

Big Data And Hadoop

Session 15 – Assignment 2

Dataset Description:

The data set consists of the following fields.

Athlete: This field consists of the athlete name

Age: This field consists of athlete ages

Country: This field consists of the country names which participated in Olympics

Year: This field consists of the year

Closing Date: This field consists of the closing date of ceremony

Sport: Consists of the sports name

Gold Medals: No. of Gold medals

Silver Medals: No. of Silver medals

Bronze Medals: No. of Bronze medals

Total Medals: Consists of total no. of medals

Problem Statement:

1. Write a Hive program to find the number of medals won by each country in swimming.
2. Write a Hive program to find the number of medals that India won year wise.
3. Write a Hive Program to find the total number of medals each country won.
4. Write a Hive program to find the number of gold medals each country won.

Solution:

Input File

The input file is present on the local file system at **/home/acadgild/Abhilasha/hive** as follows:

```
acadgild@localhost:~/Abhilasha/hive
File Edit View Search Terminal Help
[acadgild@localhost ~]$ cd Abhilasha/hive
[acadgild@localhost hive]$ ls -l
total 552
-rw-rw-r--. 1 acadgild acadgild 3058 Sep 19 08:57 commands
-rw-rw-r--. 1 acadgild acadgild 2805 Sep 18 22:16 commands~
-rw-rw-r--. 1 acadgild acadgild 170 Sep 17 14:17 complexData
-rw-rw-r--. 1 acadgild acadgild 437 Sep 16 19:29 dataset_Session14.txt
-rw-rw-r--. 1 acadgild acadgild 159 Sep 19 08:49 emp_Details
-rw-rw-r--. 1 acadgild acadgild 159 Sep 19 08:24 emp_Details~
-rw-rw-r--. 1 acadgild acadgild 84 Sep 17 13:43 empDetails~
-rw-rw-r--. 1 acadgild acadgild 107 Sep 18 22:00 employee.csv
-rw-rw-r--. 1 acadgild acadgild 107 Sep 18 21:51 employee.csv~
-rw-rw-r--. 1 acadgild acadgild 518669 Sep 19 22:14 olympix_data.csv
drwxrwxr-x. 2 acadgild acadgild 4096 Sep 19 08:53 output
-rw-rw-r--. 1 acadgild acadgild 170 Sep 17 14:17 Unsaved Document 1~
[acadgild@localhost hive]$
```

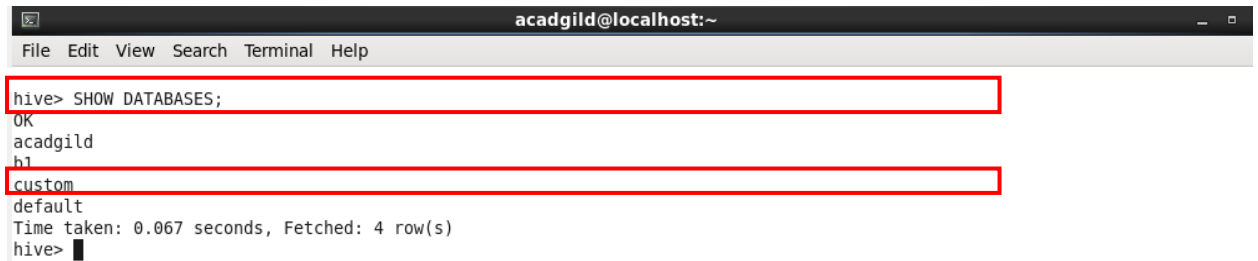
Start hive: We start the hive command line by executing the command hive as shown below:



```
acadgild@localhost:~  
File Edit View Search Terminal Help  
[acadgild@localhost ~]$ hive  
Logging initialized using configuration in jar:file:/usr/local/hive/lib/hive-common-0.14.0.jar!/hive-log4j.properties  
SLF4J: Class path contains multiple SLF4J bindings.  
SLF4J: Found binding in [jar:file:/usr/local/hive/lib/hive-jdbc-0.14.0-standalone.jar!/org/slf4j/impl/StaticLoggerBinder.class]  
SLF4J: Found binding in [jar:file:/usr/local/hadoop-2.6.0/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.slf4j.impl.Log4jLoggerFactory]  
hive>
```

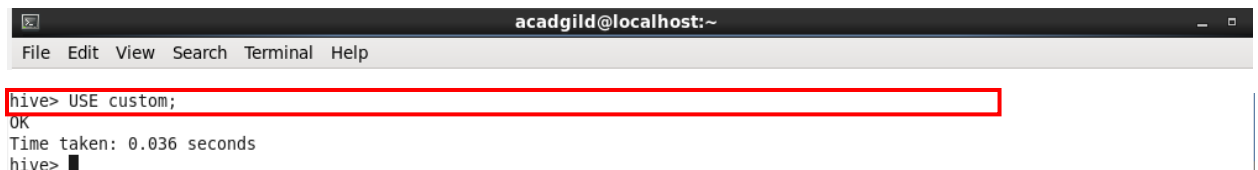
The above snapshot also shows that hive prompt has started. A pre-requisite to use hive is to start mysql server. This was done using the command `sudo service mysqld start`.

Step 1: We use **SHOW DATABASES** command to list the databases present. The database we will be using is **custom** as shown below:



```
acadgild@localhost:~  
File Edit View Search Terminal Help  
hive> SHOW DATABASES;  
OK  
acadgild  
h1  
custom  
default  
Time taken: 0.067 seconds, Fetched: 4 row(s)  
hive>
```

Step 2: We use **USE custom** command to make use of custom database, as shown below:



```
acadgild@localhost:~  
File Edit View Search Terminal Help  
hive> USE custom;  
OK  
Time taken: 0.036 seconds  
hive>
```

Step 3: We create the table using **CREATE TABLE** command. The fields of the table are as mentioned above in the description of dataset. The fields are separated by the delimiter ' '.

```
acadgild@localhost:~  
File Edit View Search Terminal Help  
hive> CREATE TABLE olympic  
(  
  athleteName STRING,  
  age INT,  
  country STRING,  
  year INT,  
  closingDate STRING,  
  sportName STRING,  
  goldMedal INT,  
  silverMedal INT,  
  bronzeMedal INT,  
  totalMedals INT  
)  
row format delimited fields terminated by '  ';  
OK  
Time taken: 0.077 seconds  
hive> █
```

Step 4: **SHOW TABLES** command will help us verify that the table is created.

```
acadgild@localhost:~  
File Edit View Search Terminal Help  
hive> SHOW TABLES;  
OK  
empdetails  
olympic  
temperature_data  
temperature_data_vw  
Time taken: 0.062 seconds, Fetched: 4 row(s)  
hive> █
```

Step 6: Next is to load the data from input file, which is located at **/home/acadgild/Abhilasha/hive** as follows. We use the **LOAD** command and use the keyword **LOCAL** to specify that the file is present in the local file system and not HDFS.

```
acadgild@localhost:~  
File Edit View Search Terminal Help  
hive> LOAD DATA LOCAL INPATH '/home/acadgild/Abhilasha/hive/olympix_data.csv'  
OVERWRITE INTO TABLE olympic;  
Loading data to table custom.olympic  
Table custom.olympic stats: [numFiles=1, numRows=0, totalSize=518669, rawDataSize=0]  
OK  
Time taken: 1.596 seconds  
hive> █
```

Problem Statement 1: Write a Hive program to find the number of medals won by each country in swimming.

Solution : The query used to solve the above problem statement is

```
SELECT country, SUM(totalMedals) FROM olympic where  
sportName='Swimming' GROUP BY country;
```

We have used the predicate **sportName='Swimming'** to get details only for Swimming and used **GROUP BY** to get details per country as follows:

```
acadgild@localhost:~  
File Edit View Search Terminal Help  
hive> SELECT country, SUM(totalMedals) FROM olympic where sportName='Swimming' GROUP BY country;  
Query ID = acadgild_20170919224545_9bb681f7-1d85-4750-9a5a-9addc0506ad4  
Total jobs = 1  
Launching Job 1 out of 1  
Number of reduce tasks not specified. Estimated from input data size: 1  
In order to change the average load for a reducer (in bytes):  
  set hive.exec.reducers.bytes.per.reducer=<number>  
In order to limit the maximum number of reducers:  
  set hive.exec.reducers.max=<number>  
In order to set a constant number of reducers:  
  set mapreduce.job.reduces=<number>  
Starting Job = job_1505839534753_0003, Tracking URL = http://localhost:8088/proxy/application_1505839534753_0003/  
Kill Command = /home/acadgild/hadoop-2.6.0/bin/hadoop job -kill job_1505839534753_0003
```

The output of the query is as follows:

```
acadgild@localhost:~  
File Edit View Search Terminal Help  
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 5.58 sec HDFS Read: 518899 HDFS Write: 386 SUCCESS  
Total MapReduce CPU Time Spent: 5 seconds 580 msec  
OK  
Argentina 1  
Australia 163  
Austria 3  
Belarus 2  
Brazil 8  
Canada 5  
China 35  
Costa Rica 2  
Croatia 1  
Denmark 1  
France 39  
Germany 32  
Great Britain 11  
Hungary 9  
Italy 16  
Japan 43  
Lithuania 1  
Netherlands 46  
Norway 2  
Poland 3  
Romania 6  
Russia 20  
Serbia 1  
Slovakia 2  
Slovenia 1  
South Africa 11  
South Korea 4  
Spain 3  
Sweden 9  
Trinidad and Tobago 1  
Tunisia 3  
Ukraine 7  
United States 267  
Zimbabwe 7  
Time taken: 28.603 seconds, Fetched: 34 row(s)  
hive>
```

Problem Statement 2: Write a Hive program to find the number of medals that India won year wise.

Solution: The query used to solve the above problem statement is

```
SELECT year, SUM(totalMedals) FROM olympic where country='India' GROUP BY year;
```

We have used the predicate `country='India'` to get details only for India and used **GROUP BY** to get details per year as follows:

```
acadgild@localhost:~  
File Edit View Search Terminal Help  
hive> SELECT year, SUM(totalMedals) FROM olympic where country='India' GROUP BY year;  
Query ID = acadgild_20170919224646_2f59cc29-7d3d-4676-b3ee-4442c25bd4bc  
Total jobs = 1  
Launching Job 1 out of 1  
Number of reduce tasks not specified. Estimated from input data size: 1  
In order to change the average load for a reducer (in bytes):  
  set hive.exec.reducers.bytes.per.reducer=<number>  
In order to limit the maximum number of reducers:  
  set hive.exec.reducers.max=<number>  
In order to set a constant number of reducers:  
  set mapreduce.job.reduces=<number>  
Starting Job = job_1505839534753_0004, Tracking URL = http://localhost:8088/proxy/application_1505839534753_0004/  
Kill Command = /home/acadgild/hadoop-2.6.0/bin/hadoop job -kill job_1505839534753_0004  
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1  
2017-09-19 22:47:03,869 Stage-1 map = 0%, reduce = 0%  
2017-09-19 22:47:12,641 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.99 sec  
2017-09-19 22:47:20,210 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 5.56 sec  
MapReduce Total cumulative CPU time: 5 seconds 560 msec  
Ended Job = job_1505839534753_0004  
MapReduce Jobs Launched:  
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 5.56 sec HDFS Read: 518899 HDFS Write: 28 SUCCESS  
Total MapReduce CPU Time Spent: 5 seconds 560 msec  
OK  
2000 1  
2004 1  
2008 3  
2012 6  
Time taken: 28.159 seconds, Fetched: 4 row(s)  
hive>
```

The output is shown in the above snapshot.

Problem Statement 3: Write a Hive Program to find the total number of medals each country won.

Solution: The query used to solve the above problem statement is

```
SELECT country, SUM(goldMedal) FROM olympic GROUP BY country;
```

We have used **GROUP BY** to get details per country as follows:

```
acadgild@localhost:~  
File Edit View Search Terminal Help  
hive> SELECT country, SUM(totalMedals) FROM olympic GROUP BY country;  
Query ID = acadgild_20170919224848_f8bc03ea-4ad3-4f8a-99f3-4bb4403809d9  
Total jobs = 1  
Launching Job 1 out of 1  
Number of reduce tasks not specified. Estimated from input data size: 1  
In order to change the average load for a reducer (in bytes):  
  set hive.exec.reducers.bytes.per.reducer=<number>  
In order to limit the maximum number of reducers:  
  set hive.exec.reducers.max=<number>  
In order to set a constant number of reducers:  
  set mapreduce.job.reduces=<number>  
Starting Job = job_1505839534753_0005, Tracking URL = http://localhost:8088/proxy/application_1505839534753_0005/  
Kill Command = /home/acadgild/hadoop-2.6.0/bin/hadoop job -kill job_1505839534753_0005  
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1  
2017-09-19 22:48:34,713 Stage-1 map = 0%, reduce = 0%  
2017-09-19 22:48:41,395 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.2 sec  
2017-09-19 22:48:50,001 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 4.66 sec  
MapReduce Total cumulative CPU time: 4 seconds 660 msec  
Ended Job = job_1505839534753_0005  
MapReduce Jobs Launched:  
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 4.66 sec HDFS Read: 518899 HDFS Write: 28 SUCCESS  
Total MapReduce CPU Time Spent: 4 seconds 660 msec  
OK  
Time taken: 28.159 seconds, Fetched: 4 row(s)  
hive>
```

We can also store this result into a file using **INSERT** command as follows. The output directory is **/home/acadgild/Abhilasha/hive/output-Query3**. The delimiter used to separate the fields in the file is ‘**,**’.

```
acadgild@localhost:~  
File Edit View Search Terminal Help  
hive> insert overwrite local directory '/home/acadgild/Abhilasha/hive/output-Query3'  
row format delimited  
fields terminated by ','  
SELECT country, SUM(totalMedals) FROM olympic GROUP BY country;  
Query ID = acadgild_20170919225151_ba95534d-be10-4b0f-a184-43974b86fa28  
Total jobs = 1  
Launching Job 1 out of 1  
Number of reduce tasks not specified. Estimated from input data size: 1  
In order to change the average load for a reducer (in bytes):  
  set hive.exec.reducers.bytes.per.reducer=<number>  
In order to limit the maximum number of reducers:  
  set hive.exec.reducers.max=<number>  
In order to set a constant number of reducers:  
  set mapreduce.job.reduces=<number>  
Starting Job = job_1505839534753_0007, Tracking URL = http://localhost:8088/proxy/application_1505839534753_0007/  
Kill Command = /home/acadgild/hadoop-2.6.0/bin/hadoop job -kill job_1505839534753_0007
```

We can see the new directory created named output as follows:

```
acadgild@localhost:~/Abhilasha/hive  
File Edit View Search Terminal Help  
[acadgild@localhost hive]$ ls -l  
total 556  
-rw-rw-r--. 1 acadgild acadgild 3058 Sep 19 08:57 commands  
-rw-rw-r--. 1 acadgild acadgild 2805 Sep 18 22:16 commands~  
-rw-rw-r--. 1 acadgild acadgild 170 Sep 17 14:17 complexData  
-rw-rw-r--. 1 acadgild acadgild 437 Sep 16 19:29 dataset Session14.txt  
-rw-rw-r--. 1 acadgild acadgild 159 Sep 19 08:49 emp_Details  
-rw-rw-r--. 1 acadgild acadgild 159 Sep 19 08:24 emp_Details~  
-rw-rw-r--. 1 acadgild acadgild 84 Sep 17 13:43 empDetails~  
-rw-rw-r--. 1 acadgild acadgild 107 Sep 18 22:00 employee.csv  
-rw-rw-r--. 1 acadgild acadgild 107 Sep 18 21:51 employee.csv~  
-rw-rw-r--. 1 acadgild acadgild 518669 Sep 19 22:14 olympix_data.csv  
drwxrwxr-x. 2 acadgild acadgild 4096 Sep 19 08:53 output  
drwxrwxr-x. 2 acadgild acadgild 4096 Sep 19 22:52 output-Query3  
-rw-rw-r--. 1 acadgild acadgild 170 Sep 17 14:17 Unsaved Document 1~  
[acadgild@localhost hive]$
```

The output directory has the following output file.

```
acadgild@localhost:~/Abhilasha/hive/output-Query3  
File Edit View Search Terminal Help  
[acadgild@localhost hive]$ cd output-Query3/  
[acadgild@localhost output-Query3]$ ls -l  
total 4  
-rw-r--r--. 1 acadgild acadgild 1315 Sep 19 22:52 000000_0  
[acadgild@localhost output-Query3]$
```

Problem Statement 4: Write a Hive Program to find the total number of medals each country won.

Solution: The query used to solve the above problem statement is

```
SELECT country, SUM(goldMedal) FROM olympic GROUP BY country;
```

We have used **GROUP BY** to get details per country and used **SUM** function to get the total as follows:

```
acadgild@localhost:~  
File Edit View Search Terminal Help  
hive> SELECT country, SUM(goldMedal) FROM olympic GROUP BY country;  
Query ID = acadgild_20170919225454_8e8dee70-283a-4510-9773-0813e6bb3be9  
Total jobs = 1  
Launching Job 1 out of 1  
Number of reduce tasks not specified. Estimated from input data size: 1  
In order to change the average load for a reducer (in bytes):  
  set hive.exec.reducers.bytes.per.reducer=<number>  
In order to limit the maximum number of reducers:  
  set hive.exec.reducers.max=<number>  
In order to set a constant number of reducers:  
  set mapreduce.job.reduces=<number>  
Starting Job = job_1505839534753_0008, Tracking URL = http://localhost:8088/proxy/application_1505839534753_0008/  
Kill Command = /home/acadgild/hadoop-2.6.0/bin/hadoop job -kill job_1505839534753_0008  
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1  
2017-09-19 22:54:29,498 Stage-1 map = 0%, reduce = 0%
```

We can also store this result into a file using **INSERT** command as follows. The output directory is **/home/acadgild/Abhilasha/hive/output-Query4**. The delimiter used to separate the fields in the file is ‘**,**’.

```
acadgild@localhost:~  
File Edit View Search Terminal Help  
hive> insert overwrite local directory '/home/acadgild/Abhilasha/hive/output-Query4'  
row format delimited  
fields terminated by ','  
SELECT country, SUM(goldMedal) FROM olympic GROUP BY country;  
Query ID = acadgild_20170919225555_e7dccc19a-84ca-4890-b71f-d676a0d75f32  
Total jobs = 1  
Launching Job 1 out of 1  
Number of reduce tasks not specified. Estimated from input data size: 1  
In order to change the average load for a reducer (in bytes):  
  set hive.exec.reducers.bytes.per.reducer=<number>  
In order to limit the maximum number of reducers:  
  set hive.exec.reducers.max=<number>  
In order to set a constant number of reducers:  
  set mapreduce.job.reduces=<number>  
Starting Job = job_1505839534753_0009, Tracking URL = http://localhost:8088/proxy/application_1505839534753_0009/  
Kill Command = /home/acadgild/hadoop-2.6.0/bin/hadoop job -kill job_1505839534753_0009
```

We can see the new directory created named output as follows:

```
acadgild@localhost:~/Abhilasha/hive  
File Edit View Search Terminal Help  
[acadgild@localhost hive]$ ls -l  
total 560  
-rw-rw-r--. 1 acadgild acadgild 3058 Sep 19 08:57 commands  
-rw-rw-r--. 1 acadgild acadgild 2805 Sep 18 22:16 commands~  
-rw-rw-r--. 1 acadgild acadgild 170 Sep 17 14:17 complexData  
-rw-rw-r--. 1 acadgild acadgild 437 Sep 16 19:29 dataset_Session14.txt  
-rw-rw-r--. 1 acadgild acadgild 159 Sep 19 08:49 emp_Details  
-rw-rw-r--. 1 acadgild acadgild 159 Sep 19 08:24 emp_Details~  
-rw-rw-r--. 1 acadgild acadgild 84 Sep 17 13:43 empDetails~  
-rw-rw-r--. 1 acadgild acadgild 107 Sep 18 22:00 employee.csv  
-rw-rw-r--. 1 acadgild acadgild 107 Sep 18 21:51 employee.csv~  
-rw-rw-r--. 1 acadgild acadgild 518669 Sep 19 22:14 olympix_data.csv  
drwxrwxr-x. 2 acadgild acadgild 4096 Sep 19 08:53 output  
drwxrwxr-x. 2 acadgild acadgild 4096 Sep 19 22:52 output-Query3  
drwxrwxr-x. 2 acadgild acadgild 4096 Sep 19 22:55 output-Query4  
-rw-rw-r--. 1 acadgild acadgild 170 Sep 17 14:17 Unsaved Document 1~  
[acadgild@localhost hive]$
```

The output directory has the following output file.

```
acadgild@localhost:~/Abhilasha/hive/output-Query4
File Edit View Search Terminal Help
[acadgild@localhost hive]$ cd output-Query4
[acadgild@localhost output-Query4]$ ls -l
total 4
-rw-r--r--. 1 acadgild acadgild 1276 Sep 19 22:55 000000 0
[acadgild@localhost output-Query4]$
```