Important Commands

**To set up GOOGLE\_APPLICATION\_CREDENTIALS**

*Linux - export GOOGLE\_APPLICATION\_CREDENTIALS=/home/shrutijoshi/helpful-cat-324414-bbf5a431efb0.json*

*Windows - set GOOGLE\_APPLICATION\_CREDENTIALS=C:\Documents\Learning\SVC\fabric-1333-29edd96ad1f1.json*

**To Configure service account with google cloud SDK**

*gcloud auth activate-service-account svc-rw@fabric-1333.iam.gserviceaccount.com --key-file=C:\Documents\Learning\SVC\fabric-1333-29edd96ad1f1.json --project=fabric-1333*

gcloud auth activate-service-account svc-dezyre-editor@helpful-cat-324414.iam.gserviceaccount.com --key-file=helpful-cat-324414-bbf5a431efb0.json --project=helpful-cat-324414

Prerequisites – Install cloud sdk and configure service account.

Install Python and install **google-cloud-bigquery==2.25.1**

**To create new dataset from cli**

bq mk dataset1

**To show dataset**

bq show --format=prettyjson dataset1

**To update dataset properties**

bq update --description "Description of dataset1" dataset1

bq update --default\_table\_expiration 3600 dataset1

**Create table using cli**

bq mk \  
  -t \  
  --expiration 7200 \  
  --description "This is my table" \  
  --label organization:development \  
  dataset1.table1 \  
  qtr:STRING,sales:FLOAT,year:STRING

bq mk \

--table \  
  --expiration 7200 \  
  --description "This is my table" \  
  --label organization:development \  
  dataset1.table2 \  
  schema.json

bq show --format=prettyjson dataset1.table1

**To get schema of table**

bq show --schema --format=prettyjson dataset1.table2

**To get metadata of tables**

bq query --nouse\_legacy\_sql ‘SELECT \* FROM dataset1.INFORMATION\_SCHEMA.TABLES’

The following command loads data gcs bucket into a table named avrotable in dataset1.

    bq load \  
    --source\_format=AVRO \  
    dataset1.avrotable \  
    gs://dezyre-test-bucket1/sample-data/avro/userdata1.avro

The following command loads data from gcs bucket into an ingestion-time partitioned table named avrotable in dataset1.

    bq load \  
    --source\_format=AVRO \  
    --time\_partitioning\_type=DAY \  
    dataset1.avrotable \  
    gs://dezyre-test-bucket1/sample-data/avro/userdata1.avro

The following shows how time based column is used for partitioning.

    bq load \  
    --source\_format=AVRO \  
    --time\_partitioning\_field timestampcolumnname \  
    dataset1.avrotable \  
    gs://dezyre-test-bucket1/sample-data/avro/data.avro

The following command loads data from multiple files in gcs bucket into a table named avrotable in dataset1. The Cloud Storage URI uses a wildcard.

    bq load \  
    --source\_format=AVRO \  
    dataset1.avrotable \  
    gs://dezyre-test-bucket1/sample-data/avro/userdata\*.avro

The following command loads data from gs://dezyre-test-bucket1/sample-data/avro/userdata2.avro and overwrites a table named avrotable in dataset1.

    bq load \  
    --replace \  
    --source\_format=AVRO \  
    dataset1.avrotable \  
    gs://dezyre-test-bucket1/sample-data/avro/userdata2.avro

The following command loads data from gs://dezyre-test-bucket1/sample-data/avro/userdata2.avro and appends data to a table named avrotable in dataset1.

    bq load \  
    --noreplace \  
    --source\_format=AVRO \  
    dataset1.avrotable \  
    gs://dezyre-test-bucket1/sample-data/avro/userdata2.avro

**Similarly we can use parquet and orc files. Below is one sample query for both.**

bq load \  
    --source\_format=ORC \  
    dataset1.orctable \  
    gs://dezyre-test-bucket1/sample-data/orc/userdata1\_orc

bq load \  
    --source\_format=PARQUET \  
    dataset1.parquettable \  
    gs://dezyre-test-bucket1/sample-data/parquet/userdata1.parquet

Create External table on GCS bucket

bq mkdef \  
--autodetect \  
--source\_format=CSV \  
gs://dezyre-test-bucket1/data/flights.csv > def\_file

For more info on external definition file refer to <https://cloud.google.com/bigquery/external-table-definition>

bq mk --external\_table\_definition=def\_file dataset1.flightexttable

**Python command to load csv data in managed table.**

**pip install google-cloud-bigquery==2.25.1**

python3 csvLoad.py