

Dataproc - Spark ML

- Dataproc은 일괄 처리, 쿼리, 스트리밍, 머신 러닝에 오픈소스 데이터 도구를 활용할 수 있는 관리형 Spark 및 Hadoop 서비스입니다. Dataproc 자동화를 통해 신속하게 클러스터를 만들고 손쉽게 관리하며 불필요한 클러스터를 사용 중지하여 비용을 절감할 수 있습니다. 관리 시간과 비용이 절감되므로 작업과 데이터에 집중할 수 있습니다.

- Created a fully-managed Cloud SQL instance for rentals
- Created tables and explored the schema with SQL
- Ingested data from CSVs
- Edited and ran a Spark ML job on Dataproc
- Viewed prediction results

1. Create a Cloud SQL instance

1. In the Google Cloud Console, Select **Navigation menu > SQL** (in the Databases section).
2. Click **Create instance**.
3. Click **Choose MySQL**.
4. For **Instance ID**, type **rentals**.

Instance ID
ID is permanent. Use lowercase letters and numbers only.

2. Create tables

```
CREATE DATABASE IF NOT EXISTS recommendation_spark;

USE recommendation_spark;

DROP TABLE IF EXISTS Recommendation;
DROP TABLE IF EXISTS Rating;
DROP TABLE IF EXISTS Accommodation;

CREATE TABLE IF NOT EXISTS Accommodation
(
  id varchar(255),
  title varchar(255),
  location varchar(255),
  price int,
  rooms int,
  rating float,
  type varchar(255),
  PRIMARY KEY (ID)
);

CREATE TABLE IF NOT EXISTS Rating
(
  userId varchar(255),
  accoId varchar(255),
  rating int,
  PRIMARY KEY(accoId, userId),
  FOREIGN KEY (accoId)
    REFERENCES Accommodation(id)
);

CREATE TABLE IF NOT EXISTS Recommendation
(
  userId varchar(255),
  accoId varchar(255),
```

```
prediction float,
PRIMARY KEY(userId, accoId),
FOREIGN KEY (accoId)
REFERENCES Accommodation(id)
);

SHOW DATABASES;
```

3. Stage Data

- Option 1: Use the command line

```
echo "Creating bucket: gs://$DEVSHHELL_PROJECT_ID"
gsutil mb gs://$DEVSHHELL_PROJECT_ID

echo "Copying data to our storage from public dataset"
gsutil cp gs://cloud-training/bdml/v2.0/data/accommodation.csv gs://$DEVSHHELL_PROJECT_ID
gsutil cp gs://cloud-training/bdml/v2.0/data/rating.csv gs://$DEVSHHELL_PROJECT_ID

echo "Show the files in our bucket"
gsutil ls gs://$DEVSHHELL_PROJECT_ID

echo "View some sample data"
gsutil cat gs://$DEVSHHELL_PROJECT_ID/accommodation.csv
```

- Option 2: Use the Cloud Console UI

- Navigate to **Storage** and select **Cloud Storage > Browser**.
- Click **Create Bucket** (if one does not already exist).
- Specify your project name as the bucket name.
- Click **Create**.
- Download the below files locally and then upload them inside of your new bucket

4. Load data from Cloud Storage into Cloud SQL tables

- 콘솔의 SQL로 이동해서 import를 통해서 accommodation, ratings load

Google Cloud Platform | qwiklabs-gcp-01-1d19b3f6a0e4 | Search products and services

SQL

MASTER INSTANCE

- Overview
- Connections
- Users
- Databases
- Backups
- Replicas
- Operations

Import data from Cloud Storage

Source

Choose the file you'd like to import data from
Browse for a file, or enter the path for one (bucket/folder/file). Make sure you have read access first. [Learn more](#)

☒ qwiklabs-gcp-01-1d19b3f6a0e4/accommodation.csv [Browse](#)

Indicate the format of the file you're importing

☐ SQL
A plain text file with a sequence of SQL commands, like the output of mysqldump

☒ CSV
If your Cloud Storage file is a CSV file, select CSV. The CSV file should be a plain text file with one line per row and comma-separated fields.

Destination

Choose the database and table in your Cloud SQL instance that you'd like to import your file into

Database

Table

[Import](#)

When you import, a Cloud SQL service account will be granted read access to your Cloud Storage file and the bucket that contains it. This will be reflected in your permissions.

5. Explore Cloud SQL data

6. Launch Dataproc

1. SQL에서 Dataproc API 허용
2. cluster 생성
3. Master/Workers 노트 생성

```
echo "Authorizing Cloud Dataproc to connect with Cloud SQL"
CLUSTER=rentals
CLOUDSQL=rentals
ZONE=us-central1-c
NWORKERS=2

machines="$CLUSTER-m"
for w in `seq 0 $((NWORKERS - 1))`; do
    machines="$machines $CLUSTER-w-$w"
done

echo "Machines to authorize: $machines in $ZONE ... finding their IP addresses"
ips=""
for machine in $machines; do
    IP_ADDRESS=$(gcloud compute instances describe $machine --zone=$ZONE --format='value(networkInterfaces.accessConfigs[].natIP)' | sed "s/\"/\\\"/g")
    echo "IP address of $machine is $IP_ADDRESS"
    if [ -z $ips ]; then
        ips=$IP_ADDRESS
    else
        ips="$ips,$IP_ADDRESS"
    fi
done

echo "Authorizing [$ips] to access cloudsql=$CLOUDSQL"
gcloud sql instances patch $CLOUDSQL --authorized-networks $ips
```

4. SQL Public IP copy

7. Run the ML model

```
gsutil cp gs://cloud-training/bdml/v2.0/model/train_and_apply.py train_and_apply.py
cloudshell edit train_and_apply.py
```

8. Run your ML job on Dataproc

1. Submit Job

Cluster
cluster-1

Job type
PySpark

Main python file
gs://cloud-training/bdml/v2.0/model/train_and_apply.py

Your bucket name here

9. Explore inserted rows with SQL