

## Assignment 8 : Hive Basic Assignment Problems

**Note:** Due to non working of the Hive terminal in my VM I am just uploading the programs on git hub. The execution for the same will be uploaded soon.

### Problem Statement

#### Task 1

Create a database named 'custom'.

Create a table named temperature\_data inside custom having below fields:

1. date (mm-dd-yyyy) format
2. zip code
3. temperature

The table will be loaded from comma-delimited file.

Load the dataset.txt (which is ',' delimited) in the table.

#### Terminal Execution :

```
[acadgild@localhost ~]$ jps
3179 Jps
[acadgild@localhost ~]$ sudo service sshd start
[sudo] password for acadgild:
[acadgild@localhost ~]$ start-all.sh
This script is Deprecated. Instead use start-dfs.sh and start-yarn.sh
18/07/11 21:36:55 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your
platform... using builtin-java classes where applicable
Starting namenodes on [localhost]
localhost: starting namenode, logging to
/home/acadgild/install/hadoop/hadoop-2.6.5/logs/hadoop-acadgild-namenode-localhost.localdomain.ou
t
localhost: starting datanode, logging to
/home/acadgild/install/hadoop/hadoop-2.6.5/logs/hadoop-acadgild-datanode-localhost.localdomain.out
Starting secondary namenodes [0.0.0.0]
0.0.0.0: starting secondarynamenode, logging to
/home/acadgild/install/hadoop/hadoop-2.6.5/logs/hadoop-acadgild-secondarynamenode-localhost.local
domain.out
18/07/11 21:37:56 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your
platform... using builtin-java classes where applicable
starting yarn daemons
starting resourcemanager, logging to
/home/acadgild/install/hadoop/hadoop-2.6.5/logs/yarn-acadgild-resourcemanager-localhost.localdomai
```

```

n.out
localhost: starting nodemanager, logging to
/home/acadgild/install/hadoop/hadoop-2.6.5/logs/yarn-acadgild-nodemanager-localhost.localdomain.out
[acadgild@localhost ~]$ jps
3428 DataNode
3556 SecondaryNameNode
3766 ResourceManager
3334 NameNode
3864 NodeManager
3903 Jps

[acadgild@localhost ~]$ hive
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in
[jar:file:/home/acadgild/install/hive/apache-hive-2.3.3-bin/lib/log4j-slf4j-impl-2.6.2.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in
[jar:file:/home/acadgild/install/hadoop/hadoop-2.6.5/share/hadoop/common/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]

Logging initialized using configuration in
jar:file:/home/acadgild/install/hive/apache-hive-2.3.3-bin/lib/hive-common-2.3.3.jar!/hive-log4j2.properties Async: true
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a
different execution engine (i.e. spark, tez) or using Hive 1.X releases.
hive>

```

### **Problem 1: Create a database named 'custom'.**

#### **Commands :**

1. show databases;
2. create IF NOT EXISTS database custom;
3. use custom;

**Problem 2: Create a table named temperature\_data inside custom having below fields:**

- 1. date (mm-dd-yyyy) format**
- 2. zip code**
- 3. temperature**

**Commands:**

```
Create table IF NOT EXISTS temperature_data
( date timestamp(MM-DD-YYYY),
  zip_code string,
  temperature int)
row format delimited
field terminated by ',';
```

**Problem 3 : Load the dataset.txt (which is ',' delimited) in the table.**

**Commands:**

```
load data local inpath '/home/acadgild/Desktop/dataset_Session 14.txt' into table temperature_data;
```

**Task 2**

- **Fetch date and temperature from temperature\_data where zip code is greater than 300000 and less than 399999.**

**Solution :**

```
Select date, temperature
```

```
from temperature_data
where zip_code >300000 && zip_code <399999;
```

- Calculate maximum temperature corresponding to every year from temperature\_data table.

```
Create table temp as Select Max(temperature) from temperature_data group by date_format(date, 'YYYY');
```

```
Select * from temp;
```

- Calculate maximum temperature from temperature\_data table corresponding to those years which have at least 2 entries in the table.

```
Create table temp1 as Select Max(temperature) from temperature_data group by date_format(date, 'YYYY');
```

```
Select Max(temp) from temp1 where count(date_format(date, 'YYYY')) >= 2
```

- Create a view on the top of last query, name it temperature\_data\_vw.

**Solution :**

```
Create View temperature_data_vw AS Select Max(temp) from temp1 where count(date_format(date, 'YYYY')) >= 2 group by date_format(date, 'YYYY');
```

- Export contents from temperature\_data\_vw to a file in local file system, such that each file is '|' delimited.

**Solution :**

```
[acadgild@localhost ~]$ pwd
/home/acadgild
[acadgild@localhost ~]$ ls /home/acadgild/Desktop
Assignment_7.11199.odt  dataset_Session 14.txt  problem3.pig~  query4.pig~
Assignment_8.11199.odt  PIG                problem4.pig~  query5.pig~
```

```
Assignment Done      problem1.pig      query1.pig~  README
Assignment_Jars      problem1.pig~    query2.pig~  sample.txt~
Datasets              problem2.pig~    query3.pig~  word_count.pig~
[acadgild@localhost ~]$ mkdir /home/acadgild/Desktop/hive_local
[acadgild@localhost ~]$ ls /home/acadgild/Desktop/
Assignment_7.11199.odt  hive_local  problem4.pig~  README
Assignment_8.11199.odt  PIG        query1.pig~   sample.txt~
Assignment Done        problem1.pig  query2.pig~   word_count.pig~
Assignment_Jars        problem1.pig~ query3.pig~
Datasets                problem2.pig~ query4.pig~
dataset_Session 14.txt  problem3.pig~ query5.pig~
```

```
>hive insert overwrite local directory '/home/acadgild/Desktop/hive_local'
-> temperature_data_vw
-> row format delimited field terminated by '|';
```