

# Google BigQuery

📅 Updated on 25 Jun 2025 • ⌚ 5 Minutes to read

## Overview

In this article we will discuss how to connect a Google BigQuery database to Preset. There are two methods that we support to do this:

- Connect using a Service Account; or
- Connect via an OAuth2 integration.

## Allowlist Preset IPs

Access to your BigQuery instance might be limited via VPC Service Controls with IP restrictions (<https://cloud.google.com/bigquery/docs/vpc-sc>). In this case, you need to allow Preset IPs to interact with BigQuery for a successful connection.

Preset Cloud runs on four regions. Below IPs should be allowed (according to the workspace region):

us-west-2 (us1a)	us-east-1 (us2a)	eu-north-1 (eu5a)	ap-northeast-1 (ap1a)
35.161.45.11	44.193.153.196	13.48.95.3	35.74.159.67
54.244.23.85	52.70.123.52	13.51.212.165	35.75.171.157
52.32.136.34	54.83.88.93	16.170.49.24	52.193.196.211

If you are not sure where your Preset workspace is located, you can refer to the URL on your browser when accessing Preset. It should look like this:  
`https://xxxxxxx.us2a.app.preset.io/superset...` , where `us2a` means it is in `us-east-1` .

## Connect using a Service Account

It's possible to use a Service Account to connect BigQuery to Preset. Let's go through this process:

## Create & Configure a Google Service Account

[What is a Google Service Account?](#)

A service account is a special type of Google account intended to represent a non-human user that needs to authenticate and be authorized to access data in Google APIs.

Access **Google Cloud Platform Console** (<https://console.cloud.google.com/projectselector2/iam-admin/serviceaccounts?pli=1&supportedpurview=project>), and select the Project that contains the BigQuery instance you're looking to connect to Preset. Copy the **Project ID**, as you will need this information to set up the connection in Preset.

You'll be redirected to the **Service Accounts** menu.

1. Click on the **+ CREATE SERVICE ACCOUNT** button.

Service accounts

+ CREATE SERVICE ACCOUNT

DELETE

MANAGE ACCESS

REFRESH

Service accounts for project "BigQuery OAuth Docs"

A service account represents a Google Cloud service identity, such as code running on Compute Engine VMs, App Engine apps, or systems running outside Google.

Organization policies can be used to secure service accounts and block risky service account features, such as automatic IAM Grants, key creation/upload, or the cr

Filter

Enter property name or value

<input type="checkbox"/>	Email	Status	Name <span>↑</span>	Description	Key ID	Key creation date	OAuth 2 Client ID <span>?</span>	Actions
No rows to display								

2. Provide a name for the service account, and optionally a description.

1

Service account details

Service account name

Preset Cloud Connection

Display name for this service account

Service account ID \*

preset-cloud-connection

Email address: preset-cloud-connection@bigquery-oauth-docs.iam.gserviceaccount.com

Service account description

Service account used to connect BQ to Preset Cloud.

Describe what this service account will do

CREATE AND CONTINUE

3. Click on **CREATE AND CONTINUE**.
4. In the **Select a role** dropdown, grant the desired permissions to the service account, according to your

needs:

- If you plan to use Preset for read-only access, granting **BigQuery Data Viewer**, **BigQuery Metadata Viewer**, **BigQuery Read Session User** and **BigQuery Job User** roles should be sufficient.
- If you also want to execute DML queries in Preset, grant the **BigQuery Data Editor** role as well.

2

## Grant this service account access to project (optional)

Grant this service account access to BigQuery OAuth Docs so that it has permission to complete specific actions on the resources in your project. [Learn more](#)

Role

BigQuery Data Viewer

Access to view datasets and all of their contents

IAM condition (optional) ?

[+ ADD IAM CONDITION](#)

Role

BigQuery Metadata Viewer

Access to view table and dataset metadata

IAM condition (optional) ?

[+ ADD IAM CONDITION](#)

Role

BigQuery Read Session User

Access to create and use read sessions

IAM condition (optional) ?

[+ ADD IAM CONDITION](#)

Role

BigQuery Job User

Access to run jobs

IAM condition (optional) ?

[+ ADD IAM CONDITION](#)

Role

BigQuery Data Editor

Access to edit all the contents of datasets

IAM condition (optional) ?

[+ ADD IAM CONDITION](#)[+ ADD ANOTHER ROLE](#)[CONTINUE](#)

3

## Grant users access to this service account (optional)

[DONE](#)[CANCEL](#)

5. Click on **DONE**.

You'll be redirected to the Service Accounts list.

[SEND FEEDBACK](#)

# Create & Save a JSON Key

Our next step is to create and download a private JSON key. This will be used later on when we configure the database connection in Preset.

- 1. In the list of accounts, click on the service account email.
- 2. Navigate to the **KEYS** tab.
- 3. Click on the **ADD KEY** dropdown and select **Create new key**.

← Preset Cloud Connection

DETAILS

PERMISSIONS

KEYS

METRICS

LOGS

Keys

⚠

Service account keys could pose a security risk if compromised. We recommend you [review your keys on Google Cloud](#).

ℹ

Google automatically disables service account keys detected in public repositories.

Add a new key pair or upload a public key certificate from an existing key pair.

Block service account key creation using [organization policies](#).  
[Learn more about setting organization policies for service accounts](#)

ADD KEY ▾

Create new key

Upload existing key

Creation date

Expiration date

- 4. The **Create private key** dialog appears. By default, the JSON key type is selected. Click on **CREATE** to download the file to your computer.

Alright! It's time to connect your BigQuery database to Preset.

## Setting up the connection In Preset

Let's start by selecting **+ Database** — have a look at [Connecting your Data](#) if you need help with this step.


Select **Google BigQuery**.

## Connect a database


×

STEP 1 OF 3


### Select a database to connect




Google BigQuery




PostgreSQL




Snowflake



MySQL



Databricks




Google Sheets

### Or choose from a list of other databases we support:

Supported databases

Choose a database... ▾



**Don't see your database listed?**

Any databases that allow connections via SQL Alchemy URIs can be added. [Contact support](#) to have your database added to this list.

There are two ways to provide the Service Account credentials:

## Method One: Using the dynamic form

1. In the **Display Name** field, enter a nickname for your database for display in the Preset environment.

2. Provide the **Project ID** previously copied from Google:

**Connect a database**

STEP 2 OF 3

**Enter the required Google BigQuery credentials**

Need help? Learn more about [connecting to Google BigQuery](#).

**IP Allowlist**

For security reasons you may need to allow Preset's Public IPs: 35.74.159.67 / 35.75.171.157 / 52.193.196.211

Connection method

Username and Password

Display Name \*

Google BigQuery

Pick a nickname for how the database will display in Superset.

Project Id \*

preset-connections-317005

Enter the unique project id for your database.

How do you want to enter service account credentials?

Upload JSON file

[Upload credentials](#)

Additional Parameters

e.g. param1=value1&param2=value2

Back Connect

3. In the **How do you want to enter service account credentials?** field, select:

- Upload JSON file** and then click on **Choose File** to upload the JSON key downloaded earlier; or
- Copy and Paste JSON credentials** and then paste the JSON key content in the **Service Account** text field.

4. (optional) Additional connection parameters can be provided in the **Additional Parameters** field.

5. Click on **Connect**.

Great work! You can now make additional configurations to your database, if needed. To learn how to do this, please visit the [Advanced Connection Settings](#) article. When ready, click on **Finish** [SEND FEEDBACK](#)

## Connect a database

×

STEP 3 OF 3

### Database connected

Create a dataset to begin visualizing your data as a chart or go to SQL Lab to query your data.

[CREATE DATASET](#)[QUERY DATA IN SQL LAB](#)

#### SQL Lab

Adjust how this database will interact with SQL Lab.

▼

#### Performance


Adjust performance settings of this database.

▼

#### Security

Add extra connection information.

▼

 **dbt Cloud**

Sync dbt Cloud models as datasets for this database.

▼

#### Other

Additional settings.

▼

[Back](#)[Finish](#)

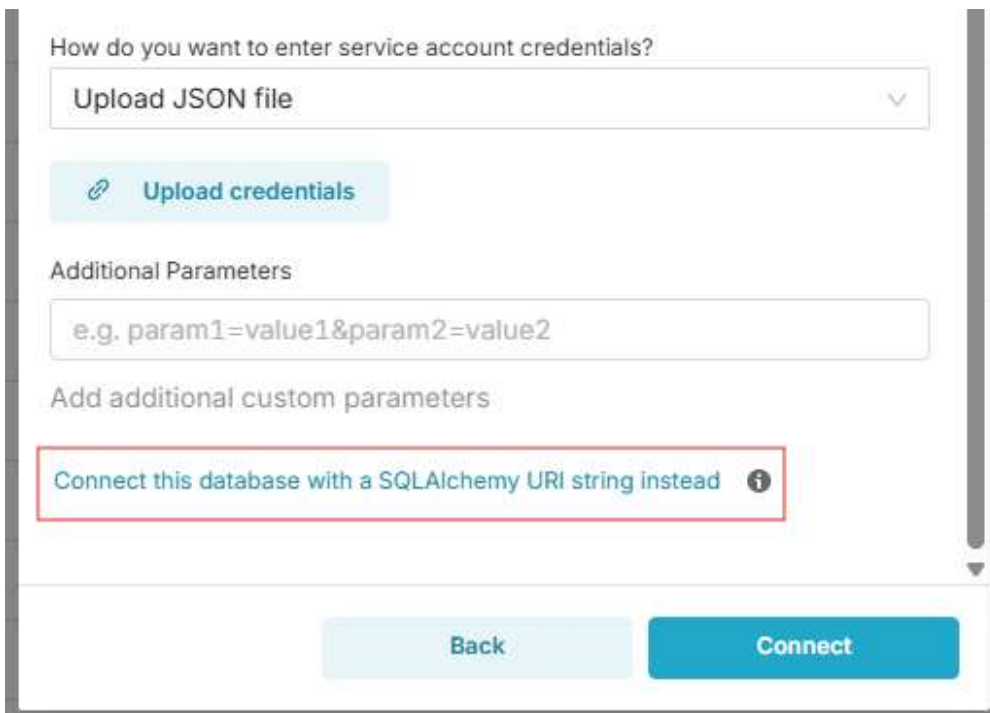
## Method Two: Connect via a SQLAlchemy URI

1. Like the previous method, start by selecting the **Google BigQuery** icon, then scroll to the bottom of the Connect a Database panel.

[SEND FEEDBACK](#)



2. Click on **Connect this database with a SQLAlchemy URI string instead.**



3. The dialog is now organized into two tabs: **Basic** and **Advanced**.

4. In the SQLAlchemy URI field, enter your URI using the following syntax (replace `${Project_ID}` with the actual Project ID):

Plain text	Copy
<code>bigquery://\${Project_ID}</code>	

5. Navigate to the **Advanced** tab and expand the **Security** section.

6. Paste below content in the **Secure Extra** field:

JSON	Copy
<pre>{   "credentials_info":  \${JSON_KEY} }</pre>	

7. Open the JSON key downloaded earlier with a text editor, and copy its entire content.

8. Replace `${JSON_KEY}` in the **Secure Extra** field with the copied key.

## Security

Add extra connection information.



Secure extra

```
1 {  
2   "credentials_info": {  
3     "type": "service_account",  
4     "project_id": "bigquery-oauth-docs",  
5     "private_key_id": "123456789aaaaaaaaaaaBbbbbbbbbbbbbbb",  
6     "private_key": "XXXXXXXXXX",  
7     "client_email": "preset-cloud-connection@bigquery-oau",  
8     "client_id": "123456789123456789",  
9     "auth_uri": "https://accounts.google.com/o/oauth2/aut",  
10    "token_uri": "https://oauth2.googleapis.com/token",  
11    "auth_provider_x509_cert_url": "https://www.googleapi
```

JSON string containing additional connection configuration. This is used to provide connection information for systems like Hive, Presto and BigQuery which do not conform to the username:password syntax normally used by SQLAlchemy.

Root certificate

9. Other configurations can be applied in the **Advanced** tab. If you want to explore them, refer to [Advanced Connection Settings](#).

10. Navigate back to the **Basic** tab and then select **Test Connection**.

Great work! The connection is now created.

## Connect using an OAuth2 integration

It's possible to connect to BigQuery via OAuth2 to apply individual user permissions from BigQuery to the queries that users run in Preset. This approach requires setting up the integration in the Google Cloud Console.

Let's go through this process.

## Select the Google Project

Access [Google Cloud Platform Console](https://console.cloud.google.com/projectselector2/iam-admin/serviceaccounts?pli=1&supportedpurview=project) (<https://console.cloud.google.com/projectselector2/iam-admin/serviceaccounts?pli=1&supportedpurview=project>), and select the Project that contains the BigQuery instance you're looking to connect to Preset. Copy the **Project ID**, as you will need this information to set up the connection in Preset.

## Setting up API & Services

Navigate to the **API & Services** menu. Then follow these steps:

SEND FEEDBACK

# Creating an OAuth consent screen

- 1. Select **OAuth consent screen** in the sidebar menu.
- 2. Set **User Type** as **Internal** and click on **CREATE**.
- 3. Provide an **App name** and **User support email**.
- 4. Click on **ADD DOMAIN** and add `preset.io` as an **Authorized domain**.
- 5. Provide **Developer email address**.
- 6. Click on **SAVE AND CONTINUE**.

Note that the **User support email** and **Developer email address** can be internal to your Organization.

## Setting up an OAuth Client ID

- 1. Select **Credentials** in the sidebar menu.
- 2. Click on **+ CREATE CREDENTIALS > OAuth Client ID**.
- 3. Set **Web application** in the **Application type** field.
- 4. Provide a **Name** to the client.
- 5. Click on **+ ADD URI** under the **Authorized redirect URIs** section.
- 6. Replace `${WORKSPACE_DOMAIN}` in the link below and use it as the URI:

Plain text	Copy
<code>https://\${WORKSPACE_DOMAIN}/api/v1/database/oauth2/</code>	

## Create OAuth client ID

A client ID is used to identify a single app to Google's OAuth servers. If your app runs on multiple platforms, each will need its own client ID. See [Setting up OAuth 2.0](#) for more information. [Learn more](#) about OAuth client types.

Application type \*

Web application ▼

Name \*

Preset Cloud OAuth Client

The name of your OAuth 2.0 client. This name is only used to identify the client in the console and will not be shown to end users.



The domains of the URIs you add below will be automatically added to your [OAuth consent screen](#) as [authorized domains](#).

### Authorized JavaScript origins

For use with requests from a browser

[+ ADD URI](#)

### Authorized redirect URIs

For use with requests from a web server

URIs 1 \*

https://abc12312.us2a.app.preset.io/api/v1/database/oauth2/

[+ ADD URI](#)

Note: It may take 5 minutes to a few hours for settings to take effect

**CREATE**

CANCEL

7. Click on **CREATE**.

8. In the confirmation dialog, click on **DOWNLOAD JSON** to retrieve the OAuth2 credentials.

It's time to set up the connection in Preset.

## Setting up the connection In Preset

Let's start by selecting **+ Database** — have a look at [Connecting your Data](#) if you need help with this step.

[SEND FEEDBACK](#)


Select **Google BigQuery**.

### Connect a database


×

STEP 1 OF 3


#### Select a database to connect




Google BigQuery




PostgreSQL




Snowflake



MySQL



Databricks




Google Sheets

Or choose from a list of other databases we support:

Supported databases

Choose a database... ▾



**Don't see your database listed?**

Any databases that allow connections via SQL Alchemy URIs can be added. [Contact support](#) to have your database added to this list.

1. In the **Display Name** field, enter a nickname for your database for display in the Preset environment.
2. Provide the **Project ID** previously copied from Google.

3. Change the **Connection method** to **OAuth2**

The screenshot shows a 'Connect a database' dialog box with a close button (X) in the top right corner. It is labeled 'STEP 2 OF 3' and has the title 'Enter the required Google BigQuery credentials'. Below the title is a link: 'Need help? Learn more about [connecting to Google BigQuery](#).' There is an information box titled 'IP Allowlist' with an 'i' icon, containing the text: 'For security reasons you may need to allow Preset's Public IPs: 35.74.159.67 / 35.75.171.157 / 52.193.196.211'. Below this is a 'Connection method' dropdown menu. The dropdown is open, showing three options: 'OAuth2' (highlighted with a red border and a checkmark), 'Username and Password', and 'OAuth2'. Below the dropdown is a text input field for a nickname, with the placeholder text 'Pick a nickname for how the database will display in Superset.' Below that is a 'Project Id' field with a red asterisk, containing the text 'preset-connections-317005'. Below the Project Id field is a text input field for 'Client ID' with a red asterisk, containing the placeholder text 'e.g. your-client-id'. Below the Client ID field is a 'Client Secret' field with a red asterisk, containing the placeholder text 'e.g. your-client-secret' and an eye icon. At the bottom are two buttons: 'Back' and 'Connect'.

4. Open the OAuth2 credentials JSON file downloaded earlier in a text editor to fill the OAuth2 form. Fill the:

- Client ID** field with the "client\_id" value.
- Client Secret** field with the "client\_secret" value.
- Authorization Request URI** field with the "auth\_uri" value.
- Token Request URI** field with the "token\_uri" value.
- Scope** field with `https://www.googleapis.com/auth/bigquery`.

5. Click on **Connect**.

6. You'll be redirected to the **Advanced** tab.
7. Expand the **Security** section and enable the **Impersonate logged in user** checkbox.
8. Other configurations can be applied in the **Advanced** tab. If you want to explore them, refer to **Advanced Connection Settings**.
9. Click **Finish** to save your database connection.

To test the integration:

1. Navigate to **SQL > SQL Lab**.
2. Select the BigQuery connection in the **Database** dropdown.
3. You should see an **Authorization needed** error. Click on **See more**.
4. Click on the provide authorization hyperlink to be redirected to Google. Authenticate to your Google account and grant consent for the integration.
5. After authentication, you're redirected back to Preset. Refresh the page.

Queries should now be properly authenticated with your OAuth2 token. Note that this **Authorization needed** prompt will be displayed to all users the first time they interact with this database (either in SQL Lab or via charts).

←	Previous Databricks	Next Google Sheets (Private)	→
---	------------------------	---------------------------------	---