56
FailedBatchItem	57
Contents	57
See Also	57
Filter	58
Contents	58
See Also	58
ImportDataSource	59
Contents	59
See Also	59
ImportDataSourceConfig	60
Contents	60
See Also	60
ImportJobSubmitter	61
Contents	61
See Also	61
ImportOptions	62
Contents	62
See Also	62
ResultRow	63
Contents	63
See Also	63
ResultSet	64
Contents	64
See Also	64
SourceDataColumnProperties	65
Contents	65
See Also	65
Table	66
Contents	66
See Also	66
TableColumn	67
Contents	67
See Also	67
TableDataImportJobMetadata	68
Contents	68
See Also	68
TableRow	69
Contents	69
See Also	69
UpdateRowData	70
Contents	70
See Also	70
UpsertRowData	71
Contents	71
See Also	71
UpsertRowsResult	73
Contents	73
See Also	73

VariableValue	74
Contents	74
See Also	74
Common Parameters	75
Common Errors	77

Welcome

Amazon Honeycode is a fully managed service that allows you to quickly build mobile and web apps for teams—without programming. Build Honeycode apps for managing almost anything, like projects, customers, operations, approvals, resources, and even your team.

This document was last published on April 19, 2021.

Actions

The following actions are supported:

- [BatchCreateTableRows](#) (p. 3)
- [BatchDeleteTableRows](#) (p. 7)
- [BatchUpdateTableRows](#) (p. 11)
- [BatchUpsertTableRows](#) (p. 15)
- [DescribeTableDataImportJob](#) (p. 19)
- [GetScreenData](#) (p. 22)
- [InvokeScreenAutomation](#) (p. 26)
- [ListTableColumns](#) (p. 30)
- [ListTableRows](#) (p. 33)
- [ListTables](#) (p. 37)
- [QueryTableRows](#) (p. 40)
- [StartTableDataImportJob](#) (p. 44)

BatchCreateTableRows

The BatchCreateTableRows API allows you to create one or more rows at the end of a table in a workbook. The API allows you to specify the values to set in some or all of the columns in the new rows.

If a column is not explicitly set in a specific row, then the column level formula specified in the table will be applied to the new row. If there is no column level formula but the last row of the table has a formula, then that formula will be copied down to the new row. If there is no column level formula and no formula in the last row of the table, then that column will be left blank for the new rows.

Request Syntax

```
POST /workbooks/workbookId/tables/tableId/rows/batchcreate HTTP/1.1  
Content-type: application/json
```

```
{  
  "clientRequestToken": "string",  
  "rowsToCreate": [  
    {  
      "batchItemId": "string",  
      "cellsToCreate": {  
        "string": {  
          "fact": "string"  
        }  
      }  
    }  
  ]  
}
```

URI Request Parameters

The request uses the following URI parameters.

[tableId \(p. 3\)](#)

The ID of the table where the new rows are being added.

If a table with the specified ID could not be found, this API throws `ResourceNotFoundException`.

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: Yes

[workbookId \(p. 3\)](#)

The ID of the workbook where the new rows are being added.

If a workbook with the specified ID could not be found, this API throws `ResourceNotFoundException`.

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: Yes

Request Body

The request accepts the following data in JSON format.

[clientRequestToken \(p. 3\)](#)

The request token for performing the batch create operation. Request tokens help to identify duplicate requests. If a call times out or fails due to a transient error like a failed network connection, you can retry the call with the same request token. The service ensures that if the first call using that request token is successfully performed, the second call will not perform the operation again.

Note that request tokens are valid only for a few minutes. You cannot use request tokens to dedupe requests spanning hours or days.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Pattern: `^(?!\\s*$).+`

Required: No

[rowsToCreate \(p. 3\)](#)

The list of rows to create at the end of the table. Each item in this list needs to have a batch item id to uniquely identify the element in the request and the cells to create for that row. You need to specify at least one item in this list.

Note that if one of the column ids in any of the rows in the request does not exist in the table, then the request fails and no updates are made to the table.

Type: Array of [CreateRowData \(p. 53\)](#) objects

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Required: Yes

Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "createdRows": {
    "string": "string"
  },
  "failedBatchItems": [
    {
      "errorMessage": "string",
      "id": "string"
    }
  ],
  "workbookCursor": 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.

createdRows (p. 4)

The map of batch item id to the row id that was created for that item.

Type: String to string map

Key Length Constraints: Minimum length of 1. Maximum length of 64.

Key Pattern: `^(?!\\s*$).+`

Value Length Constraints: Fixed length of 77.

Value Pattern: `row:[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}\/[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

failedBatchItems (p. 4)

The list of batch items in the request that could not be added to the table. Each element in this list contains one item from the request that could not be added to the table along with the reason why that item could not be added.

Type: Array of [FailedBatchItem \(p. 57\)](#) objects

Array Members: Minimum number of 0 items. Maximum number of 100 items.

workbookCursor (p. 4)

The updated workbook cursor after adding the new rows at the end of the table.

Type: Long

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 77\)](#).

AccessDeniedException

You do not have sufficient access to perform this action. Check that the workbook is owned by you and your IAM policy allows access to the resource in the request.

HTTP Status Code: 403

InternalServerErrorException

There were unexpected errors from the server.

HTTP Status Code: 500

RequestTimeoutException

The request timed out.

HTTP Status Code: 504

ResourceNotFoundException

A Workbook, Table, App, Screen or Screen Automation was not found with the given ID.

HTTP Status Code: 404

ServiceQuotaExceededException

The request caused service quota to be breached.

HTTP Status Code: 402

ServiceUnavailableException

Remote service is unreachable.

HTTP Status Code: 503

ThrottlingException

Tps(transactions per second) rate reached.

HTTP Status Code: 429

ValidationException

Request is invalid. The message in the response contains details on why the request is invalid.

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 V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

BatchDeleteTableRows

The BatchDeleteTableRows API allows you to delete one or more rows from a table in a workbook. You need to specify the ids of the rows that you want to delete from the table.

Request Syntax

```
POST /workbooks/workbookId/tables/tableId/rows/batchdelete HTTP/1.1
Content-type: application/json

{
  "clientRequestToken": "string",
  "rowIds": [ "string" ]
}
```

URI Request Parameters

The request uses the following URI parameters.

tableId (p. 7)

The ID of the table where the rows are being deleted.

If a table with the specified id could not be found, this API throws ResourceNotFoundException.

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: Yes

workbookId (p. 7)

The ID of the workbook where the rows are being deleted.

If a workbook with the specified id could not be found, this API throws ResourceNotFoundException.

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: Yes

Request Body

The request accepts the following data in JSON format.

clientRequestToken (p. 7)

The request token for performing the delete action. Request tokens help to identify duplicate requests. If a call times out or fails due to a transient error like a failed network connection, you can retry the call with the same request token. The service ensures that if the first call using that request token is successfully performed, the second call will not perform the action again.

Note that request tokens are valid only for a few minutes. You cannot use request tokens to dedupe requests spanning hours or days.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Pattern: `^(?!\\s*$).+`

Required: No

[rowIds \(p. 7\)](#)

The list of row ids to delete from the table. You need to specify at least one row id in this list.

Note that if one of the row ids provided in the request does not exist in the table, then the request fails and no rows are deleted from the table.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Fixed length of 77.

Pattern: `row:[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}\\/[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: Yes

Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "failedBatchItems": [
    {
      "errorMessage": "string",
      "id": "string"
    }
  ],
  "workbookCursor": 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.

[failedBatchItems \(p. 8\)](#)

The list of row ids in the request that could not be deleted from the table. Each element in this list contains one row id from the request that could not be deleted along with the reason why that item could not be deleted.

Type: Array of [FailedBatchItem \(p. 57\)](#) objects

Array Members: Minimum number of 0 items. Maximum number of 100 items.

[workbookCursor \(p. 8\)](#)

The updated workbook cursor after deleting the rows from the table.

Type: Long

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 77\)](#).

AccessDeniedException

You do not have sufficient access to perform this action. Check that the workbook is owned by you and your IAM policy allows access to the resource in the request.

HTTP Status Code: 403

InternalServerErrorException

There were unexpected errors from the server.

HTTP Status Code: 500

RequestTimeoutException

The request timed out.

HTTP Status Code: 504

ResourceNotFoundException

A Workbook, Table, App, Screen or Screen Automation was not found with the given ID.

HTTP Status Code: 404

ServiceUnavailableException

Remote service is unreachable.

HTTP Status Code: 503

ThrottlingException

Tps(transactions per second) rate reached.

HTTP Status Code: 429

ValidationException

Request is invalid. The message in the response contains details on why the request is invalid.

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 V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

BatchUpdateTableRows

The BatchUpdateTableRows API allows you to update one or more rows in a table in a workbook.

You can specify the values to set in some or all of the columns in the table for the specified rows. If a column is not explicitly specified in a particular row, then that column will not be updated for that row. To clear out the data in a specific cell, you need to set the value as an empty string ("").

Request Syntax

```
POST /workbooks/workbookId/tables/tableId/rows/batchupdate HTTP/1.1
Content-type: application/json

{
  "clientRequestToken": "string",
  "rowsToUpdate": [
    {
      "cellsToUpdate": {
        "string": {
          "fact": "string"
        }
      },
      "rowId": "string"
    }
  ]
}
```

URI Request Parameters

The request uses the following URI parameters.

tableId (p. 11)

The ID of the table where the rows are being updated.

If a table with the specified id could not be found, this API throws ResourceNotFoundException.

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: Yes

workbookId (p. 11)

The ID of the workbook where the rows are being updated.

If a workbook with the specified id could not be found, this API throws ResourceNotFoundException.

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: Yes

Request Body

The request accepts the following data in JSON format.

[clientRequestToken \(p. 11\)](#)

The request token for performing the update action. Request tokens help to identify duplicate requests. If a call times out or fails due to a transient error like a failed network connection, you can retry the call with the same request token. The service ensures that if the first call using that request token is successfully performed, the second call will not perform the action again.

Note that request tokens are valid only for a few minutes. You cannot use request tokens to dedupe requests spanning hours or days.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Pattern: `^(?!\\s*$).+`

Required: No

[rowsToUpdate \(p. 11\)](#)

The list of rows to update in the table. Each item in this list needs to contain the row id to update along with the map of column id to cell values for each column in that row that needs to be updated. You need to specify at least one row in this list, and for each row, you need to specify at least one column to update.

Note that if one of the row or column ids in the request does not exist in the table, then the request fails and no updates are made to the table.

Type: Array of [UpdateRowData \(p. 70\)](#) objects

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Required: Yes

Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "failedBatchItems": [
    {
      "errorMessage": "string",
      "id": "string"
    }
  ],
  "workbookCursor": 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.

[failedBatchItems \(p. 12\)](#)

The list of batch items in the request that could not be updated in the table. Each element in this list contains one item from the request that could not be updated in the table along with the reason why that item could not be updated.

Type: Array of [FailedBatchItem \(p. 57\)](#) objects

Array Members: Minimum number of 0 items. Maximum number of 100 items.

[workbookCursor \(p. 12\)](#)

The updated workbook cursor after adding the new rows at the end of the table.

Type: Long

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 77\)](#).

AccessDeniedException

You do not have sufficient access to perform this action. Check that the workbook is owned by you and your IAM policy allows access to the resource in the request.

HTTP Status Code: 403

InternalServerErrorException

There were unexpected errors from the server.

HTTP Status Code: 500

RequestTimeoutException

The request timed out.

HTTP Status Code: 504

ResourceNotFoundException

A Workbook, Table, App, Screen or Screen Automation was not found with the given ID.

HTTP Status Code: 404

ServiceUnavailableException

Remote service is unreachable.

HTTP Status Code: 503

ThrottlingException

Tps(transactions per second) rate reached.

HTTP Status Code: 429

ValidationException

Request is invalid. The message in the response contains details on why the request is invalid.

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 V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

BatchUpsertTableRows

The BatchUpsertTableRows API allows you to upsert one or more rows in a table. The upsert operation takes a filter expression as input and evaluates it to find matching rows on the destination table. If matching rows are found, it will update the cells in the matching rows to new values specified in the request. If no matching rows are found, a new row is added at the end of the table and the cells in that row are set to the new values specified in the request.

You can specify the values to set in some or all of the columns in the table for the matching or newly appended rows. If a column is not explicitly specified for a particular row, then that column will not be updated for that row. To clear out the data in a specific cell, you need to set the value as an empty string ("").

Request Syntax

```
POST /workbooks/workbookId/tables/tableId/rows/batchupsert HTTP/1.1  
Content-type: application/json
```

```
{  
  "clientRequestToken": "string",  
  "rowsToUpsert": [  
    {  
      "batchItemId": "string",  
      "cellsToUpdate": {  
        "string": {  
          "fact": "string"  
        }  
      },  
      "filter": {  
        "contextRowId": "string",  
        "formula": "string"  
      }  
    }  
  ]  
}
```

URI Request Parameters

The request uses the following URI parameters.

tableId (p. 15)

The ID of the table where the rows are being upserted.

If a table with the specified id could not be found, this API throws ResourceNotFoundException.

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: Yes

workbookId (p. 15)

The ID of the workbook where the rows are being upserted.

If a workbook with the specified id could not be found, this API throws ResourceNotFoundException.

Length Constraints: Fixed length of 36.

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: Yes

Request Body

The request accepts the following data in JSON format.

[clientRequestToken \(p. 15\)](#)

The request token for performing the update action. Request tokens help to identify duplicate requests. If a call times out or fails due to a transient error like a failed network connection, you can retry the call with the same request token. The service ensures that if the first call using that request token is successfully performed, the second call will not perform the action again.

Note that request tokens are valid only for a few minutes. You cannot use request tokens to dedupe requests spanning hours or days.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Pattern: `^(?!\\s*$).+`

Required: No

[rowsToUpsert \(p. 15\)](#)

The list of rows to upsert in the table. Each item in this list needs to have a batch item id to uniquely identify the element in the request, a filter expression to find the rows to update for that element and the cell values to set for each column in the upserted rows. You need to specify at least one item in this list.

Note that if one of the filter formulas in the request fails to evaluate because of an error or one of the column ids in any of the rows does not exist in the table, then the request fails and no updates are made to the table.

Type: Array of [UpsertRowData \(p. 71\)](#) objects

Required: Yes

Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "failedBatchItems": [
    {
      "errorMessage": "string",
      "id": "string"
    }
  ],
  "rows": {
    "string" : {
```

```
    "rowIds": [ "string" ],  
    "upsertAction": "string"  
  }  
},  
"workbookCursor": 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.

failedBatchItems (p. 16)

The list of batch items in the request that could not be updated or appended in the table. Each element in this list contains one item from the request that could not be updated in the table along with the reason why that item could not be updated or appended.

Type: Array of [FailedBatchItem](#) (p. 57) objects

Array Members: Minimum number of 0 items. Maximum number of 100 items.

rows (p. 16)

A map with the batch item id as the key and the result of the upsert operation as the value. The result of the upsert operation specifies whether existing rows were updated or a new row was appended, along with the list of row ids that were affected.

Type: String to [UpsertRowsResult](#) (p. 73) object map

Key Length Constraints: Minimum length of 1. Maximum length of 64.

Key Pattern: `^(?!\\s*$).+`

workbookCursor (p. 16)

The updated workbook cursor after updating or appending rows in the table.

Type: Long

Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 77).

AccessDeniedException

You do not have sufficient access to perform this action. Check that the workbook is owned by you and your IAM policy allows access to the resource in the request.

HTTP Status Code: 403

InternalServerErrorException

There were unexpected errors from the server.

HTTP Status Code: 500

RequestTimeoutException

The request timed out.

HTTP Status Code: 504

ResourceNotFoundException

A Workbook, Table, App, Screen or Screen Automation was not found with the given ID.

HTTP Status Code: 404

ServiceQuotaExceededException

The request caused service quota to be breached.

HTTP Status Code: 402

ServiceUnavailableException

Remote service is unreachable.

HTTP Status Code: 503

ThrottlingException

Tps(transactions per second) rate reached.

HTTP Status Code: 429

ValidationException

Request is invalid. The message in the response contains details on why the request is invalid.

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 V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

DescribeTableDataImportJob

The DescribeTableDataImportJob API allows you to retrieve the status and details of a table data import job.

Request Syntax

```
GET /workbooks/workbookId/tables/tableId/import/jobId HTTP/1.1
```

URI Request Parameters

The request uses the following URI parameters.

jobId (p. 19)

The ID of the job that was returned by the StartTableDataImportJob request.

If a job with the specified id could not be found, this API throws ResourceNotFoundException.

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

Pattern: `^[^\n\r\x00\x08\x0B\x0C\x0E\x1F]*$`

Required: Yes

tableId (p. 19)

The ID of the table into which data was imported.

If a table with the specified id could not be found, this API throws ResourceNotFoundException.

Length Constraints: Fixed length of 36.

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: Yes

workbookId (p. 19)

The ID of the workbook into which data was imported.

If a workbook with the specified id could not be found, this API throws ResourceNotFoundException.

Length Constraints: Fixed length of 36.

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
```

```
Content-type: application/json

{
  "jobMetadata": {
    "dataSource": {
      "dataSourceConfig": {
        "dataSourceUrl": "string"
      }
    },
    "importOptions": {
      "delimitedTextOptions": {
        "dataCharacterEncoding": "string",
        "delimiter": "string",
        "hasHeaderRow": boolean,
        "ignoreEmptyRows": boolean
      },
      "destinationOptions": {
        "columnMap": {
          "string": {
            "columnIndex": number
          }
        }
      }
    },
    "submitter": {
      "email": "string",
      "userArn": "string"
    },
    "submitTime": number
  },
  "jobStatus": "string",
  "message": "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.

jobMetadata (p. 19)

The metadata about the job that was submitted for import.

Type: [TableDataImportJobMetadata](#) (p. 68) object

jobStatus (p. 19)

The current status of the import job.

Type: String

Valid Values: SUBMITTED | IN_PROGRESS | COMPLETED | FAILED

message (p. 19)

A message providing more details about the current status of the import job.

Type: String

Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 77).

AccessDeniedException

You do not have sufficient access to perform this action. Check that the workbook is owned by you and your IAM policy allows access to the resource in the request.

HTTP Status Code: 403

InternalServerErrorException

There were unexpected errors from the server.

HTTP Status Code: 500

ResourceNotFoundException

A Workbook, Table, App, Screen or Screen Automation was not found with the given ID.

HTTP Status Code: 404

ServiceUnavailableException

Remote service is unreachable.

HTTP Status Code: 503

ThrottlingException

Tps(transactions per second) rate reached.

HTTP Status Code: 429

ValidationException

Request is invalid. The message in the response contains details on why the request is invalid.

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 V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

GetScreenData

The GetScreenData API allows retrieval of data from a screen in a Honeycode app. The API allows setting local variables in the screen to filter, sort or otherwise affect what will be displayed on the screen.

Request Syntax

```
POST /screendata HTTP/1.1
Content-type: application/json

{
  "appId": "string",
  "maxResults": number,
  "nextToken": "string",
  "screenId": "string",
  "variables": {
    "string": {
      "rawValue": "string"
    }
  },
  "workbookId": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

appId (p. 22)

The ID of the app that contains the screen.

Type: String

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: Yes

maxResults (p. 22)

The number of results to be returned on a single page. Specify a number between 1 and 100. The maximum value is 100.

This parameter is optional. If you don't specify this parameter, the default page size is 100.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 100.

Required: No

nextToken (p. 22)

This parameter is optional. If a nextToken is not specified, the API returns the first page of data.

Pagination tokens expire after 1 hour. If you use a token that was returned more than an hour back, the API will throw `ValidationException`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `^(?!\\s*$).+`

Required: No

[screenId \(p. 22\)](#)

The ID of the screen.

Type: String

Length Constraints: Fixed length of 36.

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: Yes

[variables \(p. 22\)](#)

Variables are optional and are needed only if the screen requires them to render correctly. Variables are specified as a map where the key is the name of the variable as defined on the screen. The value is an object which currently has only one property, `rawValue`, which holds the value of the variable to be passed to the screen.

Type: String to [VariableValue \(p. 74\)](#) object map

Key Pattern: `^(?!\\s*$).+`

Required: No

[workbookId \(p. 22\)](#)

The ID of the workbook that contains the screen.

Type: String

Length Constraints: Fixed length of 36.

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: Yes

Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "results": {
    "string": {
      "headers": [
        {
          "format": "string",
```

```
    "name": "string"
  }
],
"rows": [
  {
    "dataItems": [
      {
        "formattedValue": "string",
        "overrideFormat": "string",
        "rawValue": "string"
      }
    ],
    "rowId": "string"
  }
]
},
"workbookCursor": 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.

nextToken (p. 23)

Provides the pagination token to load the next page if there are more results matching the request. If a pagination token is not present in the response, it means that all data matching the query has been loaded.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `^(?!\\s*$).+`

results (p. 23)

A map of all the rows on the screen keyed by block name.

Type: String to [ResultSet](#) (p. 64) object map

workbookCursor (p. 23)

Indicates the cursor of the workbook at which the data returned by this workbook is read. Workbook cursor keeps increasing with every update and the increments are not sequential.

Type: Long

Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 77).

AccessDeniedException

You do not have sufficient access to perform this action. Check that the workbook is owned by you and your IAM policy allows access to the resource in the request.

HTTP Status Code: 403

InternalServerErrorException

There were unexpected errors from the server.

HTTP Status Code: 500

RequestTimeoutException

The request timed out.

HTTP Status Code: 504

ResourceNotFoundException

A Workbook, Table, App, Screen or Screen Automation was not found with the given ID.

HTTP Status Code: 404

ServiceUnavailableException

Remote service is unreachable.

HTTP Status Code: 503

ThrottlingException

Tps(transactions per second) rate reached.

HTTP Status Code: 429

ValidationException

Request is invalid. The message in the response contains details on why the request is invalid.

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 V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

InvokeScreenAutomation

The InvokeScreenAutomation API allows invoking an action defined in a screen in a Honeycode app. The API allows setting local variables, which can then be used in the automation being invoked. This allows automating the Honeycode app interactions to write, update or delete data in the workbook.

Request Syntax

```
POST /workbooks/workbookId/apps/appId/screens/screenId/automations/automationId HTTP/1.1
Content-type: application/json

{
  "clientRequestToken": "string",
  "rowId": "string",
  "variables": {
    "string": {
      "rawValue": "string"
    }
  }
}
```

URI Request Parameters

The request uses the following URI parameters.

appId (p. 26)

The ID of the app that contains the screen automation.

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: Yes

automationId (p. 26)

The ID of the automation action to be performed.

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: Yes

screenId (p. 26)

The ID of the screen that contains the screen automation.

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: Yes

workbookId (p. 26)

The ID of the workbook that contains the screen automation.

Length Constraints: Fixed length of 36.

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: Yes

Request Body

The request accepts the following data in JSON format.

[clientRequestToken \(p. 26\)](#)

The request token for performing the automation action. Request tokens help to identify duplicate requests. If a call times out or fails due to a transient error like a failed network connection, you can retry the call with the same request token. The service ensures that if the first call using that request token is successfully performed, the second call will return the response of the previous call rather than performing the action again.

Note that request tokens are valid only for a few minutes. You cannot use request tokens to dedupe requests spanning hours or days.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Pattern: `^(?!\\s*$).+`

Required: No

[rowId \(p. 26\)](#)

The row ID for the automation if the automation is defined inside a block with source or list.

Type: String

Length Constraints: Fixed length of 77.

Pattern: `row:[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}\/[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: No

[variables \(p. 26\)](#)

Variables are specified as a map where the key is the name of the variable as defined on the screen. The value is an object which currently has only one property, `rawValue`, which holds the value of the variable to be passed to the screen. Any variables defined in a screen are required to be passed in the call.

Type: String to [VariableValue \(p. 74\)](#) object map

Key Pattern: `^(?!\\s*$).+`

Required: No

Response Syntax

`HTTP/1.1 200`

```
Content-type: application/json  
  
{  
  "workbookCursor": 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.

[workbookCursor \(p. 27\)](#)

The updated workbook cursor after performing the automation action.

Type: Long

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 77\)](#).

AccessDeniedException

You do not have sufficient access to perform this action. Check that the workbook is owned by you and your IAM policy allows access to the resource in the request.

HTTP Status Code: 403

AutomationExecutionException

The automation execution did not end successfully.

HTTP Status Code: 400

AutomationExecutionTimeoutException

The automation execution timed out.

HTTP Status Code: 504

InternalServerErrorException

There were unexpected errors from the server.

HTTP Status Code: 500

RequestTimeoutException

The request timed out.

HTTP Status Code: 504

ResourceNotFoundException

A Workbook, Table, App, Screen or Screen Automation was not found with the given ID.

HTTP Status Code: 404

ServiceUnavailableException

Remote service is unreachable.

HTTP Status Code: 503

ThrottlingException

Tps(transactions per second) rate reached.

HTTP Status Code: 429

ValidationException

Request is invalid. The message in the response contains details on why the request is invalid.

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 V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

ListTableColumns

The ListTableColumns API allows you to retrieve a list of all the columns in a table in a workbook.

Request Syntax

```
GET /workbooks/workbookId/tables/tableId/columns?nextToken=nextToken HTTP/1.1
```

URI Request Parameters

The request uses the following URI parameters.

nextToken (p. 30)

This parameter is optional. If a nextToken is not specified, the API returns the first page of data.

Pagination tokens expire after 1 hour. If you use a token that was returned more than an hour back, the API will throw ValidationException.

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `^(?!\\s*$).+`

tableId (p. 30)

The ID of the table whose columns are being retrieved.

If a table with the specified id could not be found, this API throws ResourceNotFoundException.

Length Constraints: Fixed length of 36.

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: Yes

workbookId (p. 30)

The ID of the workbook that contains the table whose columns are being retrieved.

If a workbook with the specified id could not be found, this API throws ResourceNotFoundException.

Length Constraints: Fixed length of 36.

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
```

```
Content-type: application/json

{
  "nextToken": "string",
  "tableColumns": [
    {
      "format": "string",
      "tableColumnId": "string",
      "tableColumnName": "string"
    }
  ],
  "workbookCursor": 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.

nextToken (p. 30)

Provides the pagination token to load the next page if there are more results matching the request. If a pagination token is not present in the response, it means that all data matching the request has been loaded.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `^(?!\\s*$).+`

tableColumns (p. 30)

The list of columns in the table.

Type: Array of [TableColumn](#) (p. 67) objects

workbookCursor (p. 30)

Indicates the cursor of the workbook at which the data returned by this request is read. Workbook cursor keeps increasing with every update and the increments are not sequential.

Type: Long

Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 77).

AccessDeniedException

You do not have sufficient access to perform this action. Check that the workbook is owned by you and your IAM policy allows access to the resource in the request.

HTTP Status Code: 403

InternalServerErrorException

There were unexpected errors from the server.

HTTP Status Code: 500

RequestTimeoutException

The request timed out.

HTTP Status Code: 504

ResourceNotFoundException

A Workbook, Table, App, Screen or Screen Automation was not found with the given ID.

HTTP Status Code: 404

ServiceUnavailableException

Remote service is unreachable.

HTTP Status Code: 503

ThrottlingException

Tps(transactions per second) rate reached.

HTTP Status Code: 429

ValidationException

Request is invalid. The message in the response contains details on why the request is invalid.

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 V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

ListTableRows

The ListTableRows API allows you to retrieve a list of all the rows in a table in a workbook.

Request Syntax

```
POST /workbooks/workbookId/tables/tableId/rows/list HTTP/1.1
Content-type: application/json

{
  "maxResults": number,
  "nextToken": "string",
  "rowIds": [ "string" ]
}
```

URI Request Parameters

The request uses the following URI parameters.

tableId (p. 33)

The ID of the table whose rows are being retrieved.

If a table with the specified id could not be found, this API throws ResourceNotFoundException.

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: Yes

workbookId (p. 33)

The ID of the workbook that contains the table whose rows are being retrieved.

If a workbook with the specified id could not be found, this API throws ResourceNotFoundException.

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: Yes

Request Body

The request accepts the following data in JSON format.

maxResults (p. 33)

The maximum number of rows to return in each page of the results.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 100.

Required: No

nextToken (p. 33)

This parameter is optional. If a nextToken is not specified, the API returns the first page of data.

Pagination tokens expire after 1 hour. If you use a token that was returned more than an hour back, the API will throw ValidationException.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: ^(?:!\s*\$).+

Required: No

rowIds (p. 33)

This parameter is optional. If one or more row ids are specified in this list, then only the specified row ids are returned in the result. If no row ids are specified here, then all the rows in the table are returned.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Fixed length of 77.

Pattern: row:[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}\/[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: No

Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "columnIds": [ "string" ],
  "nextToken": "string",
  "rowIdsNotFound": [ "string" ],
  "rows": [
    {
      "cells": [
        {
          "format": "string",
          "formattedValue": "string",
          "formula": "string",
          "rawValue": "string"
        }
      ],
      "rowId": "string"
    }
  ],
  "workbookCursor": 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.

[columnIds \(p. 34\)](#)

The list of columns in the table whose row data is returned in the result.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Fixed length of 36.

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

[nextToken \(p. 34\)](#)

Provides the pagination token to load the next page if there are more results matching the request. If a pagination token is not present in the response, it means that all data matching the request has been loaded.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `^(?!\\s*$).+`

[rowIdsNotFound \(p. 34\)](#)

The list of row ids included in the request that were not found in the table.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Fixed length of 77.

Pattern: `row:[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}\\/[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

[rows \(p. 34\)](#)

The list of rows in the table. Note that this result is paginated, so this list contains a maximum of 100 rows.

Type: Array of [TableRow \(p. 69\)](#) objects

[workbookCursor \(p. 34\)](#)

Indicates the cursor of the workbook at which the data returned by this request is read. Workbook cursor keeps increasing with every update and the increments are not sequential.

Type: Long

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 77\)](#).

AccessDeniedException

You do not have sufficient access to perform this action. Check that the workbook is owned by you and your IAM policy allows access to the resource in the request.

HTTP Status Code: 403

InternalServerErrorException

There were unexpected errors from the server.

HTTP Status Code: 500

RequestTimeoutException

The request timed out.

HTTP Status Code: 504

ResourceNotFoundException

A Workbook, Table, App, Screen or Screen Automation was not found with the given ID.

HTTP Status Code: 404

ServiceUnavailableException

Remote service is unreachable.

HTTP Status Code: 503

ThrottlingException

Tps(transactions per second) rate reached.

HTTP Status Code: 429

ValidationException

Request is invalid. The message in the response contains details on why the request is invalid.

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 V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

ListTables

The ListTables API allows you to retrieve a list of all the tables in a workbook.

Request Syntax

```
GET /workbooks/workbookId/tables?maxResults=maxResults&nextToken=nextToken HTTP/1.1
```

URI Request Parameters

The request uses the following URI parameters.

maxResults (p. 37)

The maximum number of tables to return in each page of the results.

Valid Range: Minimum value of 1. Maximum value of 100.

nextToken (p. 37)

This parameter is optional. If a nextToken is not specified, the API returns the first page of data.

Pagination tokens expire after 1 hour. If you use a token that was returned more than an hour back, the API will throw ValidationException.

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `^(?!\\s*$).+`

workbookId (p. 37)

The ID of the workbook whose tables are being retrieved.

If a workbook with the specified id could not be found, this API throws ResourceNotFoundException.

Length Constraints: Fixed length of 36.

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "tables": [
    {
      "tableId": "string",
      "tableName": "string"
    }
  ]
}
```

```
    }  
  ],  
  "workbookCursor": 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.

nextToken (p. 37)

Provides the pagination token to load the next page if there are more results matching the request. If a pagination token is not present in the response, it means that all data matching the request has been loaded.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `^(?!\\s*$).+`

tables (p. 37)

The list of tables in the workbook.

Type: Array of [Table](#) (p. 66) objects

workbookCursor (p. 37)

Indicates the cursor of the workbook at which the data returned by this request is read. Workbook cursor keeps increasing with every update and the increments are not sequential.

Type: Long

Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 77).

AccessDeniedException

You do not have sufficient access to perform this action. Check that the workbook is owned by you and your IAM policy allows access to the resource in the request.

HTTP Status Code: 403

InternalServerErrorException

There were unexpected errors from the server.

HTTP Status Code: 500

RequestTimeoutException

The request timed out.

HTTP Status Code: 504

ResourceNotFoundException

A Workbook, Table, App, Screen or Screen Automation was not found with the given ID.

HTTP Status Code: 404

ServiceUnavailableException

Remote service is unreachable.

HTTP Status Code: 503

ThrottlingException

Tps(transactions per second) rate reached.

HTTP Status Code: 429

ValidationException

Request is invalid. The message in the response contains details on why the request is invalid.

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 V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

QueryTableRows

The QueryTableRows API allows you to use a filter formula to query for specific rows in a table.

Request Syntax

```
POST /workbooks/workbookId/tables/tableId/rows/query HTTP/1.1
Content-type: application/json

{
  "filterFormula": {
    "contextRowId": "string",
    "formula": "string"
  },
  "maxResults": number,
  "nextToken": "string"
}
```

URI Request Parameters

The request uses the following URI parameters.

tableId (p. 40)

The ID of the table whose rows are being queried.

If a table with the specified id could not be found, this API throws `ResourceNotFoundException`.

Length Constraints: Fixed length of 36.

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: Yes

workbookId (p. 40)

The ID of the workbook whose table rows are being queried.

If a workbook with the specified id could not be found, this API throws `ResourceNotFoundException`.

Length Constraints: Fixed length of 36.

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: Yes

Request Body

The request accepts the following data in JSON format.

filterFormula (p. 40)

An object that represents a filter formula along with the id of the context row under which the filter function needs to evaluate.

Type: [Filter](#) (p. 58) object

Required: Yes

maxResults (p. 40)

The maximum number of rows to return in each page of the results.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 100.

Required: No

nextToken (p. 40)

This parameter is optional. If a nextToken is not specified, the API returns the first page of data.

Pagination tokens expire after 1 hour. If you use a token that was returned more than an hour back, the API will throw ValidationException.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `^(?!\\s*$).+`

Required: No

Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "columnIds": [ "string" ],
  "nextToken": "string",
  "rows": [
    {
      "cells": [
        {
          "format": "string",
          "formattedValue": "string",
          "formula": "string",
          "rawValue": "string"
        }
      ],
      "rowId": "string"
    }
  ],
  "workbookCursor": 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.

columnIds (p. 41)

The list of columns in the table whose row data is returned in the result.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

[nextToken \(p. 41\)](#)

Provides the pagination token to load the next page if there are more results matching the request. If a pagination token is not present in the response, it means that all data matching the request has been loaded.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: ^(?!\s*\$).+

[rows \(p. 41\)](#)

The list of rows in the table that match the query filter.

Type: Array of [TableRow \(p. 69\)](#) objects

[workbookCursor \(p. 41\)](#)

Indicates the cursor of the workbook at which the data returned by this request is read. Workbook cursor keeps increasing with every update and the increments are not sequential.

Type: Long

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 77\)](#).

AccessDeniedException

You do not have sufficient access to perform this action. Check that the workbook is owned by you and your IAM policy allows access to the resource in the request.

HTTP Status Code: 403

InternalServerErrorException

There were unexpected errors from the server.

HTTP Status Code: 500

RequestTimeoutException

The request timed out.

HTTP Status Code: 504

ResourceNotFoundException

A Workbook, Table, App, Screen or Screen Automation was not found with the given ID.

HTTP Status Code: 404

ServiceUnavailableException

Remote service is unreachable.

HTTP Status Code: 503

ThrottlingException

Tps(transactions per second) rate reached.

HTTP Status Code: 429

ValidationException

Request is invalid. The message in the response contains details on why the request is invalid.

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 V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

StartTableDataImportJob

The StartTableDataImportJob API allows you to start an import job on a table. This API will only return the id of the job that was started. To find out the status of the import request, you need to call the DescribeTableDataImportJob API.

Request Syntax

```
POST /workbooks/workbookId/tables/tableId/import HTTP/1.1  
Content-type: application/json
```

```
{  
  "clientRequestToken": "string",  
  "dataFormat": "string",  
  "dataSource": {  
    "dataSourceConfig": {  
      "dataSourceUrl": "string"  
    }  
  },  
  "importOptions": {  
    "delimitedTextOptions": {  
      "dataCharacterEncoding": "string",  
      "delimiter": "string",  
      "hasHeaderRow": boolean,  
      "ignoreEmptyRows": boolean  
    },  
    "destinationOptions": {  
      "columnMap": {  
        "string": {  
          "columnIndex": number  
        }  
      }  
    }  
  }  
}
```

URI Request Parameters

The request uses the following URI parameters.

tableId (p. 44)

The ID of the table where the rows are being imported.

If a table with the specified id could not be found, this API throws ResourceNotFoundException.

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: Yes

workbookId (p. 44)

The ID of the workbook where the rows are being imported.

If a workbook with the specified id could not be found, this API throws ResourceNotFoundException.

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: Yes

Request Body

The request accepts the following data in JSON format.

clientRequestToken (p. 44)

The request token for performing the update action. Request tokens help to identify duplicate requests. If a call times out or fails due to a transient error like a failed network connection, you can retry the call with the same request token. The service ensures that if the first call using that request token is successfully performed, the second call will not perform the action again.

Note that request tokens are valid only for a few minutes. You cannot use request tokens to dedupe requests spanning hours or days.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Pattern: ^(?:!\s*\$).+

Required: Yes

dataFormat (p. 44)

The format of the data that is being imported. Currently the only option supported is "DELIMITED_TEXT".

Type: String

Valid Values: DELIMITED_TEXT

Required: Yes

dataSource (p. 44)

The source of the data that is being imported. The size of source must be no larger than 100 MB. Source must have no more than 100,000 cells and no more than 1,000 rows.

Type: [ImportDataSource](#) (p. 59) object

Required: Yes

importOptions (p. 44)

The options for customizing this import request.

Type: [ImportOptions](#) (p. 62) object

Required: Yes

Response Syntax

```
HTTP/1.1 200
Content-type: application/json
```

```
{  
  "jobId": "string",  
  "jobStatus": "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.

jobId (p. 45)

The id that is assigned to this import job. Future requests to find out the status of this import job need to send this id in the appropriate parameter in the request.

Type: String

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

Pattern: `^[^\n\r\x00\x08\x0B\x0C\x0E\x1F]*$`

jobStatus (p. 45)

The status of the import job immediately after submitting the request.

Type: String

Valid Values: `SUBMITTED | IN_PROGRESS | COMPLETED | FAILED`

Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 77).

AccessDeniedException

You do not have sufficient access to perform this action. Check that the workbook is owned by you and your IAM policy allows access to the resource in the request.

HTTP Status Code: 403

InternalServerError

There were unexpected errors from the server.

HTTP Status Code: 500

ResourceNotFoundException

A Workbook, Table, App, Screen or Screen Automation was not found with the given ID.

HTTP Status Code: 404

ServiceUnavailableException

Remote service is unreachable.

HTTP Status Code: 503

ThrottlingException

Tps(transactions per second) rate reached.

HTTP Status Code: 429

ValidationException

Request is invalid. The message in the response contains details on why the request is invalid.

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 V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

Data Types

The Amazon Honeycode 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:

- [Cell](#) (p. 49)
- [CellInput](#) (p. 51)
- [ColumnMetadata](#) (p. 52)
- [CreateRowData](#) (p. 53)
- [DataItem](#) (p. 54)
- [DelimitedTextImportOptions](#) (p. 55)
- [DestinationOptions](#) (p. 56)
- [FailedBatchItem](#) (p. 57)
- [Filter](#) (p. 58)
- [ImportDataSource](#) (p. 59)
- [ImportDataSourceConfig](#) (p. 60)
- [ImportJobSubmitter](#) (p. 61)
- [ImportOptions](#) (p. 62)
- [ResultRow](#) (p. 63)
- [ResultSet](#) (p. 64)
- [SourceDataColumnProperties](#) (p. 65)
- [Table](#) (p. 66)
- [TableColumn](#) (p. 67)
- [TableDataImportJobMetadata](#) (p. 68)
- [TableRow](#) (p. 69)
- [UpdateRowData](#) (p. 70)
- [UpsertRowData](#) (p. 71)
- [UpsertRowsResult](#) (p. 73)
- [VariableValue](#) (p. 74)

Cell

An object that represents a single cell in a table.

Contents

format

The format of the cell. If this field is empty, then the format is either not specified in the workbook or the format is set to AUTO.

Type: String

Valid Values: AUTO | NUMBER | CURRENCY | DATE | TIME | DATE_TIME | PERCENTAGE | TEXT | ACCOUNTING | CONTACT | ROWLINK

Required: No

formattedValue

The formatted value of the cell. This is the value that you see displayed in the cell in the UI.

Note that the formatted value of a cell is always represented as a string irrespective of the data that is stored in the cell. For example, if a cell contains a date, the formatted value of the cell is the string representation of the formatted date being shown in the cell in the UI. See details in the `rawValue` field below for how cells of different formats will have different raw and formatted values.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 8192.

Pattern: [`\s\S`]*

Required: No

formula

The formula contained in the cell. This field is empty if a cell does not have a formula.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 8192.

Pattern: `^=.*`

Required: No

rawValue

The raw value of the data contained in the cell. The raw value depends on the format of the data in the cell. However the attribute in the API return value is always a string containing the raw value.

Cells with format DATE, DATE_TIME or TIME have the raw value as a floating point number where the whole number represents the number of days since 1/1/1900 and the fractional part represents the fraction of the day since midnight. For example, a cell with date 11/3/2020 has the raw value "44138". A cell with the time 9:00 AM has the raw value "0.375" and a cell with date/time value of 11/3/2020 9:00 AM has the raw value "44138.375". Notice that even though the raw value is a number in all three cases, it is still represented as a string.

Cells with format NUMBER, CURRENCY, PERCENTAGE and ACCOUNTING have the raw value of the data as the number representing the data being displayed. For example, the number 1.325 with

two decimal places in the format will have its raw value as "1.325" and formatted value as "1.33". A currency value for \$10 will have the raw value as "10" and formatted value as "\$10.00". A value representing 20% with two decimal places in the format will have its raw value as "0.2" and the formatted value as "20.00%". An accounting value of -\$25 will have "-25" as the raw value and "\$ (25.00)" as the formatted value.

Cells with format TEXT will have the raw text as the raw value. For example, a cell with text "John Smith" will have "John Smith" as both the raw value and the formatted value.

Cells with format CONTACT will have the name of the contact as a formatted value and the email address of the contact as the raw value. For example, a contact for John Smith will have "John Smith" as the formatted value and "john.smith@example.com" as the raw value.

Cells with format ROWLINK (aka picklist) will have the first column of the linked row as the formatted value and the row id of the linked row as the raw value. For example, a cell containing a picklist to a table that displays task status might have "Completed" as the formatted value and "row:dfcefaee-5b37-4355-8f28-40c3e4ff5dd4/ca432b2f-b8eb-431d-9fb5-cbe0342f9f03" as the raw value.

Cells with format AUTO or cells without any format that are auto-detected as one of the formats above will contain the raw and formatted values as mentioned above, based on the auto-detected formats. If there is no auto-detected format, the raw and formatted values will be the same as the data in the cell.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 32767.

Pattern: [`\s\S`]*

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 V2](#)
- [AWS SDK for Ruby V3](#)

CellInput

CellInput object contains the data needed to create or update cells in a table.

Contents

fact

Fact represents the data that is entered into a cell. This data can be free text or a formula. Formulas need to start with the equals (=) sign.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 8192.

Pattern: [\s\S]*

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 V2](#)
- [AWS SDK for Ruby V3](#)

ColumnMetadata

Metadata for column in the table.

Contents

format

The format of the column.

Type: String

Valid Values: AUTO | NUMBER | CURRENCY | DATE | TIME | DATE_TIME | PERCENTAGE | TEXT | ACCOUNTING | CONTACT | ROWLINK

Required: Yes

name

The name of the column.

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 V2](#)
- [AWS SDK for Ruby V3](#)

CreateRowData

Data needed to create a single row in a table as part of the BatchCreateTableRows request.

Contents

batchItemId

An external identifier that represents the single row that is being created as part of the BatchCreateTableRows request. This can be any string that you can use to identify the row in the request. The BatchCreateTableRows API puts the batch item id in the results to allow you to link data in the request to data in the results.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^(?!\\s*$).+`

Required: Yes

cellsToCreate

A map representing the cells to create in the new row. The key is the column id of the cell and the value is the CellInput object that represents the data to set in that cell.

Type: String to [CellInput \(p. 51\)](#) object map

Map Entries: Maximum number of 100 items.

Key Length Constraints: Fixed length of 36.

Key Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

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 V2](#)
- [AWS SDK for Ruby V3](#)

Dataltem

The data in a particular data cell defined on the screen.

Contents

formattedValue

The formatted value of the data. e.g. John Smith.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 8192.

Pattern: [\s\S]*

Required: No

overrideFormat

The overrideFormat is optional and is specified only if a particular row of data has a different format for the data than the default format defined on the screen or the table.

Type: String

Valid Values: AUTO | NUMBER | CURRENCY | DATE | TIME | DATE_TIME | PERCENTAGE | TEXT | ACCOUNTING | CONTACT | ROWLINK

Required: No

rawValue

The raw value of the data. e.g. jsmith@example.com

Type: String

Length Constraints: Minimum length of 0. Maximum length of 32767.

Pattern: [\s\S]*

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 V2](#)
- [AWS SDK for Ruby V3](#)

DelimitedTextImportOptions

An object that contains the options relating to parsing delimited text as part of an import request.

Contents

dataCharacterEncoding

The encoding of the data in the input file.

Type: String

Valid Values: UTF-8 | US-ASCII | ISO-8859-1 | UTF-16BE | UTF-16LE | UTF-16

Required: No

delimiter

The delimiter to use for separating columns in a single row of the input.

Type: String

Length Constraints: Fixed length of 1.

Pattern: `^[^\n\r\x00\x08\x0B\x0C\x0E\x1F]?$`

Required: Yes

hasHeaderRow

Indicates whether the input file has a header row at the top containing the column names.

Type: Boolean

Required: No

ignoreEmptyRows

A parameter to indicate whether empty rows should be ignored or be included in the import.

Type: Boolean

Required: No

See Also

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

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DestinationOptions

An object that contains the options relating to the destination of the import request.

Contents

columnMap

A map of the column id to the import properties for each column.

Type: String to [SourceDataColumnProperties \(p. 65\)](#) object map

Map Entries: Maximum number of 100 items.

Key Length Constraints: Fixed length of 36.

Key Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

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 V2](#)
- [AWS SDK for Ruby V3](#)

FailedBatchItem

A single item in a batch that failed to perform the intended action because of an error preventing it from succeeding.

Contents

errorMessage

The error message that indicates why the batch item failed.

Type: String

Pattern: `^(?!\\s*$).+`

Required: Yes

id

The id of the batch item that failed. This is the batch item id for the `BatchCreateTableRows` and `BatchUpsertTableRows` operations and the row id for the `BatchUpdateTableRows` and `BatchDeleteTableRows` operations.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^(?!\\s*$).+`

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 V2](#)
- [AWS SDK for Ruby V3](#)

Filter

An object that represents a filter formula along with the id of the context row under which the filter function needs to evaluate.

Contents

contextRowId

The optional contextRowId attribute can be used to specify the row id of the context row if the filter formula contains unqualified references to table columns and needs a context row to evaluate them successfully.

Type: String

Length Constraints: Fixed length of 77.

Pattern: row:[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}\/[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: No

formula

A formula representing a filter function that returns zero or more matching rows from a table. Valid formulas in this field return a list of rows from a table. The most common ways of writing a formula to return a list of rows are to use the FindRow() or Filter() functions. Any other formula that returns zero or more rows is also acceptable. For example, you can use a formula that points to a cell that contains a filter function.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 8192.

Pattern: ^=.*

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 V2](#)
- [AWS SDK for Ruby V3](#)

ImportDataSource

An object that has details about the source of the data that was submitted for import.

Contents

dataSourceConfig

The configuration parameters for the data source of the import

Type: [ImportDataSourceConfig \(p. 60\)](#) object

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 V2](#)
- [AWS SDK for Ruby V3](#)

ImportDataSourceConfig

An object that contains the configuration parameters for the data source of an import request.

Contents

dataSourceUrl

The URL from which source data will be downloaded for the import request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 8000.

Pattern: `^https://[/\[\^\n\r\x00\x08\x0B\x0C\x0E\x1F]*$`

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 V2](#)
- [AWS SDK for Ruby V3](#)

ImportJobSubmitter

An object that contains the attributes of the submitter of the import job.

Contents

email

The email id of the submitter of the import job, if available.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 254.

Pattern: `^[a-zA-Z0-9_-\.\]+@([a-zA-Z0-9_-\.\]+)\.([a-zA-Z]{2,5})$`

Required: No

userArn

The AWS user ARN of the submitter of the import job, if available.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

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 V2](#)
- [AWS SDK for Ruby V3](#)

ImportOptions

An object that contains the options specified by the submitter of the import request.

Contents

delimitedTextOptions

Options relating to parsing delimited text. Required if dataFormat is DELIMITED_TEXT.

Type: [DelimitedTextImportOptions \(p. 55\)](#) object

Required: No

destinationOptions

Options relating to the destination of the import request.

Type: [DestinationOptions \(p. 56\)](#) 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 V2](#)
- [AWS SDK for Ruby V3](#)

ResultRow

A single row in the ResultSet.

Contents

dataItems

List of all the data cells in a row.

Type: Array of [DataItem \(p. 54\)](#) objects

Required: Yes

rowId

The ID for a particular row.

Type: String

Length Constraints: Fixed length of 77.

Pattern: row: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}\/[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

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 V2](#)
- [AWS SDK for Ruby V3](#)

ResultSet

ResultSet contains the results of the request for a single block or list defined on the screen.

Contents

headers

List of headers for all the data cells in the block. The header identifies the name and default format of the data cell. Data cells appear in the same order in all rows as defined in the header. The names and formats are not repeated in the rows. If a particular row does not have a value for a data cell, a blank value is used.

For example, a task list that displays the task name, due date and assigned person might have headers [{ "name": "Task Name"}, {"name": "Due Date", "format": "DATE"}, {"name": "Assigned", "format": "CONTACT"}]. Every row in the result will have the task name as the first item, due date as the second item and assigned person as the third item. If a particular task does not have a due date, that row will still have a blank value in the second element and the assigned person will still be in the third element.

Type: Array of [ColumnMetadata \(p. 52\)](#) objects

Required: Yes

rows

List of rows returned by the request. Each row has a row Id and a list of data cells in that row. The data cells will be present in the same order as they are defined in the header.

Type: Array of [ResultRow \(p. 63\)](#) objects

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 V2](#)
- [AWS SDK for Ruby V3](#)

SourceDataColumnProperties

An object that contains the properties for importing data to a specific column in a table.

Contents

columnIndex

The index of the column in the input file.

Type: Integer

Valid Range: Minimum value of 1.

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 V2](#)
- [AWS SDK for Ruby V3](#)

Table

An object representing the properties of a table in a workbook.

Contents

tableId

The id of the table.

Type: String

Length Constraints: Fixed length of 36.

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: No

tableName

The name of the table.

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 V2](#)
- [AWS SDK for Ruby V3](#)

TableColumn

An object that contains attributes about a single column in a table

Contents

format

The column level format that is applied in the table. An empty value in this field means that the column format is the default value 'AUTO'.

Type: String

Valid Values: AUTO | NUMBER | CURRENCY | DATE | TIME | DATE_TIME | PERCENTAGE | TEXT | ACCOUNTING | CONTACT | ROWLINK

Required: No

tableColumnId

The id of the column in the table.

Type: String

Length Constraints: Fixed length of 36.

Pattern: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: No

tableColumnName

The name of the column in the table.

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 V2](#)
- [AWS SDK for Ruby V3](#)

TableDataImportJobMetadata

The metadata associated with the table data import job that was submitted.

Contents

dataSource

The source of the data that was submitted for import.

Type: [ImportDataSource \(p. 59\)](#) object

Required: Yes

importOptions

The options that was specified at the time of submitting the import request.

Type: [ImportOptions \(p. 62\)](#) object

Required: Yes

submitter

Details about the submitter of the import request.

Type: [ImportJobSubmitter \(p. 61\)](#) object

Required: Yes

submitTime

The timestamp when the job was submitted for import.

Type: Timestamp

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 V2](#)
- [AWS SDK for Ruby V3](#)

TableRow

An object that contains attributes about a single row in a table

Contents

cells

A list of cells in the table row. The cells appear in the same order as the columns of the table.

Type: Array of [Cell \(p. 49\)](#) objects

Required: Yes

rowId

The id of the row in the table.

Type: String

Length Constraints: Fixed length of 77.

Pattern: row: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}\/[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

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 V2](#)
- [AWS SDK for Ruby V3](#)

UpdateRowData

Data needed to create a single row in a table as part of the BatchCreateTableRows request.

Contents

cellsToUpdate

A map representing the cells to update in the given row. The key is the column id of the cell and the value is the CellInput object that represents the data to set in that cell.

Type: String to [CellInput \(p. 51\)](#) object map

Map Entries: Maximum number of 100 items.

Key Length Constraints: Fixed length of 36.

Key Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: Yes

rowId

The id of the row that needs to be updated.

Type: String

Length Constraints: Fixed length of 77.

Pattern: `row:[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}\/[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

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 V2](#)
- [AWS SDK for Ruby V3](#)

UpsertRowData

Data needed to upsert rows in a table as part of a single item in the BatchUpsertTableRows request.

Contents

batchItemId

An external identifier that represents a single item in the request that is being upserted as part of the BatchUpsertTableRows request. This can be any string that you can use to identify the item in the request. The BatchUpsertTableRows API puts the batch item id in the results to allow you to link data in the request to data in the results.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^(?!\\s*$).+`

Required: Yes

cellsToUpdate

A map representing the cells to update for the matching rows or an appended row. The key is the column id of the cell and the value is the CellInput object that represents the data to set in that cell.

Type: String to [CellInput \(p. 51\)](#) object map

Map Entries: Maximum number of 100 items.

Key Length Constraints: Fixed length of 36.

Key Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: Yes

filter

The filter formula to use to find existing matching rows to update. The formula needs to return zero or more rows. If the formula returns 0 rows, then a new row will be appended in the target table. If the formula returns one or more rows, then the returned rows will be updated.

Note that the filter formula needs to return rows from the target table for the upsert operation to succeed. If the filter formula has a syntax error or it doesn't evaluate to zero or more rows in the target table for any one item in the input list, then the entire BatchUpsertTableRows request fails and no updates are made to the table.

Type: [Filter \(p. 58\)](#) object

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 V2](#)
- [AWS SDK for Ruby V3](#)

UpsertRowsResult

An object that represents the result of a single upsert row request.

Contents

rowIds

The list of row ids that were changed as part of an upsert row operation. If the upsert resulted in an update, this list could potentially contain multiple rows that matched the filter and hence got updated. If the upsert resulted in an append, this list would only have the single row that was appended.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Fixed length of 77.

Pattern: row: [0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}\/[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}

Required: Yes

upsertAction

The result of the upsert action.

Type: String

Valid Values: UPDATED | APPENDED

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 V2](#)
- [AWS SDK for Ruby V3](#)

VariableValue

The input variables to the app to be used by the InvokeScreenAutomation action request.

Contents

rawValue

Raw value of the variable.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 32767.

Pattern: [\s\S]*

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 V2](#)
- [AWS SDK for Ruby V3](#)

Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see [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'THHMMSS'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

NotAuthorized

You do not have permission to perform this action.

HTTP Status Code: 400

OptInRequired

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

RequestExpired

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

ServiceUnavailable

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 400

ValidationError

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400