

DATA CENTER SCALE COMPUTING - LAB 3

Objective - This lab is designed to have you perform and run queries on Hive by enabling a dataproc cluster. The outcome of this assignment will be

- Learn to enable and run dataproc on gcp
- Learn to create tables, update and query on Hive
- Learn to perform Data Joins.

Please find the tutorial to set up Hive at

- <https://cloud.google.com/architecture/using-apache-hive-on-cloud-dataproc#creating-an-other-cloud-dataproc-cluster>

Instructions:

1. You should be setting up Hive and answering the questions that follow.
2. Create a self contained document that has your solutions and the screenshot of the query output(for each query)
3. There will be 5 query executions and two theoretical questions.

The tutorial provides a sample dataset and you may play around with the same. However you should be performing the queries on the dataset that I have shared.

Please find the steps to create a table and query the same.

Copying the datasets (15 points)

In this section, you upload the datasets to your warehouse bucket, create a new Hive table, and run some HiveQL queries on that dataset.

Step 1:

Copy the datasets provided to your warehouse bucket: Run the following command in your cloud shell.

Dataset1 for table “**personal_details**”

```
gsutil cp gs://my-hive-warehouse-ajay/datasets/personal_details/heart01.parquet \  
gs://${WAREHOUSE_BUCKET}/datasets/personal_details/heart01.parquet
```

Dataset2 for table “**health_details**”

```
gsutil cp gs://my-hive-warehouse-ajay/datasets/health_details/heart02.parquet \  
gs://${WAREHOUSE_BUCKET}/datasets/personal_details/heart02.parquet
```

Dataset3 for table “diet”

```
gsutil cp gs://my-hive-warehouse-ajay/datasets/health_details/heart02.parquet \
gs://{WAREHOUSE_BUCKET}/datasets/personal_details/heart03.parquet
```

Step 2: Creating the Hive tables (15 points)

Create an external Hive table for the dataset:

Table “personal_details”

```
gcloud dataproc jobs submit hive \
  --cluster hive-cluster \
  --region ${REGION} \
  --execute "
    CREATE EXTERNAL TABLE personal_details
    (id INT, AGE DOUBLE, SEX STRING, dataset STRING)
    STORED AS PARQUET
    LOCATION 'gs://{WAREHOUSE_BUCKET}/datasets/personal_details';"
```

Table “health_details”

```
gcloud dataproc jobs submit hive \
  --cluster hive-cluster \
  --region ${REGION} \
  --execute "
    CREATE EXTERNAL TABLE health_deatils
    (id INT,cp STRING,trestbps DOUBLE,choi DOUBLE,fbs STRING,restecg STRING, weight
    INT)
    STORED AS PARQUET
    LOCATION 'gs://{WAREHOUSE_BUCKET}/datasets/health_details';"
```

Table “diet”

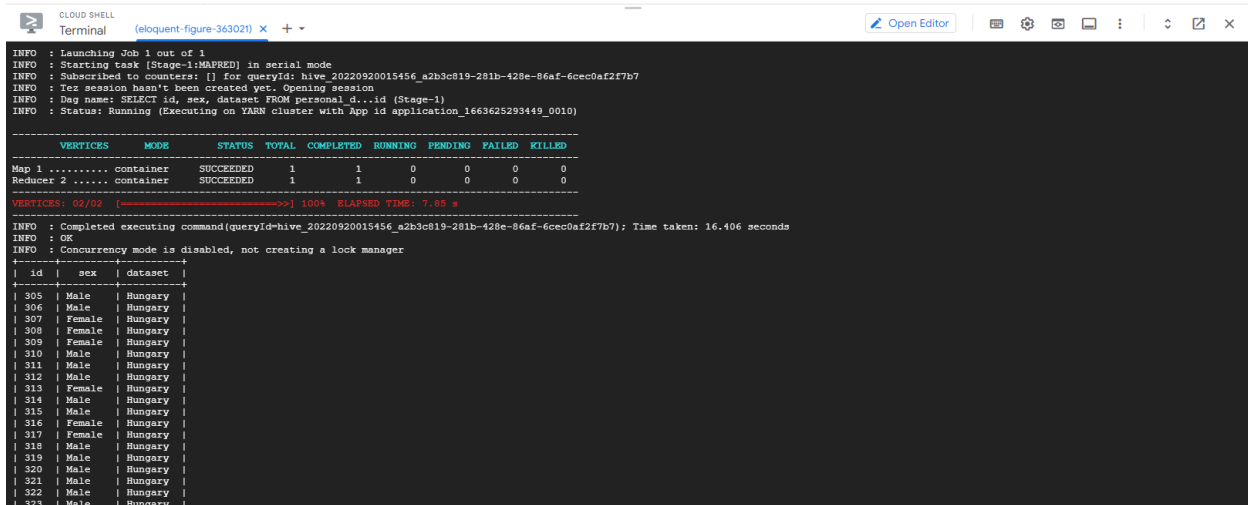
```
gcloud dataproc jobs submit hive \
  --cluster hive-cluster \
  --region ${REGION} \
  --execute "
    CREATE EXTERNAL TABLE diet
```

```
(weight INT,diet STRING)
STORED AS PARQUET
LOCATION 'gs://${WAREHOUSE_BUCKET}/datasets/diet';"
```

Sample query - Querying Hive with the Dataproc Jobs API

```
gcloud dataproc jobs submit hive \
--cluster hive-cluster \
--region ${REGION} \
--execute "
SELECT *
FROM transactions
LIMIT 10;"
```

Sample Screenshot



```
CLOUD SHELL
Terminal (eloquent-figure-363021) x +
Open Editor

INFO : Launching Job 1 out of 1
INFO : Starting task (Stage-1:MAPRED) in serial mode
INFO : Subscribed to counters: {} for queryId: hive_20220920015456_a2b3c819-281b-428e-86af-6cec0af2f7b7
INFO : Ter session hasn't been created yet. Opening session
INFO : Job name: SELECT id, sex, dataset FROM personal_details (Stage-1)
INFO : Status: Running [Executing on YARN cluster with App id application_1663625293449_0010]

-----
VERTICES      MODE      STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
-----
Map 1 ..... container  SUCCEEDED    1         1         0         0         0         0
Reducer 2 ..... container  SUCCEEDED    1         1         0         0         0         0
-----
VERTICES: 02/02 [=====] 100% ELAPSED TIME: 7.85 s

INFO : Completed executing command(queryId=hive_20220920015456_a2b3c819-281b-428e-86af-6cec0af2f7b7); Time taken: 16.406 seconds
INFO : OK
INFO : Concurrency mode is disabled, not creating a lock manager

+-----+
| id | sex | dataset |
+-----+
| 305 | Male | Hungary |
| 306 | Male | Hungary |
| 307 | Female | Hungary |
| 308 | Female | Hungary |
| 309 | Female | Hungary |
| 310 | Male | Hungary |
| 311 | Male | Hungary |
| 312 | Male | Hungary |
| 313 | Female | Hungary |
| 314 | Male | Hungary |
| 315 | Male | Hungary |
| 316 | Female | Hungary |
| 317 | Female | Hungary |
| 318 | Male | Hungary |
| 319 | Male | Hungary |
| 320 | Male | Hungary |
| 321 | Male | Hungary |
| 322 | Male | Hungary |
| 323 | Male | Hungary |
```

Questions. (10 points each)

1. Write a Hive query to retrieve id, age and dataset where the dataset value is “Hungary”.
2. Write a Hive query to retrieve id, age, dataset, chol and fbs and sort the values in ascending order of id.
3. Modify the query in Q2 by using “DISTRIBUTE BY” and explain the difference.
4. Modify the query in Q2 by using “CLUSTER BY” and explain the difference between Q2, Q3 and Q4.
5. Write a query to join tables *personal_details*, *health_details* and *diet*. Observe the results and point out the error/issue if any.

Theory Questions - (10 points each)

1. In your own words, describe the working of Hive. (Hint - how hive is on top of hadoop and internally what techniques are used for querying)
2. List out the advantages and disadvantages of HIVE