Amazon RDS Data Service

API Reference

API Version 2018-08-01

Amazon RDS Data Service: API Reference

Table of Contents

Welcome

Amazon RDS provides an HTTP endpoint to run SQL statements on an Amazon Aurora Serverless DB cluster. To run these statements, you work with the Data Service API.

For more information about the Data Service API, see Using the Data API for Aurora Serverless in the *Amazon Aurora User Guide*.

This document was last published on October 6, 2021.

Actions

The following actions are supported:

BatchExecuteStatement

Runs a batch SQL statement over an array of data.

You can run bulk update and insert operations for multiple records using a DML statement with different parameter sets. Bulk operations can provide a significant performance improvement over individual insert and update operations.

> **Important**
> If a call isn't part of a transaction because it doesn't include the `transactionID` parameter, changes that result from the call are committed automatically.

Request Syntax

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

{
   "database": "string",
   "parameterSets": [
      [
         {
            "name": "string",
            "typeHint": "string",
            "value": {
               "arrayValue": {
                  "arrayValues": [
                     "ArrayValue"
                  ],
                  "booleanValues": [ boolean ],
                  "doubleValues": [ number ],
                  "longValues": [ number ],
                  "stringValues": [ "string" ]
               },
               "blobValue": blob,
               "booleanValue": boolean,
               "doubleValue": number,
               "isNull": boolean,
               "longValue": number,
               "stringValue": "string"
            }
         }
      ]
   ],
   "resourceArn": "string",
   "schema": "string",
   "secretArn": "string",
   "sql": "string",
   "transactionId": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

database (p. 3)

The name of the database.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 64.

Required: No

parameterSets (p. 3)

The parameter set for the batch operation.

The SQL statement is executed as many times as the number of parameter sets provided. To execute a SQL statement with no parameters, use one of the following options:

- Specify one or more empty parameter sets.
- Use the `ExecuteStatement` operation instead of the `BatchExecuteStatement` operation.

  Note
  Array parameters are not supported.

Type: Array of arrays of SqlParameter (p. 35) objects

Required: No

resourceArn (p. 3)

The Amazon Resource Name (ARN) of the Aurora Serverless DB cluster.

Type: String

Length Constraints: Minimum length of 11. Maximum length of 100.

Required: Yes

schema (p. 3)

The name of the database schema.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 64.

Required: No

secretArn (p. 3)

The name or ARN of the secret that enables access to the DB cluster.

Type: String

Length Constraints: Minimum length of 11. Maximum length of 100.

Required: Yes

sql (p. 3)

The SQL statement to run.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 65536.

Required: Yes

transactionId (p. 3)

The identifier of a transaction that was started by using the `BeginTransaction` operation. Specify the transaction ID of the transaction that you want to include the SQL statement in.

If the SQL statement is not part of a transaction, don't set this parameter.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 192.

Required: No

Response Syntax

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

{
   "updateResults": [
      {
         "generatedFields": [
            {
               "arrayValue": {
                  "arrayValues": [
                     "ArrayValue"
                  ],
                  "booleanValues": [ boolean ],
                  "doubleValues": [ number ],
                  "longValues": [ number ],
                  "stringValues": [ "string" ]
               },
               "blobValue": blob,
               "booleanValue": boolean,
               "doubleValue": number,
               "isNull": boolean,
               "longValue": number,
               "stringValue": "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.

updateResults (p. 5)

The execution results of each batch entry.

Type: Array of UpdateResult (p. 38) objects

Errors

BadRequestException

There is an error in the call or in a SQL statement.

HTTP Status Code: 400

ForbiddenException

There are insufficient privileges to make the call.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred.

HTTP Status Code: 500

ServiceUnavailableError

The service specified by the `resourceArn` parameter is not available.

HTTP Status Code: 503

StatementTimeoutException

The execution of the SQL statement timed out.

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

BeginTransaction

Starts a SQL transaction.

Important
A transaction can run for a maximum of 24 hours. A transaction is terminated and rolled back automatically after 24 hours.
A transaction times out if no calls use its transaction ID in three minutes. If a transaction times out before it's committed, it's rolled back automatically.
DDL statements inside a transaction cause an implicit commit. We recommend that you run each DDL statement in a separate `ExecuteStatement` call with `continueAfterTimeout` enabled.

Request Syntax

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

{
   "database": "string",
   "resourceArn": "string",
   "schema": "string",
   "secretArn": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

database (p. 7)

The name of the database.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 64.

Required: No

resourceArn (p. 7)

The Amazon Resource Name (ARN) of the Aurora Serverless DB cluster.

Type: String

Length Constraints: Minimum length of 11. Maximum length of 100.

Required: Yes

schema (p. 7)

The name of the database schema.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 64.

Required: No

secretArn (p. 7)

The name or ARN of the secret that enables access to the DB cluster.

Type: String

Length Constraints: Minimum length of 11. Maximum length of 100.

Required: Yes

Response Syntax

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

{
   "transactionId": "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.

transactionId (p. 8)

The transaction ID of the transaction started by the call.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 192.

Errors

BadRequestException

There is an error in the call or in a SQL statement.

HTTP Status Code: 400

ForbiddenException

There are insufficient privileges to make the call.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred.

HTTP Status Code: 500

ServiceUnavailableError

The service specified by the `resourceArn` parameter is not available.

HTTP Status Code: 503

StatementTimeoutException

The execution of the SQL statement timed out.

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

CommitTransaction

Ends a SQL transaction started with the `BeginTransaction` operation and commits the changes.

Request Syntax

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

{
   "resourceArn": "string",
   "secretArn": "string",
   "transactionId": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

resourceArn (p. 10)

The Amazon Resource Name (ARN) of the Aurora Serverless DB cluster.

Type: String

Length Constraints: Minimum length of 11. Maximum length of 100.

Required: Yes

secretArn (p. 10)

The name or ARN of the secret that enables access to the DB cluster.

Type: String

Length Constraints: Minimum length of 11. Maximum length of 100.

Required: Yes

transactionId (p. 10)

The identifier of the transaction to end and commit.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 192.

Required: Yes

Response Syntax

```
HTTP/1.1 200
```

```
Content-type: application/json

{
   "transactionStatus": "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.

transactionStatus (p. 10)

The status of the commit operation.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 128.

Errors

BadRequestException

There is an error in the call or in a SQL statement.

HTTP Status Code: 400

ForbiddenException

There are insufficient privileges to make the call.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred.

HTTP Status Code: 500

NotFoundException

The `resourceArn`, `secretArn`, or `transactionId` value can't be found.

HTTP Status Code: 404

ServiceUnavailableError

The service specified by the `resourceArn` parameter is not available.

HTTP Status Code: 503

StatementTimeoutException

The execution of the SQL statement timed out.

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

ExecuteSql

Runs one or more SQL statements.

Important
This operation is deprecated. Use the `BatchExecuteStatement` or `ExecuteStatement` operation.

Request Syntax

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

{
   "awsSecretStoreArn": "string",
   "database": "string",
   "dbClusterOrInstanceArn": "string",
   "schema": "string",
   "sqlStatements": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

awsSecretStoreArn (p. 13)

The Amazon Resource Name (ARN) of the secret that enables access to the DB cluster.

Type: String

Length Constraints: Minimum length of 11. Maximum length of 100.

Required: Yes

database (p. 13)

The name of the database.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 64.

Required: No

dbClusterOrInstanceArn (p. 13)

The ARN of the Aurora Serverless DB cluster.

Type: String

Length Constraints: Minimum length of 11. Maximum length of 100.

Required: Yes

schema (p. 13)

The name of the database schema.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 64.

Required: No

sqlStatements (p. 13)

One or more SQL statements to run on the DB cluster.

You can separate SQL statements from each other with a semicolon (;). Any valid SQL statement is permitted, including data definition, data manipulation, and commit statements.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 65536.

Required: Yes

Response Syntax

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

{
   "sqlStatementResults": [
      {
         "numberOfRecordsUpdated": number,
         "resultFrame": {
            "records": [
               {
                  "values": [
                     {
                        "arrayValues": [
                           "Value"
                        ],
                        "bigIntValue": number,
                        "bitValue": boolean,
                        "blobValue": blob,
                        "doubleValue": number,
                        "intValue": number,
                        "isNull": boolean,
                        "realValue": number,
                        "stringValue": "string",
                        "structValue": {
                           "attributes": [
                              "Value"
                           ]
                        }
                     }
                  ]
               }
            ],
            "resultSetMetadata": {
               "columnCount": number,
               "columnMetadata": [
                  {
                     "arrayBaseColumnType": number,
                     "isAutoIncrement": boolean,
```

```
                    "isCaseSensitive": boolean,
                    "isCurrency": boolean,
                    "isSigned": boolean,
                    "label": "string",
                    "name": "string",
                    "nullable": number,
                    "precision": number,
                    "scale": number,
                    "schemaName": "string",
                    "tableName": "string",
                    "type": number,
                    "typeName": "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.

sqlStatementResults (p. 14)

The results of the SQL statement or statements.

Type: Array of SqlStatementResult (p. 36) objects

Errors

BadRequestException

There is an error in the call or in a SQL statement.

HTTP Status Code: 400

ForbiddenException

There are insufficient privileges to make the call.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred.

HTTP Status Code: 500

ServiceUnavailableError

The service specified by the `resourceArn` parameter is not available.

HTTP Status Code: 503

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

ExecuteStatement

Runs a SQL statement against a database.

> **Important**
> If a call isn't part of a transaction because it doesn't include the `transactionID` parameter, changes that result from the call are committed automatically.

The response size limit is 1 MB. If the call returns more than 1 MB of response data, the call is terminated.

Request Syntax

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

{
   "continueAfterTimeout": boolean,
   "database": "string",
   "includeResultMetadata": boolean,
   "parameters": [
      {
         "name": "string",
         "typeHint": "string",
         "value": {
            "arrayValue": {
               "arrayValues": [
                  "ArrayValue"
               ],
               "booleanValues": [ boolean ],
               "doubleValues": [ number ],
               "longValues": [ number ],
               "stringValues": [ "string" ]
            },
            "blobValue": blob,
            "booleanValue": boolean,
            "doubleValue": number,
            "isNull": boolean,
            "longValue": number,
            "stringValue": "string"
         }
      }
   ],
   "resourceArn": "string",
   "resultSetOptions": {
      "decimalReturnType": "string"
   },
   "schema": "string",
   "secretArn": "string",
   "sql": "string",
   "transactionId": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

continueAfterTimeout (p. 17)

A value that indicates whether to continue running the statement after the call times out. By default, the statement stops running when the call times out.

> **Important**
> For DDL statements, we recommend continuing to run the statement after the call times out. When a DDL statement terminates before it is finished running, it can result in errors and possibly corrupted data structures.

Type: Boolean

Required: No

database (p. 17)

The name of the database.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 64.

Required: No

includeResultMetadata (p. 17)

A value that indicates whether to include metadata in the results.

Type: Boolean

Required: No

parameters (p. 17)

The parameters for the SQL statement.

> **Note**
> Array parameters are not supported.

Type: Array of SqlParameter (p. 35) objects

Required: No

resourceArn (p. 17)

The Amazon Resource Name (ARN) of the Aurora Serverless DB cluster.

Type: String

Length Constraints: Minimum length of 11. Maximum length of 100.

Required: Yes

resultSetOptions (p. 17)

Options that control how the result set is returned.

Type: ResultSetOptions (p. 34) object

Required: No

schema (p. 17)

The name of the database schema.

> **Note**
> Currently, the `schema` parameter isn't supported.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 64.

Required: No

secretArn (p. 17)

The name or ARN of the secret that enables access to the DB cluster.

Type: String

Length Constraints: Minimum length of 11. Maximum length of 100.

Required: Yes

sql (p. 17)

The SQL statement to run.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 65536.

Required: Yes

transactionId (p. 17)

The identifier of a transaction that was started by using the `BeginTransaction` operation. Specify the transaction ID of the transaction that you want to include the SQL statement in.

If the SQL statement is not part of a transaction, don't set this parameter.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 192.

Required: No

Response Syntax

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

{
   "columnMetadata": [
      {
         "arrayBaseColumnType": number,
         "isAutoIncrement": boolean,
         "isCaseSensitive": boolean,
         "isCurrency": boolean,
         "isSigned": boolean,
         "label": "string",
         "name": "string",
         "nullable": number,
         "precision": number,
         "scale": number,
         "schemaName": "string",
         "tableName": "string",
         "type": number,
         "typeName": "string"
      }
   ],
   "generatedFields": [
```

```
      {
         "arrayValue": {
            "arrayValues": [
               "ArrayValue"
            ],
            "booleanValues": [ boolean ],
            "doubleValues": [ number ],
            "longValues": [ number ],
            "stringValues": [ "string" ]
         },
         "blobValue": blob,
         "booleanValue": boolean,
         "doubleValue": number,
         "isNull": boolean,
         "longValue": number,
         "stringValue": "string"
      }
   ],
   "numberOfRecordsUpdated": number,
   "records": [
      [
         {
            "arrayValue": {
               "arrayValues": [
                  "ArrayValue"
               ],
               "booleanValues": [ boolean ],
               "doubleValues": [ number ],
               "longValues": [ number ],
               "stringValues": [ "string" ]
            },
            "blobValue": blob,
            "booleanValue": boolean,
            "doubleValue": number,
            "isNull": boolean,
            "longValue": number,
            "stringValue": "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.

columnMetadata (p. 19)

Metadata for the columns included in the results.

Type: Array of ColumnMetadata (p. 27) objects

generatedFields (p. 19)

Values for fields generated during the request.

Note
The `generatedFields` data isn't supported by Aurora PostgreSQL. To get the values of generated fields, use the `RETURNING` clause. For more information, see Returning Data From Modified Rows in the PostgreSQL documentation.

Type: Array of Field (p. 29) objects

numberOfRecordsUpdated (p. 19)

The number of records updated by the request.

Type: Long

records (p. 19)

The records returned by the SQL statement.

Type: Array of arrays of Field (p. 29) objects

Errors

BadRequestException

There is an error in the call or in a SQL statement.

HTTP Status Code: 400

ForbiddenException

There are insufficient privileges to make the call.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred.

HTTP Status Code: 500

ServiceUnavailableError

The service specified by the `resourceArn` parameter is not available.

HTTP Status Code: 503

StatementTimeoutException

The execution of the SQL statement timed out.

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

RollbackTransaction

Performs a rollback of a transaction. Rolling back a transaction cancels its changes.

Request Syntax

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

{
   "resourceArn": "string",
   "secretArn": "string",
   "transactionId": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

resourceArn (p. 22)

The Amazon Resource Name (ARN) of the Aurora Serverless DB cluster.

Type: String

Length Constraints: Minimum length of 11. Maximum length of 100.

Required: Yes

secretArn (p. 22)

The name or ARN of the secret that enables access to the DB cluster.

Type: String

Length Constraints: Minimum length of 11. Maximum length of 100.

Required: Yes

transactionId (p. 22)

The identifier of the transaction to roll back.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 192.

Required: Yes

Response Syntax

```
HTTP/1.1 200
```

```
Content-type: application/json

{
   "transactionStatus": "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.

transactionStatus (p. 22)

The status of the rollback operation.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 128.

Errors

BadRequestException

There is an error in the call or in a SQL statement.

HTTP Status Code: 400

ForbiddenException

There are insufficient privileges to make the call.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred.

HTTP Status Code: 500

NotFoundException

The `resourceArn`, `secretArn`, or `transactionId` value can't be found.

HTTP Status Code: 404

ServiceUnavailableError

The service specified by the `resourceArn` parameter is not available.

HTTP Status Code: 503

StatementTimeoutException

The execution of the SQL statement timed out.

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 RDS Data Service 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:

ArrayValue

Contains an array.

Contents

arrayValues

An array of arrays.

Type: Array of ArrayValue (p. 26) objects

Required: No

booleanValues

An array of Boolean values.

Type: Array of booleans

Required: No

doubleValues

An array of integers.

Type: Array of doubles

Required: No

longValues

An array of floating point numbers.

Type: Array of longs

Required: No

stringValues

An array of strings.

Type: Array of strings

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ColumnMetadata

Contains the metadata for a column.

Contents

arrayBaseColumnType

The type of the column.

Type: Integer

Required: No

isAutoIncrement

A value that indicates whether the column increments automatically.

Type: Boolean

Required: No

isCaseSensitive

A value that indicates whether the column is case-sensitive.

Type: Boolean

Required: No

isCurrency

A value that indicates whether the column contains currency values.

Type: Boolean

Required: No

isSigned

A value that indicates whether an integer column is signed.

Type: Boolean

Required: No

label

The label for the column.

Type: String

Required: No

name

The name of the column.

Type: String

Required: No

nullable

A value that indicates whether the column is nullable.

Type: Integer

Required: No

precision

The precision value of a decimal number column.

Type: Integer

Required: No

scale

The scale value of a decimal number column.

Type: Integer

Required: No

schemaName

The name of the schema that owns the table that includes the column.

Type: String

Required: No

tableName

The name of the table that includes the column.

Type: String

Required: No

type

The type of the column.

Type: Integer

Required: No

typeName

The database-specific data type of the column.

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

Field

Contains a value.

Contents

arrayValue

An array of values.

Type: ArrayValue (p. 26) object

Required: No

blobValue

A value of BLOB data type.

Type: Base64-encoded binary data object

Required: No

booleanValue

A value of Boolean data type.

Type: Boolean

Required: No

doubleValue

A value of double data type.

Type: Double

Required: No

isNull

A NULL value.

Type: Boolean

Required: No

longValue

A value of long data type.

Type: Long

Required: No

stringValue

A value of string data type.

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

Record

A record returned by a call.

Contents

values

The values returned in the record.

Type: Array of Value (p. 39) objects

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ResultFrame

The result set returned by a SQL statement.

Contents

records

The records in the result set.

Type: Array of Record (p. 31) objects

Required: No

resultSetMetadata

The result-set metadata in the result set.

Type: ResultSetMetadata (p. 33) 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

ResultSetMetadata

The metadata of the result set returned by a SQL statement.

Contents

columnCount

The number of columns in the result set.

Type: Long

Required: No

columnMetadata

The metadata of the columns in the result set.

Type: Array of ColumnMetadata (p. 27) objects

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ResultSetOptions

Options that control how the result set is returned.

Contents

decimalReturnType

A value that indicates how a field of `DECIMAL` type is represented in the response. The value of `STRING`, the default, specifies that it is converted to a String value. The value of `DOUBLE_OR_LONG` specifies that it is converted to a Long value if its scale is 0, or to a Double value otherwise.

Important
Conversion to Double or Long can result in roundoff errors due to precision loss. We recommend converting to String, especially when working with currency values.

Type: String

Valid Values: `STRING | DOUBLE_OR_LONG`

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

SqlParameter

A parameter used in a SQL statement.

Contents

name

The name of the parameter.

Type: String

Required: No

typeHint

A hint that specifies the correct object type for data type mapping. Possible values are as follows:

- `DATE` - The corresponding `String` parameter value is sent as an object of `DATE` type to the database. The accepted format is `YYYY-MM-DD`.
- `DECIMAL` - The corresponding `String` parameter value is sent as an object of `DECIMAL` type to the database.
- `JSON` - The corresponding `String` parameter value is sent as an object of `JSON` type to the database.
- `TIME` - The corresponding `String` parameter value is sent as an object of `TIME` type to the database. The accepted format is `HH:MM:SS[.FFF]`.
- `TIMESTAMP` - The corresponding `String` parameter value is sent as an object of `TIMESTAMP` type to the database. The accepted format is `YYYY-MM-DD HH:MM:SS[.FFF]`.
- `UUID` - The corresponding `String` parameter value is sent as an object of `UUID` type to the database.

Type: String

Valid Values: `JSON | UUID | TIMESTAMP | DATE | TIME | DECIMAL`

Required: No

value

The value of the parameter.

Type: Field (p. 29) 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

SqlStatementResult

The result of a SQL statement.

Important
This data type is deprecated.

Contents

numberOfRecordsUpdated

The number of records updated by a SQL statement.

Type: Long

Required: No

resultFrame

The result set of the SQL statement.

Type: ResultFrame (p. 32) 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

StructValue

A structure value returned by a call.

Contents

attributes

The attributes returned in the record.

Type: Array of Value (p. 39) objects

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

UpdateResult

The response elements represent the results of an update.

Contents

generatedFields

Values for fields generated during the request.

Type: Array of Field (p. 29) objects

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Value

Contains the value of a column.

Important
This data type is deprecated.

Contents

arrayValues

An array of column values.

Type: Array of Value (p. 39) objects

Required: No

bigIntValue

A value for a column of big integer data type.

Type: Long

Required: No

bitValue

A value for a column of BIT data type.

Type: Boolean

Required: No

blobValue

A value for a column of BLOB data type.

Type: Base64-encoded binary data object

Required: No

doubleValue

A value for a column of double data type.

Type: Double

Required: No

intValue

A value for a column of integer data type.

Type: Integer

Required: No

isNull

A NULL value.

Type: Boolean

Required: No

realValue

A value for a column of real data type.

Type: Float

Required: No

stringValue

A value for a column of string data type.

Type: String

Required: No

structValue

A value for a column of STRUCT data type.

Type: StructValue (p. 37) 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