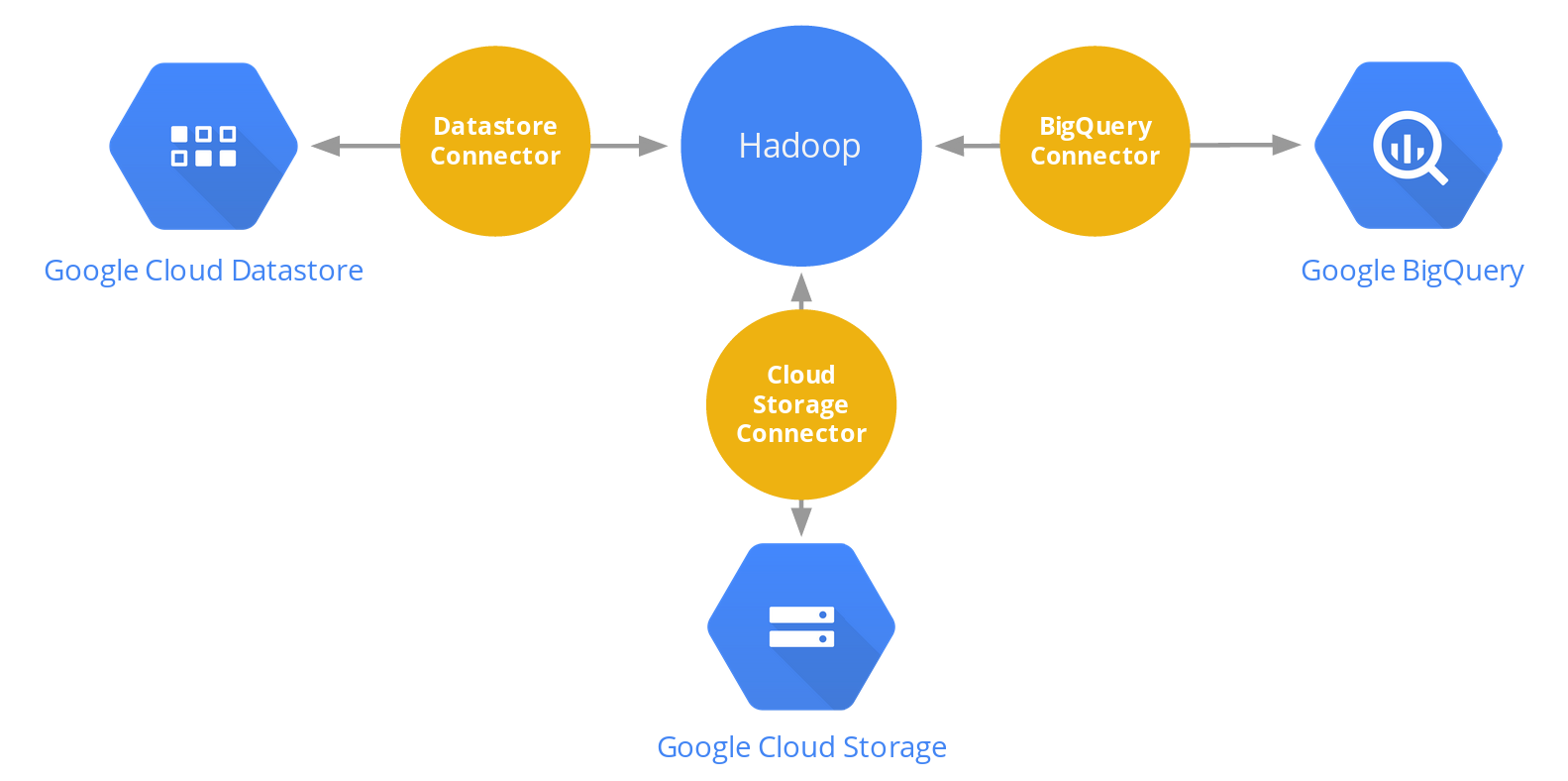
Google BigQuery from Pentaho Client Tools

Introduction

BigQuery is Google's serverless, highly scalable, low cost enterprise data warehouse designed to make all your data analysts productive. Because there is no infrastructure to manage, you can focus on analyzing data to find meaningful insights using familiar SQL and you don't need a database administrator.

* BigQuery enables you to analyze all your data by creating a logical data warehouse over managed, columnar storage as well as data from object storage, and spreadsheets.



* BigQuery makes it easy to securely share insights within your organization and beyond as datasets, queries, spreadsheets and reports.
* BigQuery allows organizations to capture and analyze data in real-time using its powerful streaming ingestion capability so that your insights are always current.
* BigQuery is free for up to 1TB of data analyzed each month and 10GB of data stored.

Pre-requisites

The following pre-requisites are required:

* Pentaho Client Tools with optional BA Pentaho Server.
* Installation of Google BigQuery JDBC drivers
* Google Cloud Platform Account
* Service Account Authentication
* Uploading Dataset

Step 1: Installation of Google BigQuery JDBC Drivers

Google has partnered with Simba Technologies Inc, a Magnitude Software Company to provide ODBC and JDBC drivers that leverage the power of BigQuery's Standard SQL.

The intent of these drivers is to help users connect the power of Google BigQuery with existing tooling and infrastructure that does not have native integration. To download the

1. Enter:

https://cloud.google.com/bigquery/partners/simba-drivers/

1. Click on the link:

JDBC 4.2-compatible

1. Save the drivers
2. Unzip and copy the Following .jar files to:

|  |  |
| --- | --- |
| CLIENT TOOL | PATH |
| Data Integration | C:\Pentaho\design-tools\data-integration\lib |
| Report Designer | C:\Pentaho\design-tools\report-designer\lib\jdbc |
| Metadata Editor | C:\Pentaho\design-tools\metadata-editor\lib |
| Schema Workbench | C:\Pentaho\design-tools\schema-workbench\drivers |
|  |  |
| Pentaho Server | C:\Pentaho\pentaho-server\tomcat\webapps\pentaho\WEB-INF\lib |

List of .jar files

* GoogleBigQueryJDBC42.jar
* google-http-client-1.23.0.jar
* google-http-client-jackson2-1.23.0.jar
* google-oauth-client-1.23.0.jar
* google-api-client-1.23.0.jar
* google-api-services-bigquery-v2-rev377-1.23.0.jar
* Jackson-core-2.1.3.jar (not required)

**

*Either rename or delete:* Jackson-core-2.1.3.jar

Once configured the Google BigQuery connection will appear in the list of JDBC connections.

Step 2: Enable Google BigQuery API

You can access BigQuery by using a web UI or a command-line tool, or by making calls to the BigQuery REST API using a variety of client libraries such as Java, .NET, or Python. There are also a variety of third-party tools that you can use to interact with BigQuery, such as visualizing the data or loading the data.

Enabling the Google BigQuery API can be achieved via the following steps:

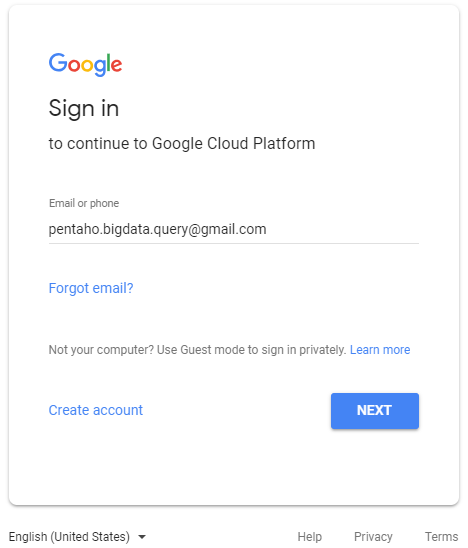
1. Connect with the browser to:

https://console.developers.google.com/apis/library

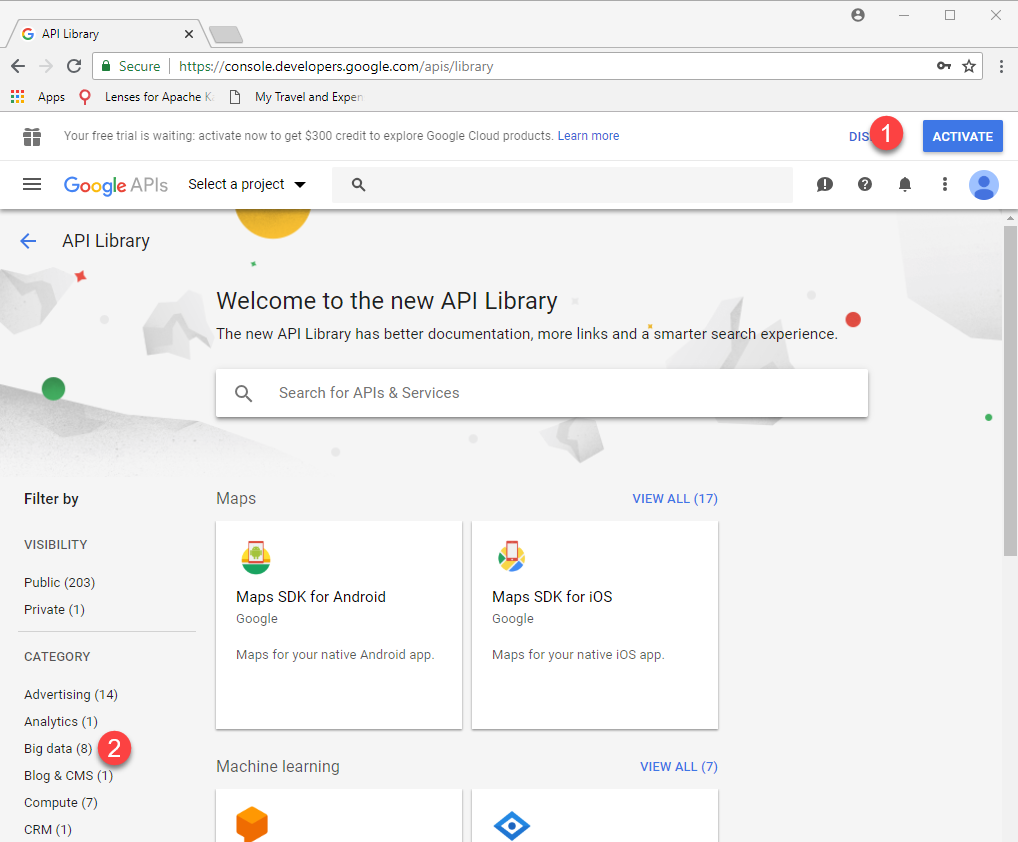
1. Login with the following Google Account that will be used to connect:

**eMail:** pentaho.bigdata.query@gmail.com

**Password:** BigQuery01!



The Google API Library main page will be shown:



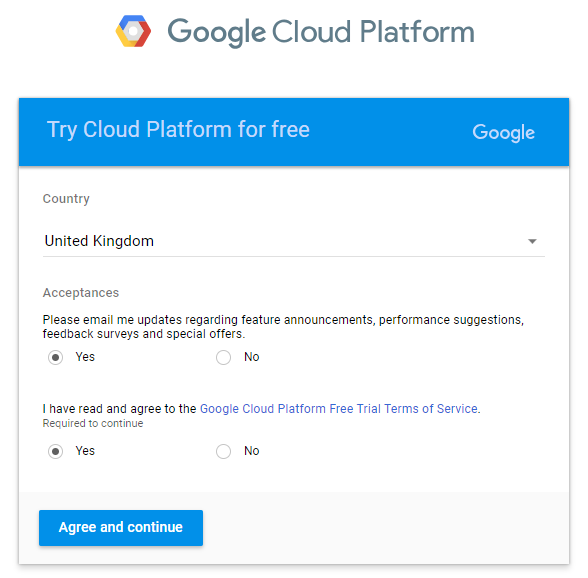
*If this is the first time you’re setting up the account, you will need to ACTIVATE.*

*You will require your credit card details. The account will not be charged unless you manually upgrade to a paid account.*

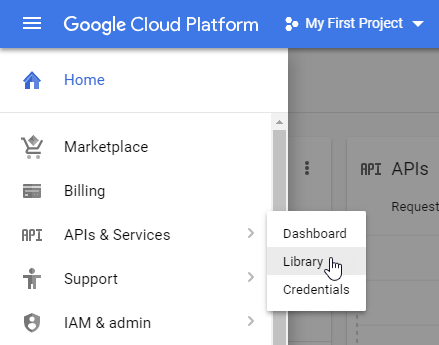
For further details of the Google Free Tier:

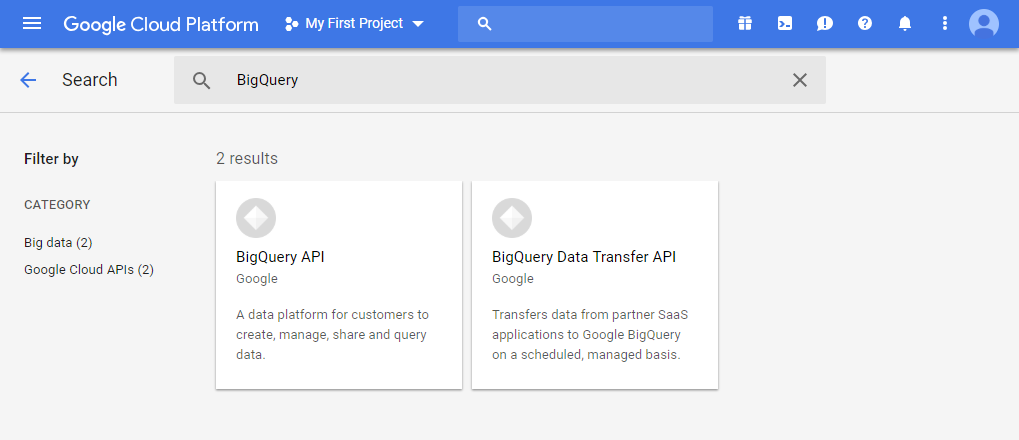
https://cloud.google.com/free/

https://cloud.google.com/bigquery/pricing#free

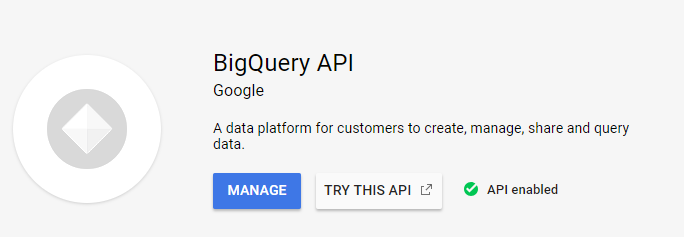


1. Accept the Terms
2. From the Account Dashboard, select APIs & Services > Library.





1. Click on Big data category or search for BigQuery
2. Click on “BigQuery API”
3. Ensure that the BigQuery is enabled.

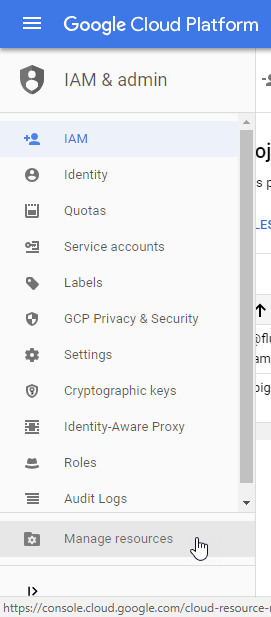


Step 3: Creating and Managing a Google BigQuery Project

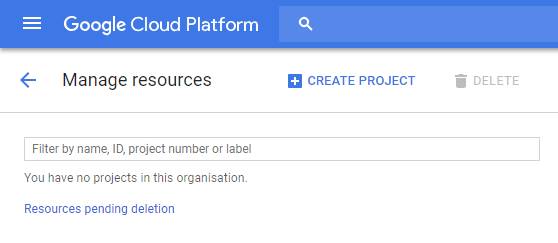
Google Cloud Platform projects form the basis for creating, enabling, and using all GCP services including managing APIs, enabling billing, adding and removing collaborators, and managing permissions for GCP resources.

To create a new project:

1. Go to the GCP Console Manage resources page.



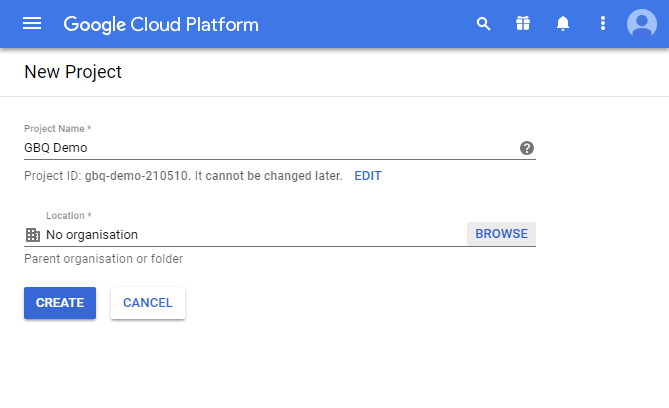
1. Select the option Manage Resources
2. On the drop-down at the top of the page, select the organization in which you want to create a project. (Not applicable as Individual account)
3. Click Create Project



1. In the New Project window that appears, enter:

GBQ Demo

as the project name and select a billing account as applicable.



If you want to add the project to a folder, enter the folder name in the Location box.

1. When you're finished entering new project details, click Create.

Opening the web UI

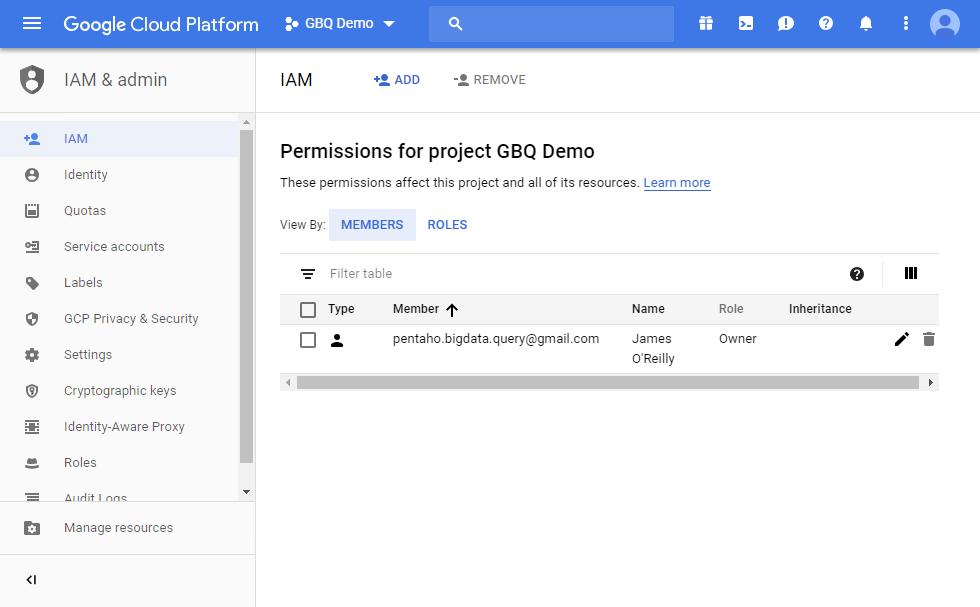
To open the web UI, enter the following URL in your browser and replace project\_id with the id of the project you're opening in the web UI:

https://bigquery.cloud.google.com/spry-utility-210510 (project id)

Identity and Access Management

Google Cloud Platform offers Identity and Access Management (IAM), which lets you give more granular access to specific Google Cloud Platform resources and prevents unwanted access to other resources. IAM lets you adopt the security principle of least privilege, so you grant only the necessary access to your resources.

IAM lets you control who (users) has what access (roles) to which resources by setting IAM policies. IAM policies grant specific role(s) to a user giving the user certain permissions.



Service Accounts

A service account is a special Google account that belongs to your application or a virtual machine (VM), instead of to an individual end user. Your application uses the service account to call the Google API of a service, so that the users aren't directly involved.

For example, a Compute Engine VM may run as a service account, and that account can be given permissions to access the resources it needs. This way the service account is the identity of the service, and the service account's permissions control which resources the service can access.

A service account is identified by its email address, which is unique to the account.

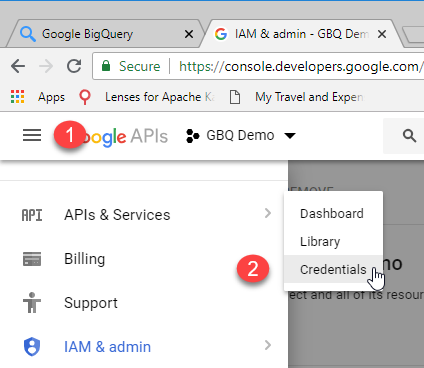
Service account keys

Each service account is associated with a key pair, which is managed by Google Cloud Platform (GCP). It is used for service-to-service authentication within GCP. Google rotates the keys approximately once a week.

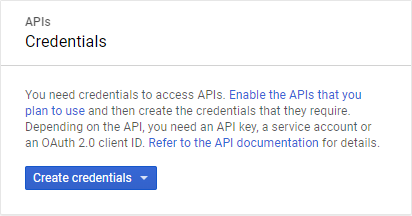
You may optionally create one or more external key pairs for use from outside GCP (for example, for use with Application Default Credentials). When you create a new key pair, you download the private key (which is not retained by Google). With external keys, you are responsible for security of the private key and other management operations such as key rotation. External keys can be managed by the IAM API, gcloud command-line tool, or the Service Accounts page in the Google Cloud Platform Console. You can create up to 10 service account keys per service account to facilitate key rotation.

To create a Service Account:

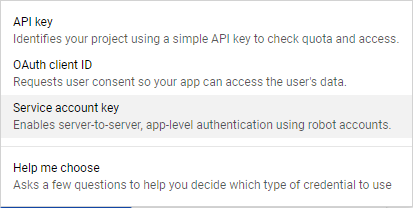
1. In the Google Cloud Platform Console, select Credentials



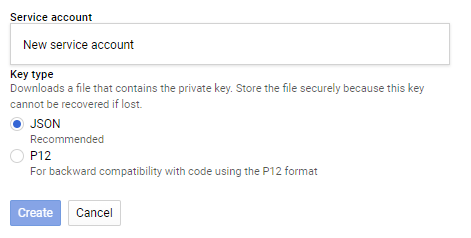
1. Click on Create Credentials



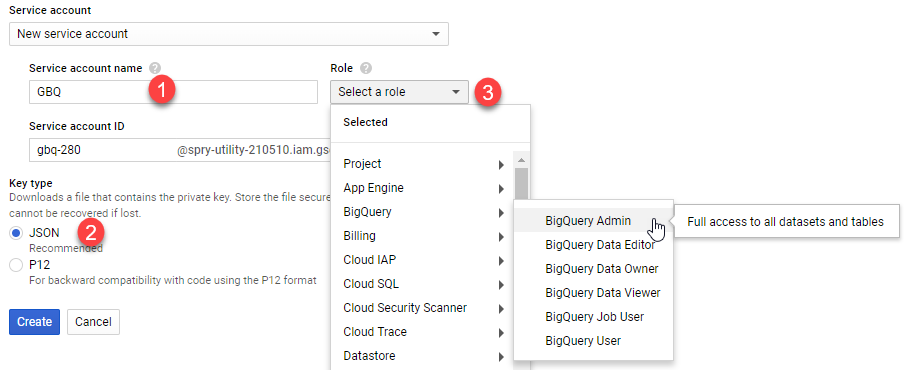
1. Select Service account key



1. Click the drop-down box below *Service account*, then click **New service account**.

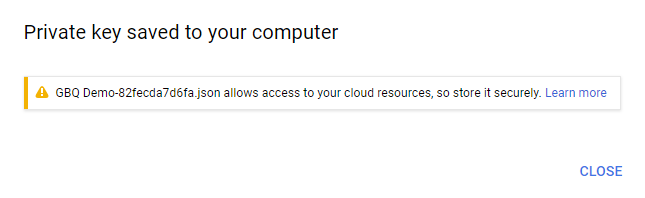


1. Enter a name for the service account in **Name**.



1. Choose a Cloud Storage Role that grants the service account the desired level of access.
2. Use the default Service account ID or generate a different one.
3. Select the **Key type**: **JSON**
4. Click Create.

A Service account created window is displayed and the private key for the Key type you selected is downloaded automatically. If you selected a P12 key, the private key's password ("notasecret") is displayed.



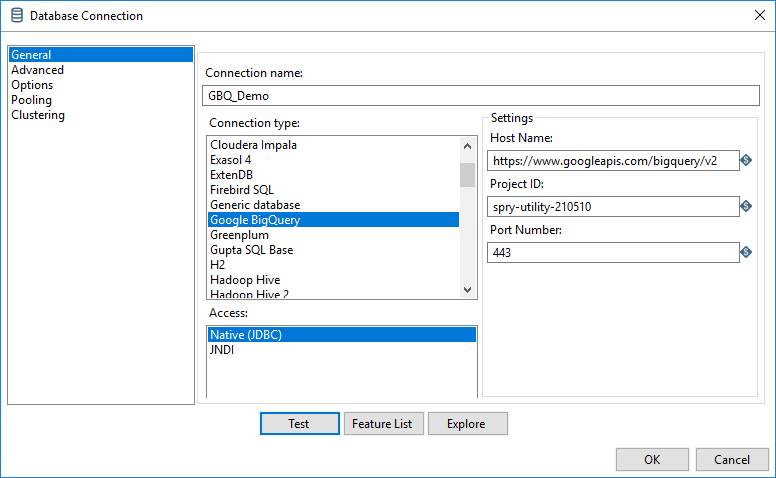
1. Save the key to:

C:\Private-key

1. Click Close.

Using a Google Service Account

You can configure the driver to authenticate the connection with a Google service account. The service account can handle the authentication process so that no user input is required.



You must provide a Google service account email address and the full path to a private key file for the service account. You can download the private key file from the Google API console web page. For more information about OAuth authentication using a service account, see "Using OAuth 2.0 for Server to Server Applications" in the Google Identity Platform documentation:

https://developers.google.com/identity/protocols/OAuth2ServiceAccount

To configure service account authentication:

* Set the OAuthType property to 0.
* Set the ProjectID property to the name of your BigQuery project.
* Set the OAuthServiceAcctEmail property to your Google service account email address.
* Set the OAuthPvtKeyPath property to the full path to the key file that is used to authenticate the service account email address. This parameter supports keys in .pl2 or .json format.

For example, the following connection URL authenticates the connection using a service account:

Hostname:https://www.googleapis.com/bigquery/v2:443;  
ProjectId=spry-utility-210510;OAuthType=0;  
OAuthServiceAcctEmail=gbq-280@spry-utility-210510.iam.gserviceaccount.com;  
OAuthPvtKeyPath=C:\Private-key\GBQ Demo-82fecda7d6fa.json;

Step 4. Validating Google BigQuery

To ensure the Google BigQuery API is successfully activated, a simple test can be executed via the Google BigQuery Web Interface. Google provides a set of samples that can be used for the validation of the BigQuery connectivity.

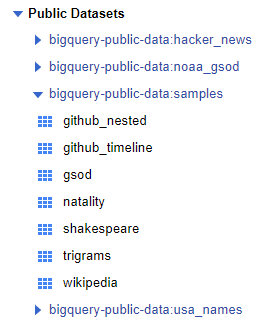
To validate the activation of the BigQuery API for the account defined in the previous section:

1. Enter:

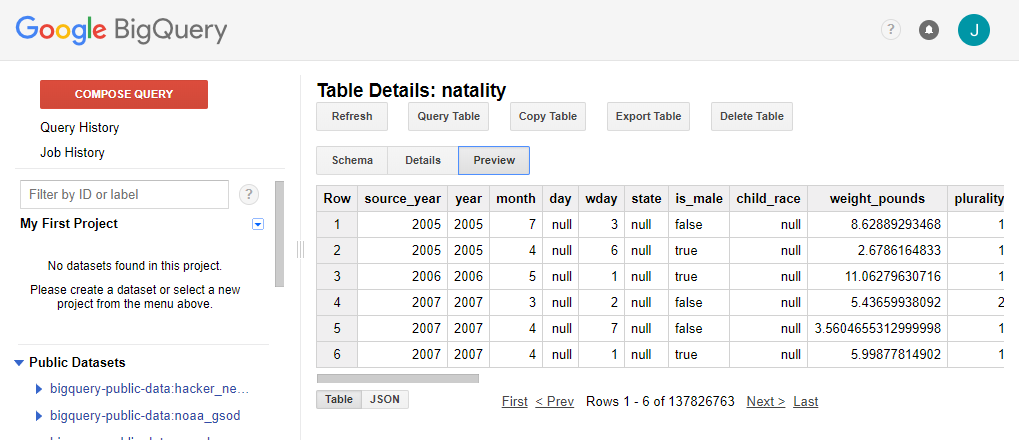
https://bigquery.cloud.google.com/spry-utility-210510

1. Expand:

bigquery-public-data:samples



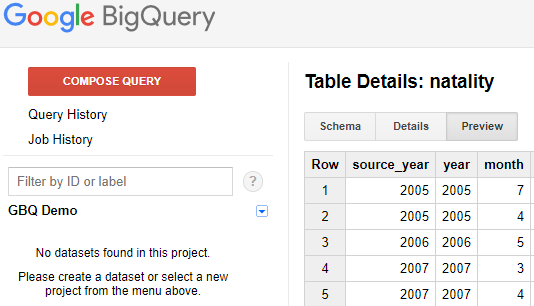
1. Select the **natality** table.
2. Click on Preview to view the data.



Query a public dataset

The BigQuery web UI provides an interface to query tables, including public datasets offered by BigQuery.

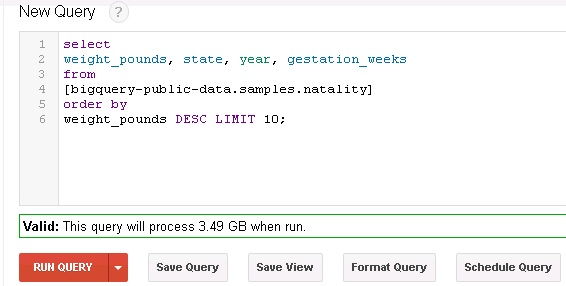
1. Click on the Compose Query red button



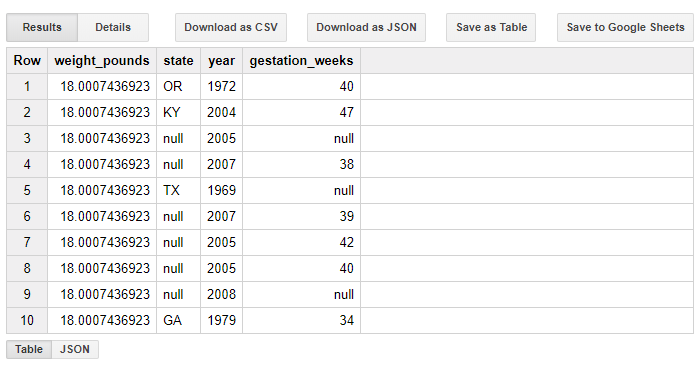
1. Copy and paste the following query into the New Query text area:

#standardSQL  
SELECT  
 weight\_pounds, state, year, gestation\_weeks  
FROM  
 [bigquery-public-data.samples.natality]  
ORDER BY weight\_pounds DESC LIMIT 10;

Click on the RUN QUERY button



The results can easily be exported:



*You can browse the schema of other public datasets by clicking* bigquery-public-data:samples *in the left-hand navigation of the web UI. The expanded list of titles are all the public tables you can query against.*

Load data into a table

Next, we'll load custom data into a table and run a query against it.

The custom data contains approximately 7 MB of data about popular names, and it is provided by the US Social Security Administration.

1. Download the baby names zip file.

http://www.ssa.gov/OACT/babynames/names.zip

1. Unzip the file onto your hard drive.

The zip file contains a read me file that describes the dataset schema.

1. Open the file named yob2014.txt in Notepad++.

The file is a comma-separated value (CSV) file with the following three columns: name, sex (M or F), and number of children with that name. The file has no header row.

Create a dataset

Next, create a dataset in the web UI to hold the data:

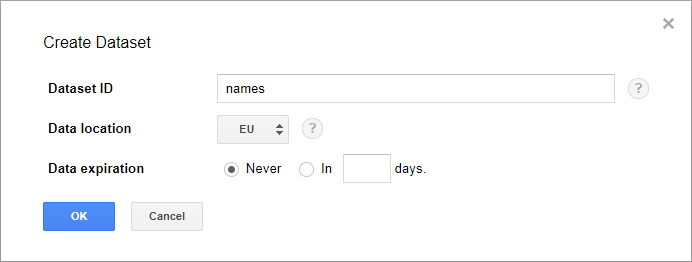
1. Go to the BigQuery web UI.

https://bigquery.cloud.google.com

1. Click the down arrow icon down arrow icon next to your project name in the navigation, then click **Create new dataset**.
2. Input the following name for the dataset ID.

names

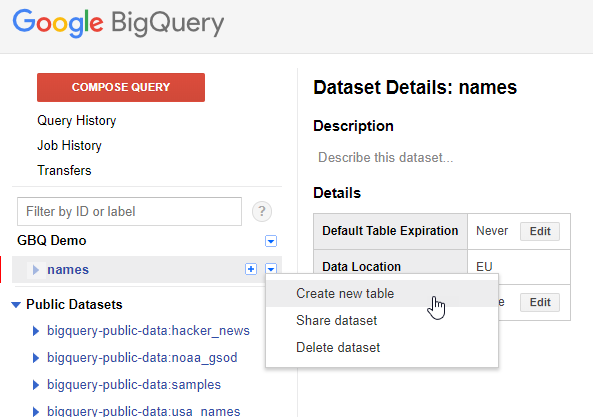
*Dataset IDs are unique on a per-project basis, so if***names***are already listed under your project name in the navigation, append a number to the name to make it unique. Click the question mark icon to see ID limitations.*

1. Select the location where your data is going to be stored.
2. You specify a location for storing your BigQuery data when you create a dataset. After you create the dataset, the location cannot be changed.  
3. Click OK.

Load the data into a new table

Next, load the data into a new table:

1. In the navigation, hover on the **names** dataset ID that you just created.
2. Click the down arrow icon down arrow icon image next to the ID and click Create new table.

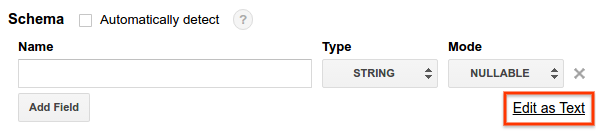


Use the default values for all settings unless otherwise indicated.

1. Under Source Data, click the Choose file button. Navigate to the data you unzipped in the step above and select the yob2017.txt file.
2. Under Destination Table, enter the following value for the destination table name.

names\_2017

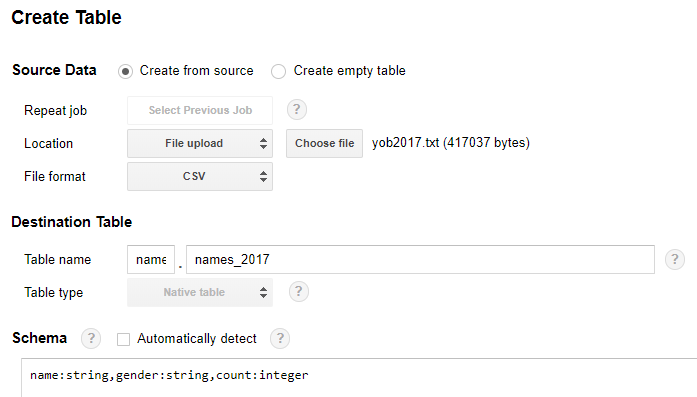
1. In the Schema section, click the Edit as Text link.



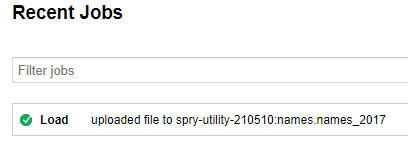
1. Then replace the contents of the Schema input area with the following schema:

name:string,gender:string,count:integer

1. Click the Create Table button.



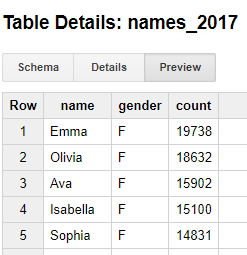
Wait for BigQuery to create the table and load the data. While BigQuery loads the data, a **(loading)** string displays after your table name in the navigation. The string disappears after the data has been fully loaded.



Preview the table

After the **(loading)** string disappears, you can access the table. To preview the first few rows of the data:

1. Select names\_2017 in the navigation.
2. Click Preview in the Table Details:names\_2017 section.



Query the table

Now that you've loaded custom data into a table, you can run queries against it.

1. Click the Compose query button.
2. Copy and paste the following query into the New Query text area.

#standardSQL  
SELECT  
 name, count  
FROM  
 [spry-utility-210510:names.names\_2017]  
WHERE  
 gender = 'M'  
Click the Run query button. The query displays the top 5 men's names for the year of data you loaded into the table.

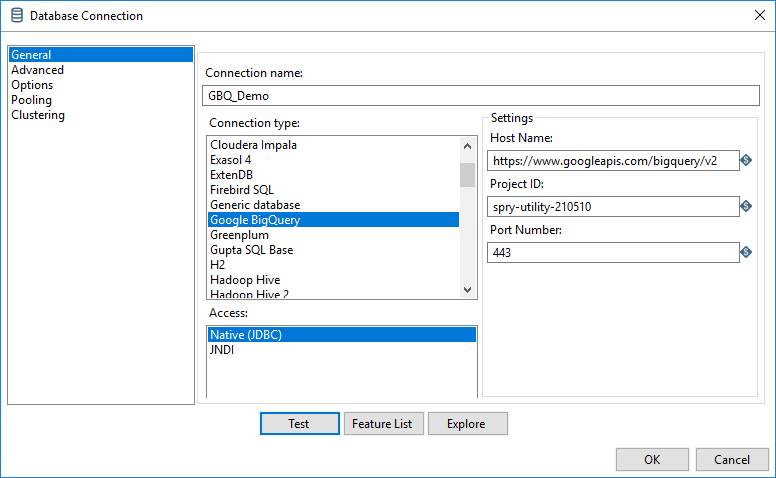
Clean up

To avoid incurring charges to your Google Cloud Platform account for the resources used:

1. Go to the BigQuery web UI.
2. In the navigation, hover on the names dataset you created.
3. Click the down arrow icon down arrow image next to your dataset name in the navigation, then click Delete dataset.
4. In the Delete dataset dialog box, confirm the delete command by typing the name of your dataset (names) and clicking OK.

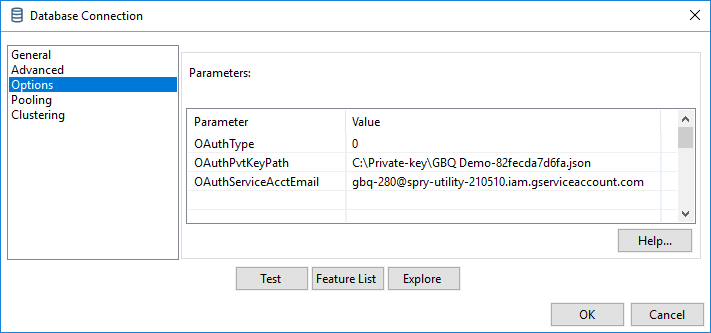
Step 5: Pentaho Data Source Configuration

The Google BigQuery can now be defined in any Pentaho database connection.



|  |  |
| --- | --- |
| SETTING | VALUE |
| Connection Name | GBQ\_Demo |
| Hostname | https://www.googleapis.com/bigquery/v2 |
| Project ID | spry-utility-210510 |
| Port | 443 |
| PARAMETER | VALUE |
| OAuthType | 0 |
| OAuthServiceAcctEmail | gbq-280@spry-utility-210510.iam.gserviceaccount.com |
| OAuthPvtKeyPath | C:\Private-key\GBQ Demo-82fecda7d6fa.json |

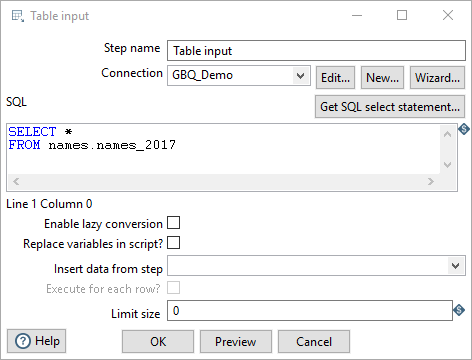
The parameter values are set under ‘Options’



BigQuery and Pentaho Data Integration

To onboard the Google BigQuery dataset:

1. Create a New Transformation
2. Configure the Database Connection as outlined above.
3. Drag and drop a Table Input step onto the Spoon canvas.
4. Double-click to configure step properties.



Preview the data

To Preview the data:

1. Click on the Preview button.

