# AuditData

BigQuery request and response messages for audit log. Note: `Table.schema` has been deprecated in favor of `Table.schemaJson`. `Table.schema` may continue to be present in your logs during this transition.

### JSON representation

```
{
  "jobCompletedEvent": {
    object (JobCompletedEvent (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditDa
  },
  "tableDataReadEvents": [
    {
      object (TableDataReadEvent (/logging/docs/reference/audit/bigquery/rest/Shared.Types/Auc
    }
  ],

  // Union field request can be only one of the following:
  "tableInsertRequest": {
    object (TableInsertRequest (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditD
  },
  "tableUpdateRequest": {
    object (TableUpdateRequest (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditD
  },
  "datasetListRequest": {
    object (DatasetListRequest (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditD
  },
  "datasetInsertRequest": {
    object (DatasetInsertRequest (/logging/docs/reference/audit/bigquery/rest/Shared.Types/Auc
  },
  "datasetUpdateRequest": {
    object (DatasetUpdateRequest (/logging/docs/reference/audit/bigquery/rest/Shared.Types/Auc
  },
  "jobInsertRequest": {
    object (JobInsertRequest (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditDat
  },
  "jobQueryRequest": {
    object (JobQueryRequest (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData
  },
  "jobGetQueryResultsRequest": {
    object (JobGetQueryResultsRequest (/logging/docs/reference/audit/bigquery/rest/Shared.Typ
  },
```

## JSON representation

```
  "tableDataListRequest": {
    object (TableDataListRequest (/logging/docs/reference/audit/bigquery/rest/Shared.Types/Aud
  },
  "setIamPolicyRequest": {
    object (SetIamPolicyRequest (/logging/docs/reference/audit/bigquery/rest/Shared.Types/Audi
  }
  // End of list of possible types for union field request.

  // Union field response can be only one of the following:
  "tableInsertResponse": {
    object (TableInsertResponse (/logging/docs/reference/audit/bigquery/rest/Shared.Types/Audi
  },
  "tableUpdateResponse": {
    object (TableUpdateResponse (/logging/docs/reference/audit/bigquery/rest/Shared.Types/Audi
  },
  "datasetInsertResponse": {
    object (DatasetInsertResponse (/logging/docs/reference/audit/bigquery/rest/Shared.Types/Au
  },
  "datasetUpdateResponse": {
    object (DatasetUpdateResponse (/logging/docs/reference/audit/bigquery/rest/Shared.Types/Au
  },
  "jobInsertResponse": {
    object (JobInsertResponse (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditDa
  },
  "jobQueryResponse": {
    object (JobQueryResponse (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditDat
  },
  "jobGetQueryResultsResponse": {
    object (JobGetQueryResultsResponse (/logging/docs/reference/audit/bigquery/rest/Shared.Ty
  },
  "jobQueryDoneResponse": {
    object (JobQueryDoneResponse (/logging/docs/reference/audit/bigquery/rest/Shared.Types/Aud
  },
  "policyResponse": {
    object (Policy (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Policy))
  }
  // End of list of possible types for union field response.
}
```

**Fields**

| | |
|---|---|
| `jobCompletedEvent` | object (**JobCompletedEvent** (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#JobCompletedEvent) )<br><br>A job completion event. |
| `tableDataRead Events[]` | object (**TableDataReadEvent** (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableDataReadEvent) )<br><br>Information about the table access events. |

Union field `request`. Request data for each BigQuery method. `request` can be only one of the following:

| | |
|---|---|
| `tableInsertRequest` | object (**TableInsertRequest** (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableInsertRequest) )<br><br>Table insert request. |
| `tableUpdateRequest` | object (**TableUpdateRequest** (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableUpdateRequest) )<br><br>Table update request. |
| `datasetListRequest` | object (**DatasetListRequest** (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#DatasetListRequest) )<br><br>Dataset list request. |
| `datasetInsertReques t` | object (**DatasetInsertRequest** (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#DatasetInsertRequest) )<br><br>Dataset insert request. |

**Fields**

| | |
|---|---|
| `datasetUpdateRequest` | object (**DatasetUpdateRequest** (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Dataset UpdateRequest) ) <br><br> Dataset update request. |
| `jobInsertRequest` | object (**JobInsertRequest** (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#JobInse rtRequest) ) <br><br> Job insert request. |
| `jobQueryRequest` | object (**JobQueryRequest** (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#JobQue ryRequest) ) <br><br> Job query request. |
| `jobGetQueryResultsRequest` | object (**JobGetQueryResultsRequest** (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#JobGet QueryResultsRequest) ) <br><br> Job get query results request. |
| `tableDataListRequest` | object (**TableDataListRequest** (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableDa taListRequest) ) <br><br> Table data-list request. |
| `setIamPolicyRequest` | object (**SetIamPolicyRequest** (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#SetIam PolicyRequest) ) <br><br> Iam policy request. |

Union field `response`. Response data for each BigQuery method. `response` can be only one of the following:

**Fields**

| | |
|---|---|
| `tableInsertResponse` | object ([TableInsertResponse](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableInsertResponse)) <br><br> Table insert response. |
| `tableUpdateResponse` | object ([TableUpdateResponse](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableUpdateResponse)) <br><br> Table update response. |
| `datasetInsertResponse` | object ([DatasetInsertResponse](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#DatasetInsertResponse)) <br><br> Dataset insert response. |
| `datasetUpdateResponse` | object ([DatasetUpdateResponse](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#DatasetUpdateResponse)) <br><br> Dataset update response. |
| `jobInsertResponse` | object ([JobInsertResponse](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#JobInsertResponse)) <br><br> Job insert response. |
| `jobQueryResponse` | object ([JobQueryResponse](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#JobQueryResponse)) <br><br> Job query response. |
| `jobGetQueryResultsResponse` | object ([JobGetQueryResultsResponse](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#JobGetQueryResultsResponse)) |

**Fields**

| | |
|---|---|
| | ) |
| | Job get query results response. |
| `jobQueryDoneRespons` `e` | object (JobQueryDoneResponse (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#JobQue ryDoneResponse) ) |
| | Deprecated: Job query-done response. Use this information for usage analysis. |
| `policyResponse` | object (Policy (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Policy)) |
| | Iam Policy. |

# TableInsertRequest

Table insert request.

**JSON representation**

```
{
  "resource": {
    object (Table (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Table))
  }
}
```

**Fields**

| | |
|---|---|
| `resource` | object (Table (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Table)) |
| | The new table. |

# Table

Describes a BigQuery table. See the Table (/bigquery/docs/reference/v2/tables) API resource for more details on individual fields. Note: `Table.schema` has been deprecated in favor of `Table.schemaJson`. `Table.schema` may continue to be present in your logs during this transition.

---

**JSON representation**

```
{
  "tableName": {
    object (TableName (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableN
  },
  "info": {
    object (TableInfo (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableIr
  },
  "schemaJson": string,
  "view": {
    object (TableViewDefinition (/logging/docs/reference/audit/bigquery/rest/Shared.Types/Audi
  },
  "expireTime": string,
  "createTime": string,
  "truncateTime": string,
  "updateTime": string,
  "encryption": {
    object (EncryptionInfo (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#
  }
}
```

---

**Fields**

| tableName | object (TableName (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableName) ) |
|---|---|
| | The name of the table. |
| info | object (TableInfo (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableInfo) ) |
| | User-provided metadata for the table. |

**Fields**

| | |
|---|---|
| `schemaJson` | `string`<br><br>A JSON representation of the table's schema. |
| `view` | object ([TableViewDefinition](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableViewDefinition))<br><br>If present, this is a virtual table defined by a SQL query. |
| `expireTime` | string ([Timestamp](https://developers.google.com/protocol-buffers/docs/reference/google.protobuf#google.protobuf.Timestamp) `format`)<br><br>The expiration date for the table, after which the table is deleted and the storage reclaimed. If not present, the table persists indefinitely.<br><br>A timestamp in RFC3339 UTC "Zulu" format, with nanosecond resolution and up to nine fractional digits. Examples: `"2014-10-02T15:01:23Z"` and `"2014-10-02T15:01:23.045123456Z"`. |
| `createTime` | string ([Timestamp](https://developers.google.com/protocol-buffers/docs/reference/google.protobuf#google.protobuf.Timestamp) `format`)<br><br>The time the table was created.<br><br>A timestamp in RFC3339 UTC "Zulu" format, with nanosecond resolution and up to nine fractional digits. Examples: `"2014-10-02T15:01:23Z"` and `"2014-10-02T15:01:23.045123456Z"`. |

## Fields

| | |
|---|---|
| `truncateTime` | `string (`<u>`Timestamp`</u> `(https://developers.google.com/protocol-buffers/docs/reference/google.protobuf#google.protobuf.Timestamp)` `format)` <br><br> The time the table was last truncated by an operation with a `writeDisposition` of `WRITE_TRUNCATE`. <br><br> A timestamp in RFC3339 UTC "Zulu" format, with nanosecond resolution and up to nine fractional digits. Examples: `"2014-10-02T15:01:23Z"` and `"2014-10-02T15:01:23.045123456Z"`. |
| `updateTime` | `string (`<u>`Timestamp`</u> `(https://developers.google.com/protocol-buffers/docs/reference/google.protobuf#google.protobuf.Timestamp)` `format)` <br><br> The time the table was last modified. <br><br> A timestamp in RFC3339 UTC "Zulu" format, with nanosecond resolution and up to nine fractional digits. Examples: `"2014-10-02T15:01:23Z"` and `"2014-10-02T15:01:23.045123456Z"`. |
| `encryption` | `object (`<u>`EncryptionInfo`</u> `(/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#EncryptionInfo)` `)` <br><br> The table encryption information. Set when non-default encryption is used. |

# TableName

The fully-qualified name for a table.

## JSON representation

```
{
  "projectId": string,
  "datasetId": string,
  "tableId": string
}
```

**Fields**

| | |
|---|---|
| `projectId` | `string` |
| | The project ID. |
| `datasetId` | `string` |
| | The dataset ID within the project. |
| `tableId` | `string` |
| | The table ID of the table within the dataset. |

# TableInfo

User-provided metadata for a table.

**JSON representation**

```
{
  "friendlyName": string,
  "description": string,
  "labels": {
    string: string,
    ...
  }
}
```

**Fields**

| | |
|---|---|
| `friendlyName` | `string` |
| | A short name for the table, such as`"Analytics Data - Jan 2011"`. |
| `description` | `string` |
| | A long description, perhaps several paragraphs, describing the table contents in detail. |
| `labels` | `map (key: string, value: string)` |
| | Labels provided for the table. |

**Fields**

An object containing a list of `"key": value` pairs. Example: `{ "name": "wrench", "mass": "1.3kg", "count": "3" }`.

# TableViewDefinition

Describes a virtual table defined by a SQL query.

**JSON representation**

```
{
  "query": string
}
```

**Fields**

query               string

SQL query defining the view.

# EncryptionInfo

Describes encryption properties for a table or a job

**JSON representation**

```
{
  "kmsKeyName": string
}
```

**Fields**

kmsKeyName          string

unique identifier for cloud kms key

# TableUpdateRequest

Table update request.

## JSON representation

```
{
  "resource": {
    object (Table (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Table))
  }
}
```

## Fields

| | |
|---|---|
| `resource` | `object (`Table `(/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Table))`<br><br>The table to be updated. |

# DatasetListRequest

Dataset list request.

## JSON representation

```
{
  "listAll": boolean
}
```

## Fields

| | |
|---|---|
| `listAll` | `boolean`<br><br>Whether to list all datasets, including hidden ones. |

# DatasetInsertRequest

Dataset insert request.

**JSON representation**

```
{
  "resource": {
    object (Dataset (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Dataset))
  }
}
```

**Fields**

| | |
|---|---|
| `resource` | `object (`<u>`Dataset`</u>` (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Dataset ) )`<br><br>The dataset to be inserted. |

# Dataset

BigQuery dataset information. See the <u>Dataset</u> (/bigquery/docs/reference/v2/datasets) API resource for more details on individual fields.

**JSON representation**

```
{
  "datasetName": {
    object (DatasetName (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Data
  },
  "info": {
    object (DatasetInfo (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Data
  },
  "createTime": string,
  "updateTime": string,
  "acl": {
    object (BigQueryAcl (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#BigC
  },
  "defaultTableExpireDuration": string
}
```

## Fields

| | |
|---|---|
| `datasetName` | `object (`[`DatasetName`](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#DatasetName)`)`<br><br>The name of the dataset. |
| `info` | `object (`[`DatasetInfo`](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#DatasetInfo)`)`<br><br>User-provided metadata for the dataset. |
| `createTime` | `string (`[`Timestamp`](https://developers.google.com/protocol-buffers/docs/reference/google.protobuf#google.protobuf.Timestamp)` format)`<br><br>The time the dataset was created.<br><br>A timestamp in RFC3339 UTC "Zulu" format, with nanosecond resolution and up to nine fractional digits. Examples: `"2014-10-02T15:01:23Z"` and `"2014-10-02T15:01:23.045123456Z"`. |
| `updateTime` | `string (`[`Timestamp`](https://developers.google.com/protocol-buffers/docs/reference/google.protobuf#google.protobuf.Timestamp)` format)`<br><br>The time the dataset was last modified.<br><br>A timestamp in RFC3339 UTC "Zulu" format, with nanosecond resolution and up to nine fractional digits. Examples: `"2014-10-02T15:01:23Z"` and `"2014-10-02T15:01:23.045123456Z"`. |
| `acl` | `object (`[`BigQueryAcl`](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#BigQueryAcl)`)`<br><br>The access control list for the dataset. |

## Fields

| | |
|---|---|
| `defaultTableExpireDuration` | string (<u>Duration</u> (https://developers.google.com/protocol-buffers/docs/reference/google.protobuf#google.protobuf.Duration) `format`)<br><br>If this field is present, each table that does not specify an expiration time is assigned an expiration time by adding this duration to the table's `createTime`. If this field is empty, there is no default table expiration time.<br><br>A duration in seconds with up to nine fractional digits, ending with '`s`'. Example: "`3.5s`". |

# DatasetName

The fully-qualified name for a dataset.

### JSON representation

```
{
  "projectId": string,
  "datasetId": string
}
```

## Fields

| | |
|---|---|
| `projectId` | string<br><br>The project ID. |
| `datasetId` | string<br><br>The dataset ID within the project. |

# DatasetInfo

User-provided metadata for a dataset.

**JSON representation**

```
{
  "friendlyName": string,
  "description": string,
  "labels": {
    string: string,
    ...
  }
}
```

**Fields**

| | |
|---|---|
| `friendlyName` | `string`<br><br>A short name for the dataset, such as`"Analytics Data 2011"`. |
| `description` | `string`<br><br>A long description, perhaps several paragraphs, describing the dataset contents in detail. |
| `labels` | `map (key: string, value: string)`<br><br>Labels provided for the dataset.<br><br>An object containing a list of `"key": value` pairs. Example: `{ "name": "wrench", "mass": "1.3kg", "count": "3" }`. |

# BigQueryAcl

An access control list.

**JSON representation**

```
{
  "entries": [
    {
      object (Entry (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Entry))
    }
  ]
}
```

**Fields**

| | |
|---|---|
| `entries[]` | object (Entry (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Entry)) |
| | Access control entry list. |

# Entry

Access control entry.

**JSON representation**

```
{
  "role": string,
  "groupEmail": string,
  "userEmail": string,
  "domain": string,
  "specialGroup": string,
  "viewName": {
    object (TableName (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableN
  }
}
```

**Fields**

| | |
|---|---|
| `role` | string |
| | The granted role, which can be `READER`, `WRITER`, or `OWNER`. |
| `groupEmail` | string |
| | Grants access to a group identified by an email address. |
| `userEmail` | string |
| | Grants access to a user identified by an email address. |
| `domain` | string |
| | Grants access to all members of a domain. |

### Fields

| specialGroup | string |
| --- | --- |
| | Grants access to special groups. Valid groups are `PROJECT_OWNERS`, `PROJECT_READERS`, `PROJECT_WRITERS` and `ALL_AUTHENTICATED_USERS`. |
| viewName | object (`TableName` (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableName)) |
| | Grants access to a BigQuery View. |

# DatasetUpdateRequest

Dataset update request.

### JSON representation

```
{
  "resource": {
    object (Dataset (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Dataset))
  }
}
```

### Fields

| resource | object (`Dataset` (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Dataset)) |
| --- | --- |
| | The dataset to be updated. |

# JobInsertRequest

Job insert request.

**JSON representation**

```
{
  "resource": {
    object (Job (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Job))
  }
}
```

**Fields**

| | |
|---|---|
| `resource` | `object (`Job`` (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Job))` <br><br> Job insert request. |

# Job

Describes a job.

**JSON representation**

```
{
  "jobName": {
    object (JobName (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#JobName
  },
  "jobConfiguration": {
    object (JobConfiguration (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditDat
  },
  "jobStatus": {
    object (JobStatus (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#JobSta
  },
  "jobStatistics": {
    object (JobStatistics (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#J
  }
}
```

**Fields**

| | |
|---|---|
| `jobName` | object (`JobName` (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#JobName) ) Job name. |
| `jobConfiguration` | object (`JobConfiguration` (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#JobConfiguration) ) Job configuration. |
| `jobStatus` | object (`JobStatus` (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#JobStatus) ) Job status. |
| `jobStatistics` | object (`JobStatistics` (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#JobStatistics) ) Job statistics. |

# JobName

The fully-qualified name for a job.

**JSON representation**

```
{
  "projectId": string,
  "jobId": string,
  "location": string
}
```

**Fields**

| | |
|---|---|
| `projectId` | string<br><br>The project ID. |
| `jobId` | string<br><br>The job ID within the project. |
| `location` | string<br><br>The job location. |

# JobConfiguration

Job configuration information. See the <u>Jobs</u> (/bigquery/docs/reference/v2/jobs) API resource for more details on individual fields.

**JSON representation**

```
{
  "dryRun": boolean,
  "labels": {
    string: string,
    ...
  },

  // Union field configuration can be only one of the following:
  "query": {
    object (Query (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Query))
  },
  "load": {
    object (Load (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Load))
  },
  "extract": {
    object (Extract (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Extract))
  },
  "tableCopy": {
    object (TableCopy (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableC
  }
```

**JSON representation**

```
  // End of list of possible types for union field configuration.
}
```

**Fields**

| | |
|---|---|
| `dryRun` | `boolean`<br><br>If true, don't actually run the job. Just check that it would run. |
| `labels` | `map (key: string, value: string)`<br><br>Labels provided for the job.<br><br>An object containing a list of `"key": value` pairs. Example: `{ "name": "wrench", "mass": "1.3kg", "count": "3" }`. |

Union field `configuration`. Job configuration information. `configuration` can be only one of the following:

| | |
|---|---|
| `query` | `object (`<u>Query</u><br>`(/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Query))`<br><br>Query job information. |
| `load` | `object (`<u>Load</u><br>`(/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Load))`<br><br>Load job information. |
| `extract` | `object (`<u>Extract</u><br>`(/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Extract)`<br>`)`<br><br>Extract job information. |
| `tableCopy` | `object (`<u>TableCopy</u><br>`(/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableCopy)`<br>`)`<br><br>TableCopy job information. |

# Query

Describes a query job, which executes a SQL-like query.

---

**JSON representation**

```
{
  "query": string,
  "destinationTable": {
    object (TableName (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableN
  },
  "createDisposition": string,
  "writeDisposition": string,
  "defaultDataset": {
    object (DatasetName (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Data
  },
  "tableDefinitions": [
    {
      object (TableDefinition (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditDa
    }
  ],
  "queryPriority": string,
  "destinationTableEncryption": {
    object (EncryptionInfo (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#
  },
  "statementType": string
}
```

---

**Fields**

| | |
|---|---|
| **query** | **string** |
| | The SQL query to run. |
| **destinationTable** | **object (TableName** **(/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableN ame)** **)** |
| | The table where results are written. |
| **createDisposition** | **string** |

**Fields**

| | |
|---|---|
| | Describes when a job is allowed to create a table: `CREATE_IF_NEEDED`, `CREATE_NEVER`. |
| `writeDisposition` | `string` <br><br> Describes how writes affect existing tables: `WRITE_TRUNCATE`, `WRITE_APPEND`, `WRITE_EMPTY`. |
| `defaultDataset` | `object (`[`DatasetName`](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#DatasetName)`)` <br><br> If a table name is specified without a dataset in a query, this dataset will be added to table name. |
| `tableDefinitions[]` | `object (`[`TableDefinition`](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableDefinition)`)` <br><br> Describes data sources outside BigQuery, if needed. |
| `queryPriority` | `string` <br><br> Describes the priority given to the query: `QUERY_INTERACTIVE` or `QUERY_BATCH`. |
| `destinationTableEncryption` | `object (`[`EncryptionInfo`](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#EncryptionInfo)`)` <br><br> Result table encryption information. Set when non-default encryption is used. |
| `statementType` | `string` <br><br> Type of the statement (e.g. SELECT, INSERT, CREATE_TABLE, CREATE_MODEL..) |

# TableDefinition

Describes an external data source used in a query.

**JSON representation**

```
{
  "name": string,
  "sourceUris": [
    string
  ]
}
```

**Fields**

| name | string |
| --- | --- |
| | Name of the table, used in queries. |
| sourceUris[] | string |
| | Google Cloud Storage URIs for the data to be imported. |

# Load

Describes a load job, which loads data from an external source via the import pipeline.

**JSON representation**

```
{
  "sourceUris": [
    string
  ],
  "schemaJson": string,
  "destinationTable": {
    object (TableName (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableN
  },
  "createDisposition": string,
  "writeDisposition": string,
  "destinationTableEncryption": {
    object (EncryptionInfo (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#
  }
}
```

**Fields**

| | |
|---|---|
| `sourceUris[]` | `string`<br><br>URIs for the data to be imported. Only Google Cloud Storage URIs are supported. |
| `schemaJson` | `string`<br><br>The table schema in JSON format representation of a TableSchema. |
| `destinationTable` | `object (`[`TableName`](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableName)`)`<br><br>The table where the imported data is written. |
| `createDisposition` | `string`<br><br>Describes when a job is allowed to create a table: `CREATE_IF_NEEDED`, `CREATE_NEVER`. |
| `writeDisposition` | `string`<br><br>Describes how writes affect existing tables: `WRITE_TRUNCATE`, `WRITE_APPEND`, `WRITE_EMPTY`. |
| `destinationTableEncryption` | `object (`[`EncryptionInfo`](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#EncryptionInfo)`)`<br><br>Result table encryption information. Set when non-default encryption is used. |

# Extract

Describes an extract job, which exports data to an external source via the export pipeline.

**JSON representation**

```
{
  "destinationUris": [
    string
```

**JSON representation**

```
  ],
  "sourceTable": {
    object (TableName (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableN
  }
}
```

**Fields**

| | |
|---|---|
| `destinationUris[]` | `string` |
| | Google Cloud Storage URIs where extracted data should be written. |
| `sourceTable` | `object (TableName` `(/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableName)` `)` |
| | The source table. |

# TableCopy

Describes a copy job, which copies an existing table to another table.

**JSON representation**

```
{
  "sourceTables": [
    {
      object (TableName (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Tabl
    }
  ],
  "destinationTable": {
    object (TableName (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableN
  },
  "createDisposition": string,
  "writeDisposition": string,
  "destinationTableEncryption": {
    object (EncryptionInfo (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#
```

## JSON representation

```
    }
}
```

## Fields

| | |
|---|---|
| `sourceTables[]` | object (`TableName` (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableName) ) <br><br> Source tables. |
| `destinationTable` | object (`TableName` (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableName) ) <br><br> Destination table. |
| `createDisposition` | string <br><br> Describes when a job is allowed to create a table: `CREATE_IF_NEEDED`, `CREATE_NEVER`. |
| `writeDisposition` | string <br><br> Describes how writes affect existing tables: `WRITE_TRUNCATE`, `WRITE_APPEND`, `WRITE_EMPTY`. |
| `destinationTableEncryption` | object (`EncryptionInfo` (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#EncryptionInfo) ) <br><br> Result table encryption information. Set when non-default encryption is used. |

# JobStatus

Running state of a job.

**JSON representation**

```
{
  "state": string,
  "error": {
    object (Status (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Status))
  },
  "additionalErrors": [
    {
      object (Status (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Status))
    }
  ]
}
```

**Fields**

| | |
|---|---|
| state | string<br><br>State of a job: PENDING, RUNNING, or DONE. |
| error | object (Status (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Status))<br><br>If the job did not complete successfully, this field describes why. |
| additionalErrors[] | object (Status (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Status))<br><br>Errors encountered during the running of the job. Do not necessarily mean that the job has completed or was unsuccessful. |

## Status

The Status type defines a logical error model that is suitable for different programming environments, including REST APIs and RPC APIs. It is used by gRPC (https://github.com/grpc). Each Status message contains three pieces of data: error code, error message, and error details.

You can find out more about this error model and how to work with it in the <u>API Design Guide</u>
 (https://cloud.google.com/apis/design/errors).

---

**JSON representation**

```
{
  "code": integer,
  "message": string,
  "details": [
    {
      "@type": string,
      field1: ...,
      ...
    }
  ]
}
```

---

**Fields**

| | |
|---|---|
| `code` | `integer`<br><br>The status code, which should be an enum value of `google.rpc.Code`. |
| `message` | `string`<br><br>A developer-facing error message, which should be in English. Any user-facing error message should be localized and sent in the `google.rpc.Status.details` (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Status.FIELDS.details) field, or localized by the client. |
| `details[]` | `object`<br><br>A list of messages that carry the error details. There is a common set of message types for APIs to use.<br><br>An object containing fields of an arbitrary type. An additional field `"@type"` contains a URI identifying the type. Example: `{ "id": 1234, "@type": "types.example.com/standard/id" }`. |

# JobStatistics

Job statistics that may change after a job starts.

## JSON representation

```
{
  "createTime": string,
  "startTime": string,
  "endTime": string,
  "totalProcessedBytes": string,
  "totalBilledBytes": string,
  "billingTier": integer,
  "totalSlotMs": string,
  "reservationUsage": [
    {
      object (ReservationResourceUsage (/logging/docs/reference/audit/bigquery/rest/Shared.Ty
    }
  ],
  "referencedTables": [
    {
      object (TableName (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Tabl
    }
  ],
  "totalTablesProcessed": integer,
  "referencedViews": [
    {
      object (TableName (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Tabl
    }
  ],
  "totalViewsProcessed": integer,
  "queryOutputRowCount": string,
  "totalLoadOutputBytes": string
}
```

## Fields

| | |
|---|---|
| createTime | string (Timestamp (https://developers.google.com/protocol-buffers/docs/reference/google.protobuf#google.protobuf.Timestamp) format)<br><br>Time when the job was created.<br><br>A timestamp in RFC3339 UTC "Zulu" format, with nanosecond resolution and up to nine fractional digits. Examples: `"2014-10-02T15:01:23Z"` and `"2014-` |

## Fields

| | |
|---|---|
| | 10-02T15:01:23.045123456Z". |
| startTime | string (<u>Timestamp</u> (https://developers.google.com/protocol-buffers/docs/reference/google.protobuf#google.protobuf.Timestamp) format) <br><br> Time when the job started. <br><br> A timestamp in RFC3339 UTC "Zulu" format, with nanosecond resolution and up to nine fractional digits. Examples: "2014-10-02T15:01:23Z" and "2014-10-02T15:01:23.045123456Z". |
| endTime | string (<u>Timestamp</u> (https://developers.google.com/protocol-buffers/docs/reference/google.protobuf#google.protobuf.Timestamp) format) <br><br> Time when the job ended. <br><br> A timestamp in RFC3339 UTC "Zulu" format, with nanosecond resolution and up to nine fractional digits. Examples: "2014-10-02T15:01:23Z" and "2014-10-02T15:01:23.045123456Z". |
| totalProcessedBytes | string (<u>int64</u> (https://developers.google.com/discovery/v1/type-format) format) <br><br> Total bytes processed for a job. |
| totalBilledBytes | string (<u>int64</u> (https://developers.google.com/discovery/v1/type-format) format) <br><br> Processed bytes, adjusted by the job's CPU usage. |
| billingTier | integer <br><br> The tier assigned by CPU-based billing. |
| totalSlotMs | string (<u>int64</u> (https://developers.google.com/discovery/v1/type-format) format) <br><br> The total number of slot-ms consumed by the query job. |
| reservationUsage[] (deprecated) | object (<u>ReservationResourceUsage</u> (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#ReservationResourceUsage) |

**Fields**

)

⚠ This item is deprecated!

Deprecated as of 12/15/2022.

| `referencedTables[]` | object ([TableName](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableName)) |
|---|---|

The first N tables accessed by the query job. Older queries that reference a large number of tables may not have all of their tables in this list. You can use the totalTablesProcessed count to know how many total tables were read in the query. For new queries, there is currently no limit.

| `totalTablesProcessed` | integer |
|---|---|

Total number of unique tables referenced in the query.

| `referencedViews[]` | object ([TableName](/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableName)) |
|---|---|

The first N views accessed by the query job. Older queries that reference a large number of views may not have all of their views in this list. You can use the totalTablesProcessed count to know how many total tables were read in the query. For new queries, there is currently no limit.

| `totalViewsProcessed` | integer |
|---|---|

Total number of unique views referenced in the query.

| `queryOutputRowCount` | string ([int64](https://developers.google.com/discovery/v1/type-format) format) |
|---|---|

Number of output rows produced by the query job.

| `totalLoadOutputBytes` | string ([int64](https://developers.google.com/discovery/v1/type-format) format) |
|---|---|

Total bytes loaded for an import job.

# ReservationResourceUsage

Job resource usage breakdown by reservation.

## JSON representation

```
{
  "name": string,
  "slotMs": string
}
```

## Fields

| name | string |
| --- | --- |
| | Reservation name or "unreserved" for on-demand resources usage. |
| slotMs | string (int64 (https://developers.google.com/discovery/v1/type-format) format) |
| | Total slot milliseconds used by the reservation for a particular job. |

# JobQueryRequest

Job query request.

## JSON representation

```
{
  "query": string,
  "maxResults": integer,
  "defaultDataset": {
    object (DatasetName (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Data
  },
  "projectId": string,
  "dryRun": boolean
}
```

**Fields**

| query | `string` |
|---|---|
| | The query. |

| maxResults | `integer` (`uint32` (https://developers.google.com/discovery/v1/type-format) `format`) |
|---|---|
| | The maximum number of results. |

| defaultDataset | `object` (`DatasetName` (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Dataset Name) `)` |
|---|---|
| | The default dataset for tables that do not have a dataset specified. |

| projectId | `string` |
|---|---|
| | Project that the query should be charged to. |

| dryRun | `boolean` |
|---|---|
| | If true, don't actually run the job. Just check that it would run. |

# JobGetQueryResultsRequest

Job getQueryResults request.

**JSON representation**

```
{
  "maxResults": integer,
  "startRow": string
}
```

**Fields**

| | |
|---|---|
| `maxResults` | `integer` (<u>uint32</u> (https://developers.google.com/discovery/v1/type-format) `format`) <br><br> Maximum number of results to return. |
| `startRow` | `string` <br><br> Zero-based row number at which to start. |

# TableDataListRequest

Table data-list request.

**JSON representation**

```
{
  "startRow": string,
  "maxResults": integer
}
```

**Fields**

| | |
|---|---|
| `startRow` | `string` <br><br> Starting row offset. |
| `maxResults` | `integer` (<u>uint32</u> (https://developers.google.com/discovery/v1/type-format) `format`) <br><br> Maximum number of results to return. |

# SetIamPolicyRequest

Request message for `SetIamPolicy` method.

**JSON representation**

## JSON representation

```
{
  "resource": string,
  "policy": {
    object (Policy (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Policy))
  },
  "updateMask": string
}
```

## Fields

| | |
|---|---|
| `resource` | `string`<br><br>REQUIRED: The resource for which the policy is being specified. See Resource names (https://cloud.google.com/apis/design/resourceNames) for the appropriate value for this field. |
| `policy` | `object (`Policy<br>(/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Policy)`)`<br><br>REQUIRED: The complete policy to be applied to the `resource`. The size of the policy is limited to a few 10s of KB. An empty policy is a valid policy but certain Google Cloud services (such as Projects) might reject them. |
| `updateMask` | `string (`FieldMask<br>(https://developers.google.com/protocol-buffers/docs/reference/google.protobuf#google.protobuf.FieldMask)<br>`format)`<br><br>OPTIONAL: A FieldMask specifying which fields of the policy to modify. Only the fields in the mask will be modified. If no mask is provided, the following default mask is used:<br><br>`paths: "bindings, etag"`<br><br>This is a comma-separated list of fully qualified names of fields. Example: `"user.displayName,photo"`. |

# Policy

An Identity and Access Management (IAM) policy, which specifies access controls for Google Cloud resources.

A `Policy` is a collection of `bindings`. A `binding` binds one or more `members`, or principals, to a single `role`. Principals can be user accounts, service accounts, Google groups, and domains (such as G Suite). A `role` is a named list of permissions; each `role` can be an IAM predefined role or a user-created custom role.

For some types of Google Cloud resources, a `binding` can also specify a `condition`, which is a logical expression that allows access to a resource only if the expression evaluates to `true`. A condition can add constraints based on attributes of the request, the resource, or both. To learn which resources support conditions in their IAM policies, see the <u>IAM documentation</u> (https://cloud.google.com/iam/help/conditions/resource-policies).

**JSON example:**

```
{
  "bindings": [
    {
      "role": "roles/resourcemanager.organizationAdmin",
      "members": [
        "user:mike@example.com",
        "group:admins@example.com",
        "domain:google.com",
        "serviceAccount:my-project-id@appspot.gserviceaccount.com"
      ]
    },
    {
      "role": "roles/resourcemanager.organizationViewer",
      "members": [
        "user:eve@example.com"
      ],
      "condition": {
        "title": "expirable access",
        "description": "Does not grant access after Sep 2020",
        "expression": "request.time < timestamp('2020-10-01T00:00:00.000Z')"
      }
    }
  ],
  "etag": "BwWWja0YfJA=",
  "version": 3
```

```
        }
```

**YAML example:**

```
    bindings:
    - members:
      - user:mike@example.com
      - group:admins@example.com
      - domain:google.com
      - serviceAccount:my-project-id@appspot.gserviceaccount.com
      role: roles/resourcemanager.organizationAdmin
    - members:
      - user:eve@example.com
      role: roles/resourcemanager.organizationViewer
      condition:
        title: expirable access
        description: Does not grant access after Sep 2020
        expression: request.time < timestamp('2020-10-01T00:00:00.000Z')
    etag: BwWWja0YfJA=
    version: 3
```

For a description of IAM and its features, see the IAM documentation
 (https://cloud.google.com/iam/docs/).

---

**JSON representation**

```
{
  "version": integer,
  "bindings": [
    {
      object (Binding (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Binding
    }
  ],
  "auditConfigs": [
    {
      object (AuditConfig (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#A
    }
  ],
  "etag": string
}
```

---

## Fields

| | |
|---|---|
| `version` | `integer`<br><br>Specifies the format of the policy.<br><br>Valid values are `0`, `1`, and `3`. Requests that specify an invalid value are rejected.<br><br>Any operation that affects conditional role bindings must specify version `3`. This requirement applies to the following operations:<br><br>• Getting a policy that includes a conditional role binding<br><br>• Adding a conditional role binding to a policy<br><br>• Changing a conditional role binding in a policy<br><br>• Removing any role binding, with or without a condition, from a policy that includes conditions<br><br>**Important:** If you use IAM Conditions, you must include the `etag` field whenever you call `setIamPolicy`. If you omit this field, then IAM allows you to overwrite a version `3` policy with a version `1` policy, and all of the conditions in the version `3` policy are lost.<br><br>If a policy does not include any conditions, operations on that policy may specify any valid version or leave the field unset.<br><br>To learn which resources support conditions in their IAM policies, see the IAM documentation (https://cloud.google.com/iam/help/conditions/resource-policies). |
| `bindings[]` | object (Binding (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Binding ) )<br><br>Associates a list of `members`, or principals, with a `role`. Optionally, may specify a `condition` that determines how and when the `bindings` are applied. Each of the `bindings` must contain at least one principal.<br><br>The `bindings` in a `Policy` can refer to up to 1,500 principals; up to 250 of these principals can be Google groups. Each occurrence of a principal counts towards these limits. For example, if the `bindings` grant 50 different roles to `user:alice@example.com`, and not to any other principal, then you can add another 1,450 principals to the `bindings` in the `Policy`. |

## Fields

| | |
|---|---|
| `auditConfigs[]` | object (<u>AuditConfig</u> (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#AuditConfig) ) <br><br> Specifies cloud audit logging configuration for this policy. |
| `etag` | string (<u>bytes</u> (https://developers.google.com/discovery/v1/type-format) `format`) <br><br> `etag` is used for optimistic concurrency control as a way to help prevent simultaneous updates of a policy from overwriting each other. It is strongly suggested that systems make use of the `etag` in the read-modify-write cycle to perform policy updates in order to avoid race conditions: An `etag` is returned in the response to `getIamPolicy`, and systems are expected to put that etag in the request to `setIamPolicy` to ensure that their change will be applied to the same version of the policy. <br><br> **Important:** If you use IAM Conditions, you must include the `etag` field whenever you call `setIamPolicy`. If you omit this field, then IAM allows you to overwrite a version **3** policy with a version **1** policy, and all of the conditions in the version **3** policy are lost. <br><br> A base64-encoded string. |

# Binding

Associates `members`, or principals, with a `role`.

### JSON representation

```
{
  "role": string,
  "members": [
    string
  ],
  "condition": {
    object (Expr (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Expr))
  }
}
```

## Fields

| | |
|---|---|
| role | string |
| | Role that is assigned to the list of `members`, or principals. For example, `roles/viewer`, `roles/editor`, or `roles/owner`. |
| members[] | string |

Specifies the principals requesting access for a Google Cloud resource. `members` can have the following values:

- `allUsers`: A special identifier that represents anyone who is on the internet; with or without a Google account.

- `allAuthenticatedUsers`: A special identifier that represents anyone who is authenticated with a Google account or a service account. Does not include identities that come from external identity providers (IdPs) through identity federation.

- `user:{emailid}`: An email address that represents a specific Google account. For example, `alice@example.com` .

- `serviceAccount:{emailid}`: An email address that represents a Google service account. For example, `my-other-app@appspot.gserviceaccount.com`.

- `serviceAccount:{projectid}.svc.id.goog[{namespace}/{kubernetes-sa}]`: An identifier for a Kubernetes service account (https://cloud.google.com/kubernetes-engine/docs/how-to/kubernetes-service-accounts). For example, `my-project.svc.id.goog[my-namespace/my-kubernetes-sa]`.

- `group:{emailid}`: An email address that represents a Google group. For example, `admins@example.com`.

- `domain:{domain}`: The G Suite domain (primary) that represents all the users of that domain. For example, `google.com` or `example.com`.

- `deleted:user:{emailid}?uid={uniqueid}`: An email address (plus unique identifier) representing a user that has been recently deleted. For example, `alice@example.com?uid=123456789012345678901`. If the user is recovered, this value reverts to `user:{emailid}` and the recovered user retains the role in the binding.

**Fields**

- `deleted:serviceAccount:{emailid}?uid={uniqueid}`: An email address (plus unique identifier) representing a service account that has been recently deleted. For example, `my-other-app@appspot.gserviceaccount.com?uid=123456789012345678901`. If the service account is undeleted, this value reverts to `serviceAccount:{emailid}` and the undeleted service account retains the role in the binding.

- `deleted:group:{emailid}?uid={uniqueid}`: An email address (plus unique identifier) representing a Google group that has been recently deleted. For example, `admins@example.com?uid=123456789012345678901`. If the group is recovered, this value reverts to `group:{emailid}` and the recovered group retains the role in the binding.

| | |
|---|---|
| `condition` | `object` (Expr (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Expr)) <br><br> The condition that is associated with this binding. <br><br> If the condition evaluates to `true`, then this binding applies to the current request. <br><br> If the condition evaluates to `false`, then this binding does not apply to the current request. However, a different role binding might grant the same role to one or more of the principals in this binding. <br><br> To learn which resources support conditions in their IAM policies, see the IAM documentation (https://cloud.google.com/iam/help/conditions/resource-policies). |

# Expr

Represents a textual expression in the Common Expression Language (CEL) syntax. CEL is a C-like expression language. The syntax and semantics of CEL are documented at https://github.com/google/cel-spec (https://github.com/google/cel-spec).

Example (Comparison):

```
title: "Summary size limit"
description: "Determines if a summary is less than 100 chars"
```

```
expression: "document.summary.size() < 100"
```

Example (Equality):

```
title: "Requestor is owner"
description: "Determines if requestor is the document owner"
expression: "document.owner == request.auth.claims.email"
```

Example (Logic):

```
title: "Public documents"
description: "Determine whether the document should be publicly visible"
expression: "document.type != 'private' && document.type != 'internal'"
```

Example (Data Manipulation):

```
title: "Notification string"
description: "Create a notification string with a timestamp."
expression: "'New message received at ' + string(document.create_time)"
```

The exact variables and functions that may be referenced within an expression are determined by the service that evaluates it. See the service documentation for additional information.

---

### JSON representation

```
{
  "expression": string,
  "title": string,
  "description": string,
  "location": string
}
```

**Fields**

| | |
|---|---|
| `expression` | `string` |
| | Textual representation of an expression in Common Expression Language syntax. |
| `title` | `string` |
| | Optional. Title for the expression, i.e. a short string describing its purpose. This can be used e.g. in UIs which allow to enter the expression. |
| `description` | `string` |
| | Optional. Description of the expression. This is a longer text which describes the expression, e.g. when hovered over it in a UI. |
| `location` | `string` |
| | Optional. String indicating the location of the expression for error reporting, e.g. a file name and a position in the file. |

# AuditConfig

Specifies the audit configuration for a service. The configuration determines which permission types are logged, and what identities, if any, are exempted from logging. An AuditConfig must have one or more AuditLogConfigs.

If there are AuditConfigs for both `allServices` and a specific service, the union of the two AuditConfigs is used for that service: the log_types specified in each AuditConfig are enabled, and the exemptedMembers in each AuditLogConfig are exempted.

Example Policy with multiple AuditConfigs:

```
{
  "auditConfigs": [
    {
      "service": "allServices",
      "auditLogConfigs": [
        {
```

```
          "logType": "DATA_READ",
          "exemptedMembers": [
            "user:jose@example.com"
          ]
        },
        {
          "logType": "DATA_WRITE"
        },
        {
          "logType": "ADMIN_READ"
        }
      ]
    },
    {
      "service": "sampleservice.googleapis.com",
      "auditLogConfigs": [
        {
          "logType": "DATA_READ"
        },
        {
          "logType": "DATA_WRITE",
          "exemptedMembers": [
            "user:aliya@example.com"
          ]
        }
      ]
    }
  ]
}
```

For sampleservice, this policy enables DATA_READ, DATA_WRITE and ADMIN_READ logging. It also exempts `jose@example.com` from DATA_READ logging, and `aliya@example.com` from DATA_WRITE logging.

---

### JSON representation

```
{
  "service": string,
  "auditLogConfigs": [
    {
      object (AuditLogConfig (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditDat
    }
```

**JSON representation**

```
   ]
}
```

**Fields**

| | |
|---|---|
| `service` | `string`<br><br>Specifies a service that will be enabled for audit logging. For example, `storage.googleapis.com`, `cloudsql.googleapis.com`. `allServices` is a special value that covers all services. |
| `auditLogConfigs[]` | `object (`**AuditLogConfig** (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#AuditLogConfig)`)`<br><br>The configuration for logging of each type of permission. |

# AuditLogConfig

Provides the configuration for logging a type of permissions. Example:

```
{
  "auditLogConfigs": [
    {
      "logType": "DATA_READ",
      "exemptedMembers": [
        "user:jose@example.com"
      ]
    },
    {
      "logType": "DATA_WRITE"
    }
  ]
}
```

This enables 'DATA_READ' and 'DATA_WRITE' logging, while exempting jose@example.com
 (mailto:jose@example.com) from DATA_READ logging.

**JSON representation**

```
{
  "logType": enum (LogType (/logging/docs/reference/audit/bigquery/rest/Shared.Types/LogType)),
  "exemptedMembers": [
    string
  ]
}
```

**Fields**

| | |
|---|---|
| `logType` | enum (LogType (/logging/docs/reference/audit/bigquery/rest/Shared.Types/LogType)) <br><br> The log type that this config enables. |
| `exemptedMembers[]` | `string` <br><br> Specifies the identities that do not cause logging for this type of permission. Follows the same format of **Binding.members** (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Binding.FIELDS.members) . |

# TableInsertResponse

Table insert response.

**JSON representation**

```
{
  "resource": {
    object (Table (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Table))
  }
}
```

**Fields**

| | |
|---|---|
| `resource` | object (Table (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Table)) |
| | Final state of the inserted table. |

# TableUpdateResponse

Table update response.

**JSON representation**

```
{
  "resource": {
    object (Table (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Table))
  }
}
```

**Fields**

| | |
|---|---|
| `resource` | object (Table (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Table)) |
| | Final state of the updated table. |

# DatasetInsertResponse

Dataset insert response.

**JSON representation**

```
{
  "resource": {
    object (Dataset (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Dataset))
  }
}
```

**Fields**

| | |
|---|---|
| `resource` | object (Dataset (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Dataset)) <br><br> Final state of the inserted dataset. |

# DatasetUpdateResponse

Dataset update response.

**JSON representation**

```
{
  "resource": {
    object (Dataset (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Dataset))
  }
}
```

**Fields**

| | |
|---|---|
| `resource` | object (Dataset (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Dataset)) <br><br> Final state of the updated dataset. |

# JobInsertResponse

Job insert response.

**JSON representation**

**JSON representation**

```
{
  "resource": {
    object (Job (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Job))
  }
}
```

**Fields**

| | |
|---|---|
| `resource` | object (Job (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Job)) <br><br> Job insert response. |

# JobQueryResponse

Job query response.

**JSON representation**

```
{
  "totalResults": string,
  "job": {
    object (Job (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Job))
  }
}
```

**Fields**

| | |
|---|---|
| `totalResults` | string <br><br> The total number of rows in the full query result set. |
| `job` | object (Job (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Job)) <br><br> Information about the queried job. |

# JobGetQueryResultsResponse

Job getQueryResults response.

## JSON representation

```
{
  "totalResults": string,
  "job": {
    object (Job (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Job))
  }
}
```

## Fields

| | |
|---|---|
| **totalResults** | **string**<br><br>Total number of results in query results. |
| **job** | **object** (**Job** (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Job))<br><br>The job that was created to run the query. It completed if `job.status.state` is `DONE`. It failed if `job.status.errorResult` is also present. |

# JobQueryDoneResponse

Job getQueryDone response.

## JSON representation

```
{
  "job": {
    object (Job (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Job))
  }
}
```

**Fields**

| | |
|---|---|
| `job` | object (`Job` (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Job)) |
| | The job and status information. The job completed if `job.status.state` is `DONE`. |

# JobCompletedEvent

Query job completed event.

**JSON representation**

```
{
  "eventName": string,
  "job": {
    object (Job (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Job))
  }
}
```

**Fields**

| | |
|---|---|
| `eventName` | string |
| | Name of the event. |
| `job` | object (`Job` (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#Job)) |
| | Job information. |

# TableDataReadEvent

Table data read event. Only present for tables, not views, and is only included in the log record for the project that owns the table.

## JSON representation

```
{
  "tableName": {
    object (TableName (/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableN
  },
  "referencedFields": [
    string
  ]
}
```

## Fields

| | |
|---|---|
| `tableName` | `object (`TableName `(/logging/docs/reference/audit/bigquery/rest/Shared.Types/AuditData#TableName) )`<br><br>Name of the accessed table. |
| `referencedFields[]` | `string`<br><br>A list of referenced fields. This information is not included by default. To enable this in the logs, please contact BigQuery support or open a bug in the BigQuery issue tracker. |