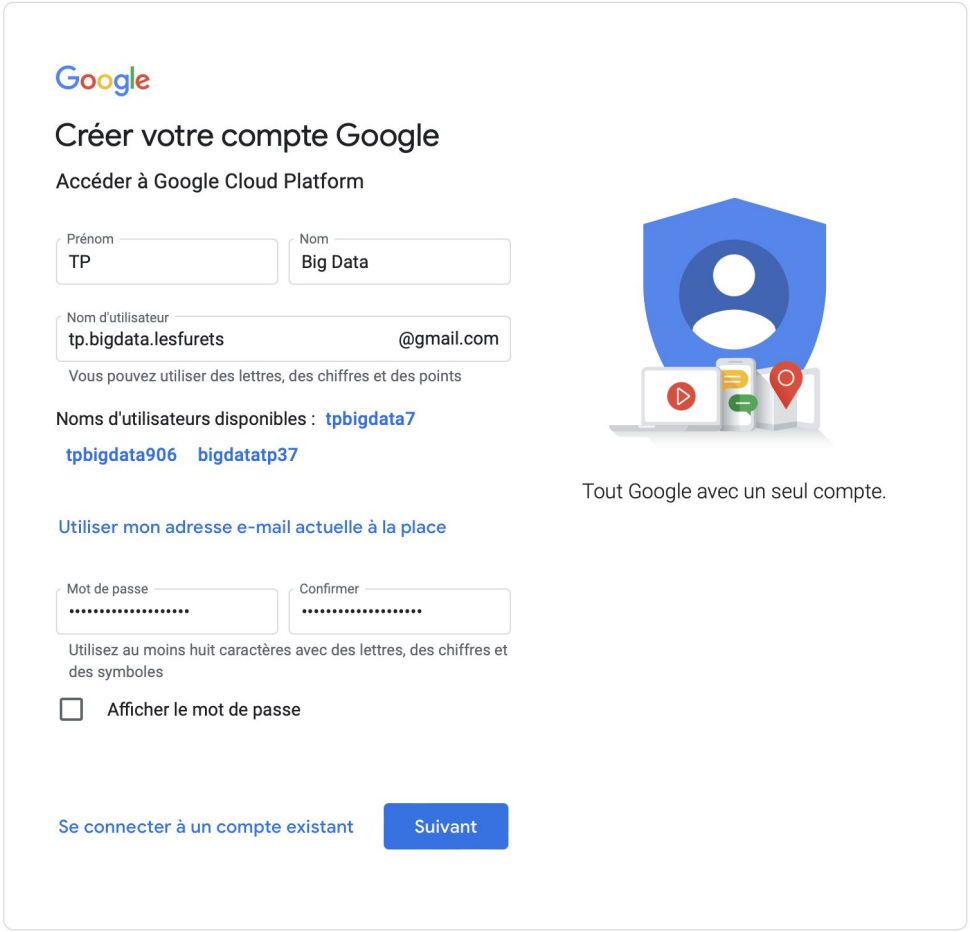
TP DBT & BigQuery

Chafik NOUIRA, LesFurets.com
tp-bigdata@lesfurets.com

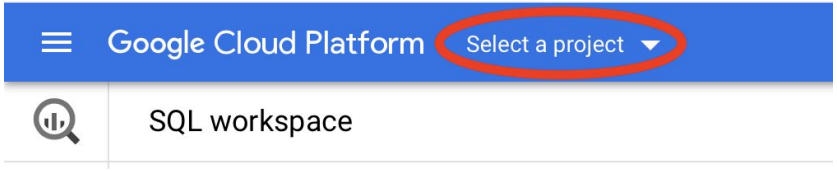
1 - Create a GCP account using BigQuery Sandbox Web UI https://console.cloud.google.com/bigquery

No credit card or phone number are needed



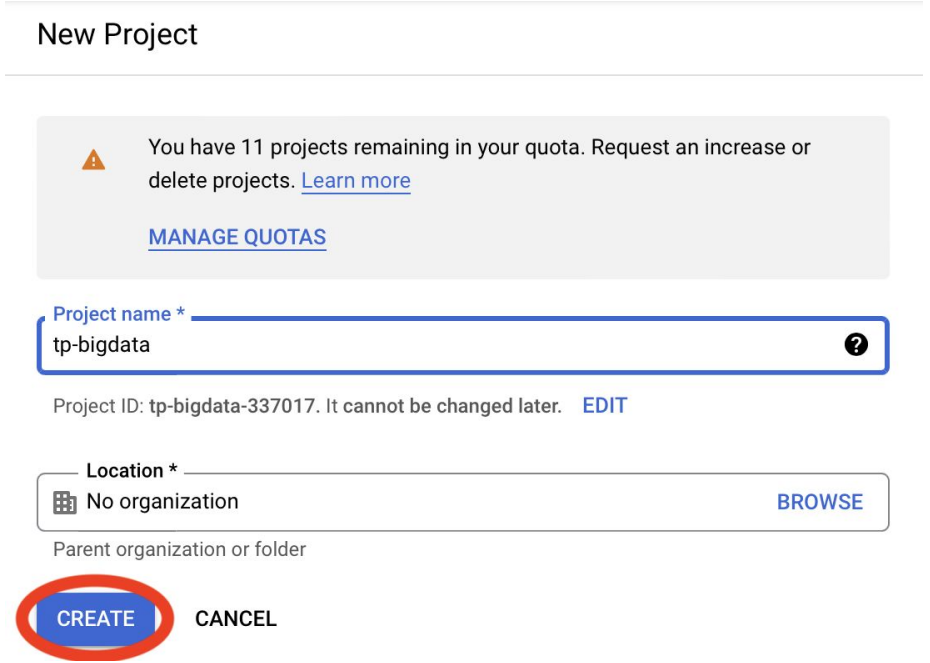
2 - Create a new project and test the query in *customers_original.sql*

1



2

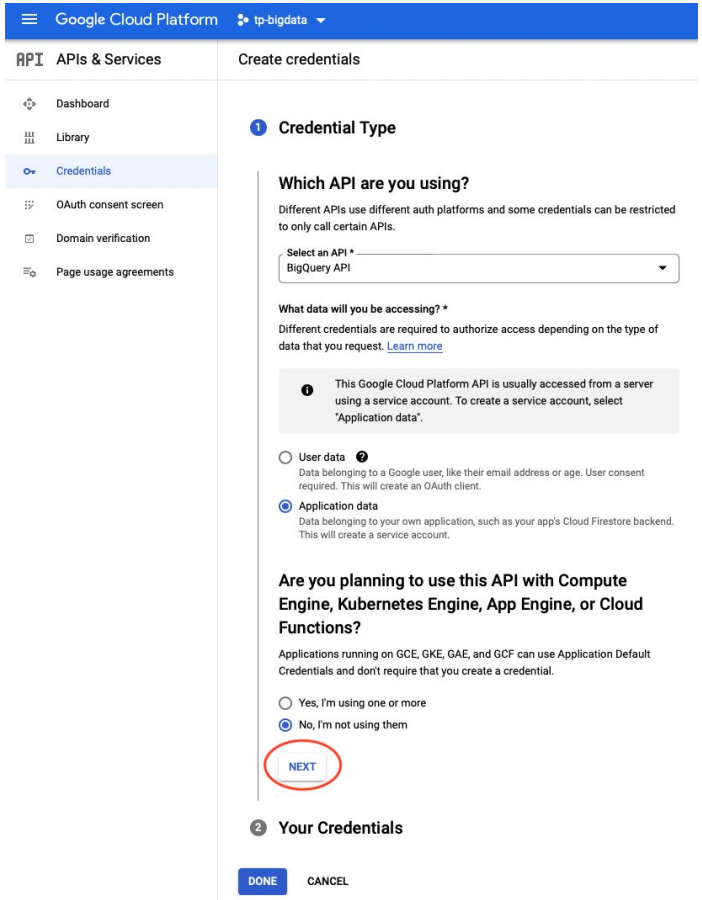
3



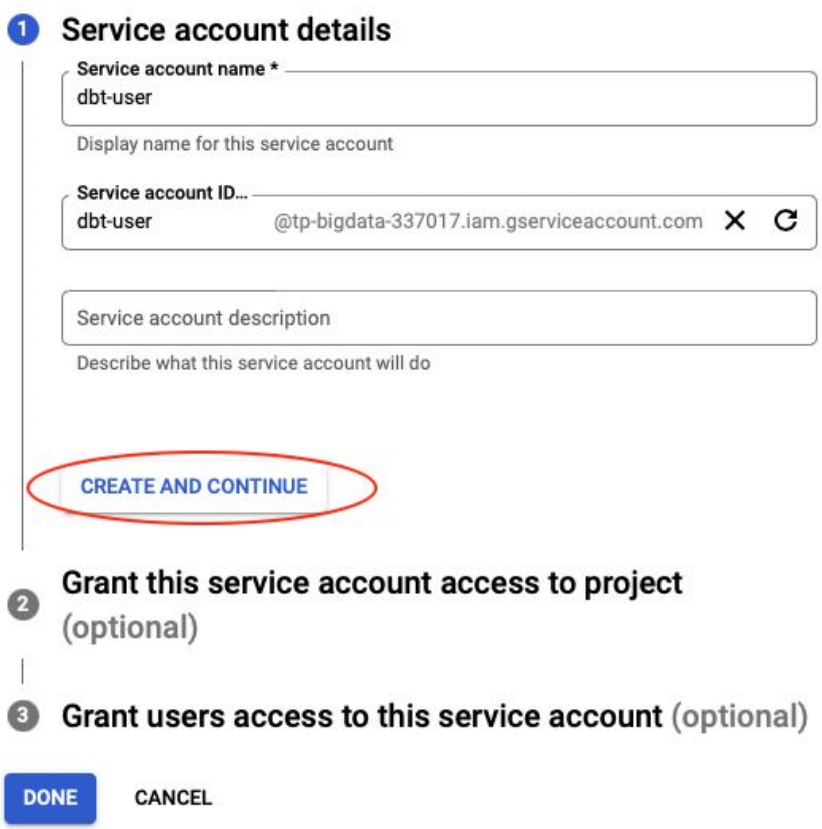
3 - Create a Service Account for dbt-user

BigQuery Credentials page : https://console.cloud.google.com/apis/credentials/wizard

1

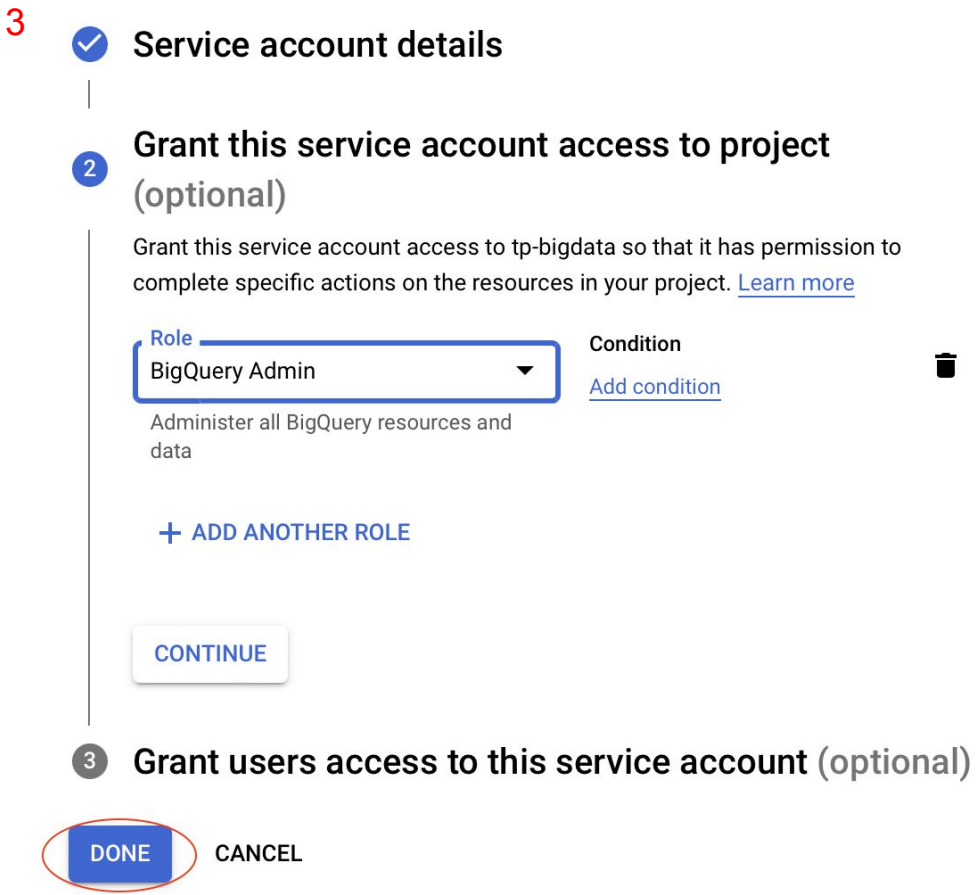


2



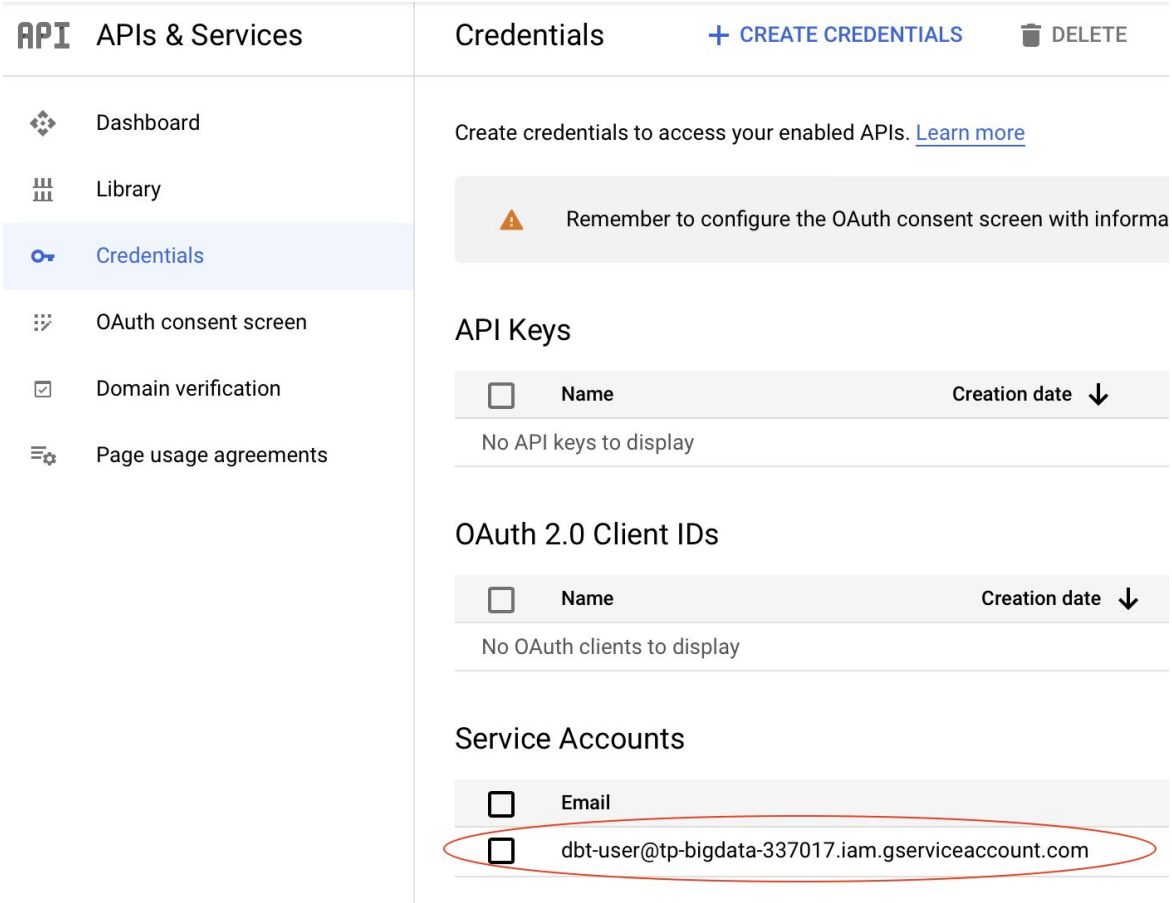
3 - Create a Service Account for dbt-user

BigQuery Credentials page : https://console.cloud.google.com/apis/credentials/wizard



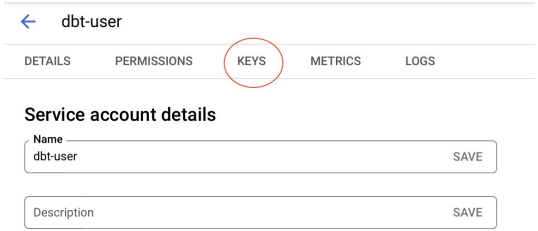
4 - Generate and download BigQuery API Key (JSON file). We'll use it later to connect ***dbt*** to ***BigQUery***

1

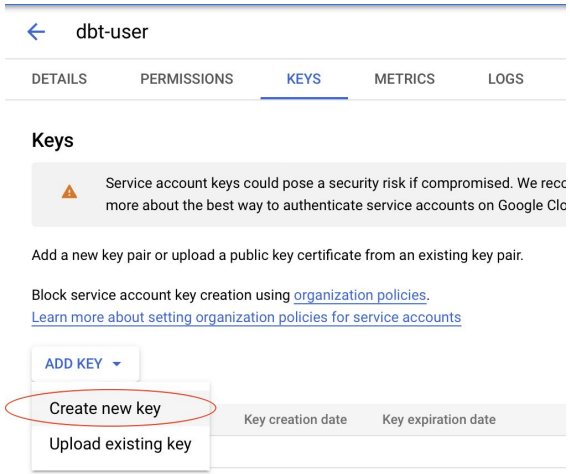


4 - Generate and download BigQuery API Key (JSON file). We'll use it later to connect ***dbt*** to ***BigQUery***

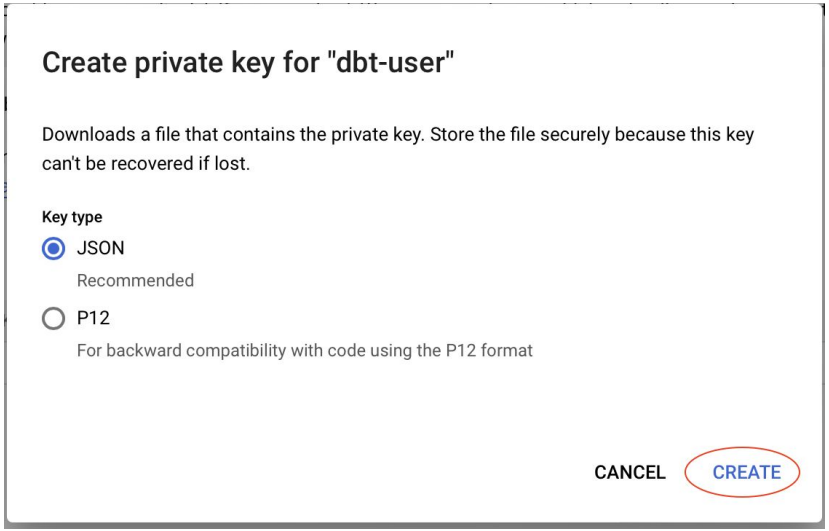
2



3



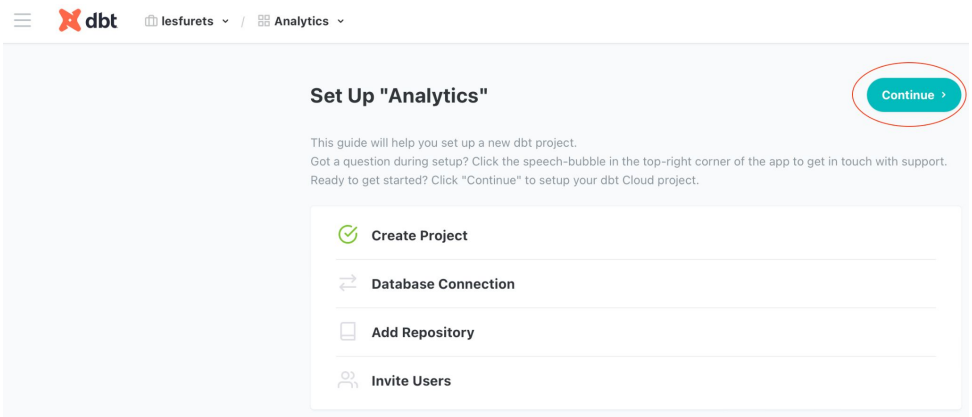
4 - the json file is downloaded automatically



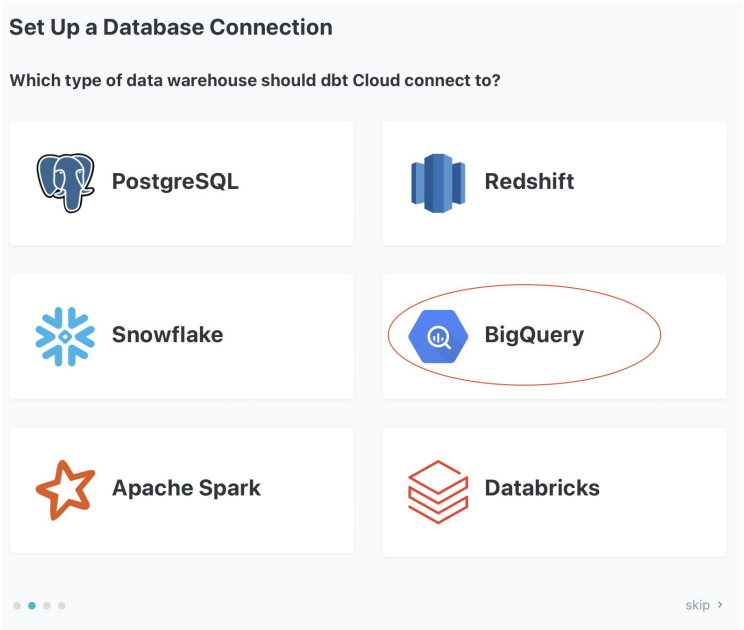
5 - Create a dbt account : https://cloud.getdbt.com/signup/

6 - Configure the new project using the JSON file generated in step 4

1

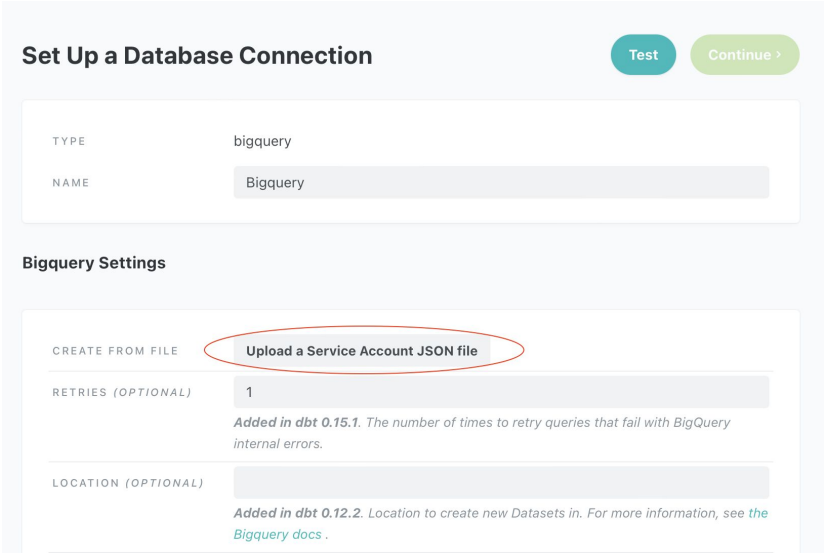


2

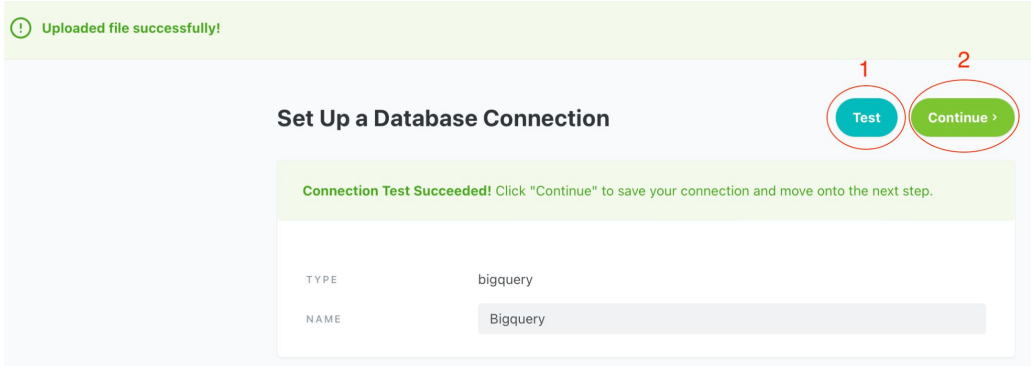


6 - Configure the new project using the JSON file generated in step 4

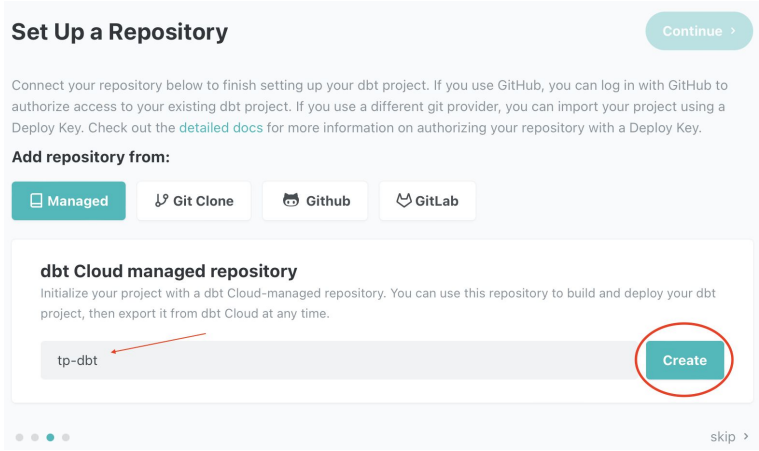
3



4

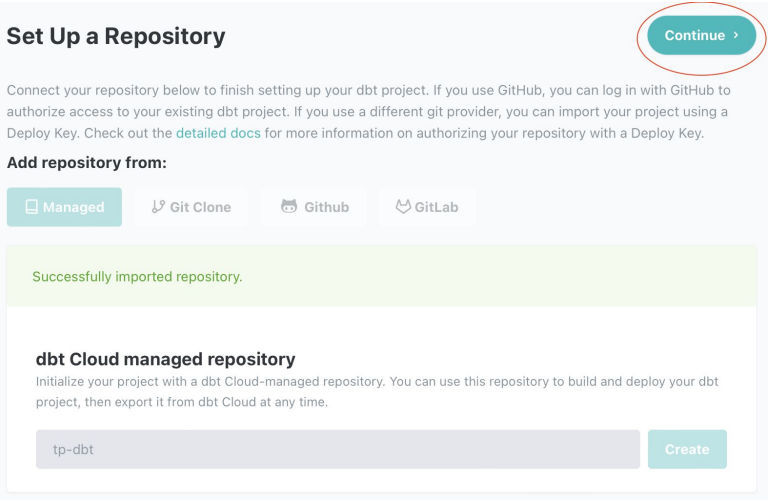


5

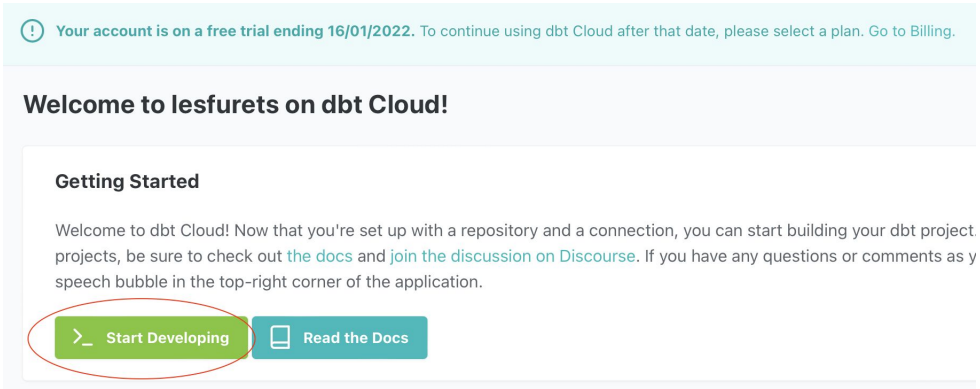


6 - Configure the new project using the JSON file generated in step 4

6



7



7 - Initialize your project

Note : in the next steps, you should always hit Save (CTRL + S) to save your files before executing dbt commands

Note 2 : for every change in the model, files must be saved and dbt run executed to rebuild models before testing or generating docs

8 - In the ***dbt_project.yml*** file, replace ***my_new_project*** by ***tp_dbt*** (Lines 5 and 35)

8.1 - Go to line 38 : ***materialized : view***, what does it mean ?

8.2 - Go to the file ***models/example/my_first_dbt_model.sql***, line 10, what does it mean ?

8.3 - Explain the relationship between the two models.

8.4 - Explain ***models/example/schema.yml*** file.

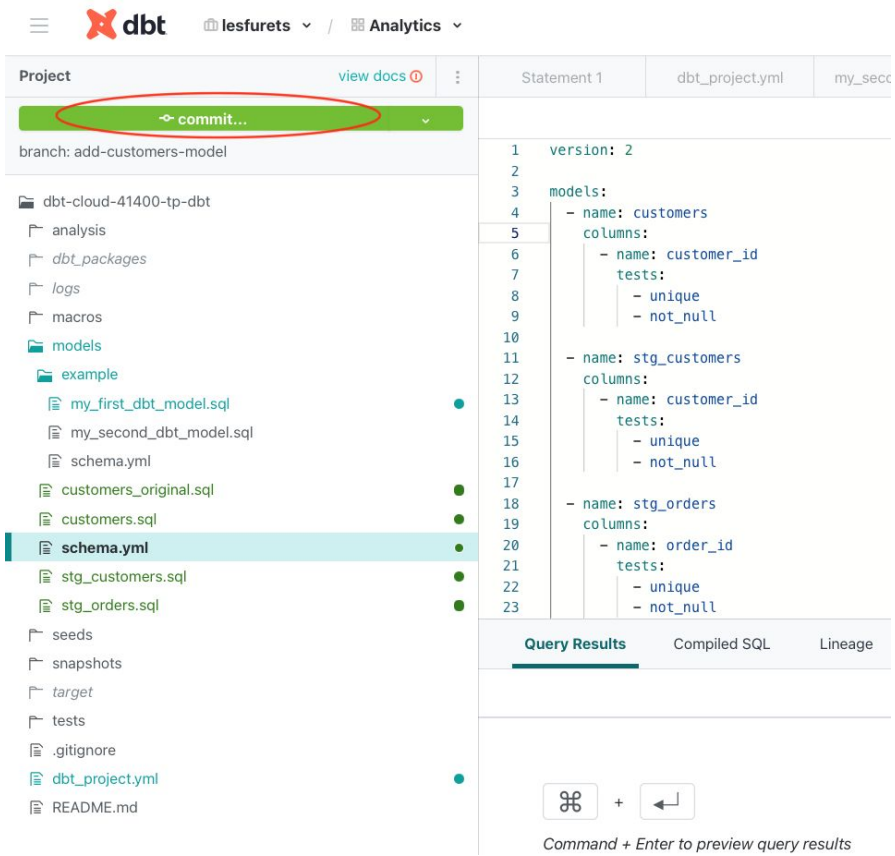
8.5 - At the bottom of the screen, write ***dbt run*** then click Enter. The project should compile with no errors.

8.6 - Execute the command ***dbt test*** to run tests. As you can see, there's a test failure for the first model. Fix it.

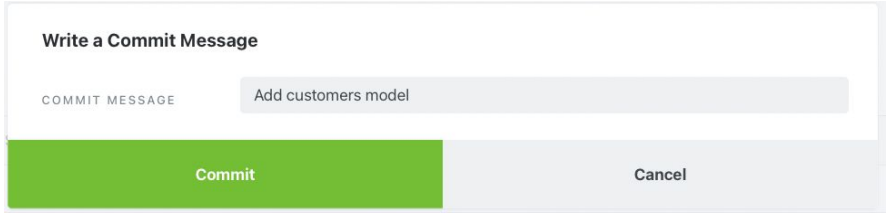
9 - Make your first commit.
10 - The master branch is now “read-only”. Go ahead and create a new branch named ***add-customers-model***.
11 - Create a file named ***customers_original.sql*** in ***models*** folder and copy the query from the provided file with the same name (in the .zip file).
12 - In ***dbt_project.yml*** file, make sure that models in “***models***” folder are materialized as tables (do not apply this change to “***example***” folder)
13 - Execute ***dbt run*** and check the new table created in BigQuery (you’ll need to refresh the page)
14 - Duplicate ***models/customers_original.sql*** in a file named ***customers.sql*** (same folder)
15 - Separate ***customer.sql*** model in 3 different models : ***stg_customers***, ***stg_orders*** and ***customers***. <
Make sure to add ***stg_customers*** and ***stg_orders*** references in the ***customers*** model.
(see ***models/example/my_second_dbt_model.sql***)
16 - Run “***dbt run***” and then check BigQuery tables.
17 - Add tests and documentation to ***stg_customers***, ***stg_orders*** and ***customers*** models.
18 - Run “ ***dbt test*** ”

19 - Commit your changes and then merge to master

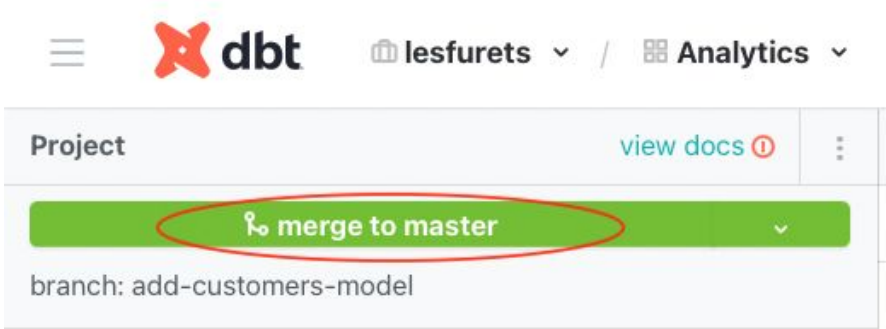
1



2

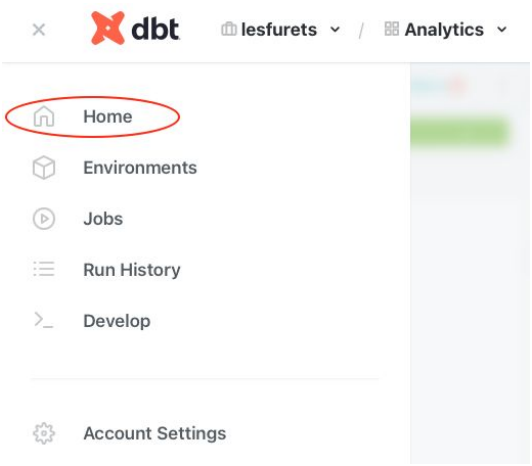


3

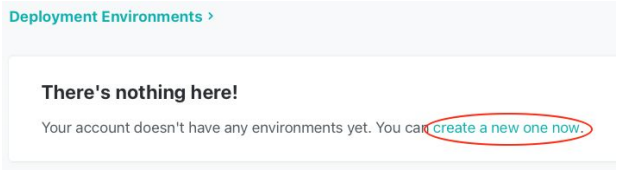


20 - Deploy your project in production

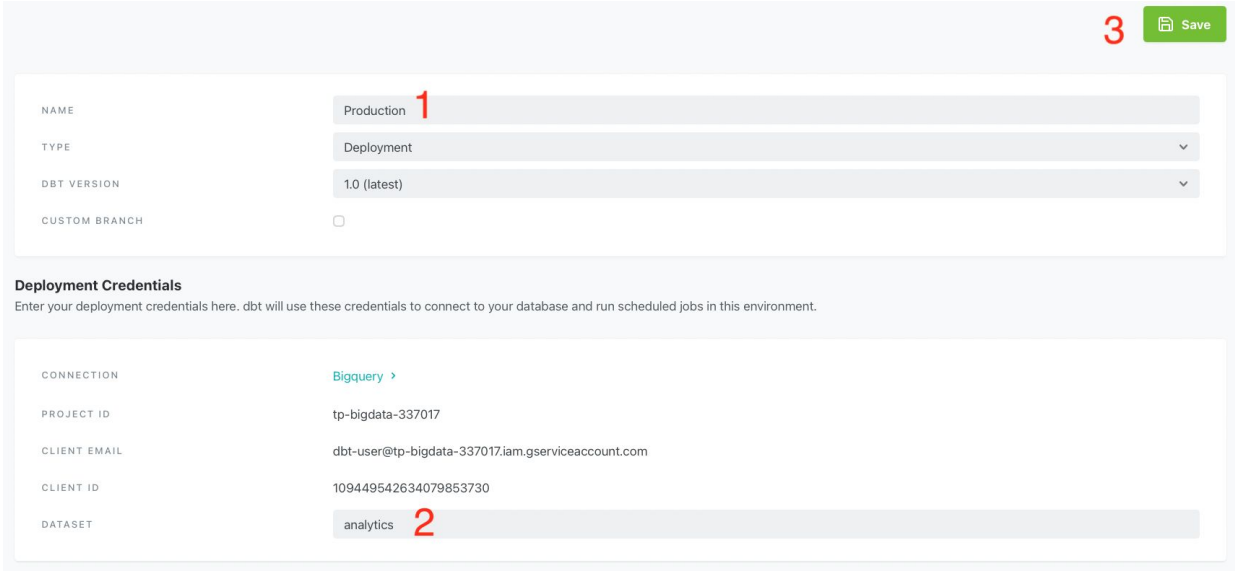
1



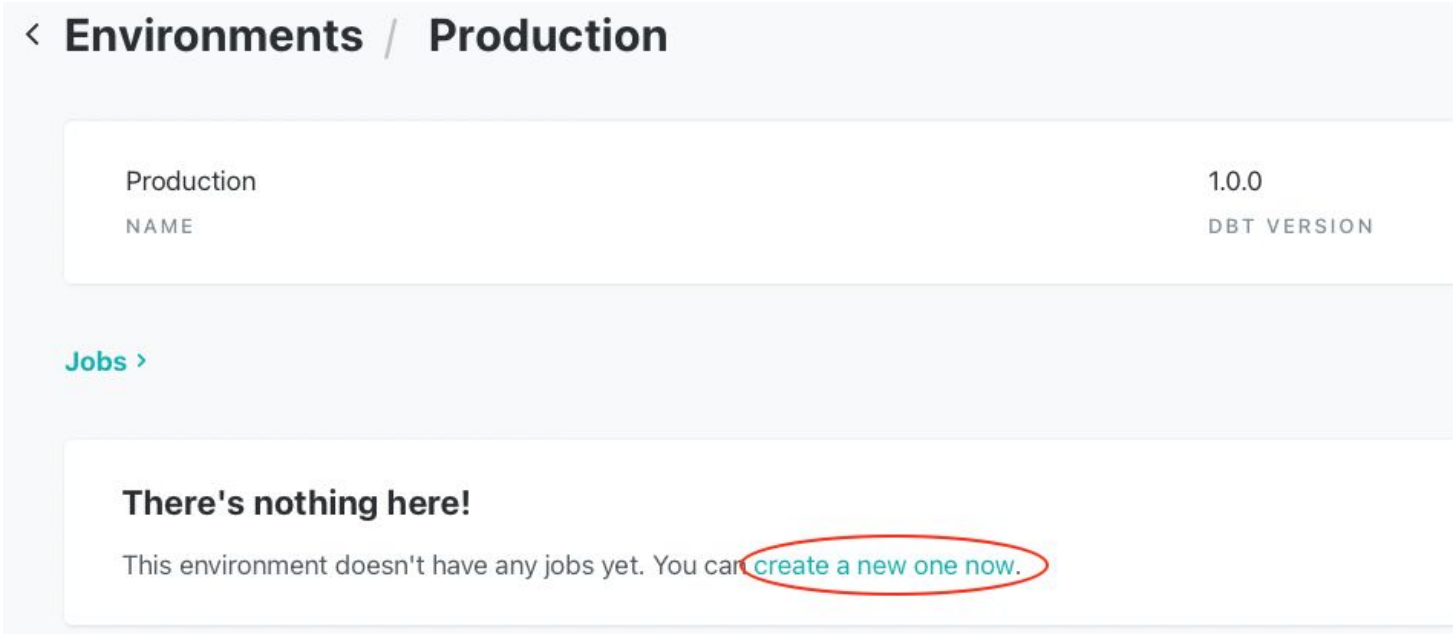
2



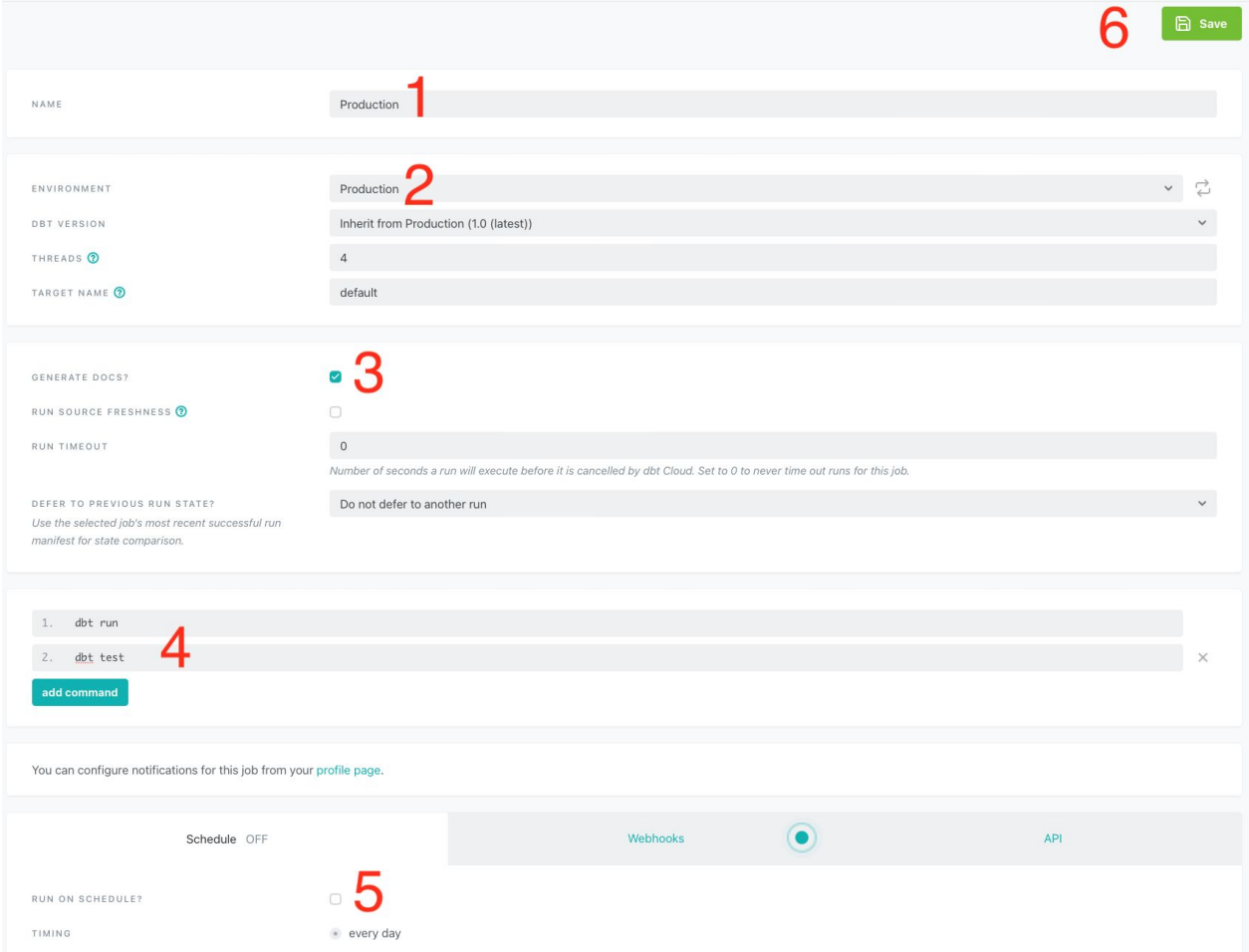
3



21 - Create a job to run you models in production



21 - Create a job to run you models in production



22 - Run your job and check BigQuery tables