

ASSIGNMENT 7.2

DATE SET DESCRIPTION:

Olympix_data.csv

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 fields 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

Let's create a table in Hive based on above description.

Table creation query:

```
CREATE TABLE IF NOT EXISTS olympics_data
```

```
(  
  athlete_name STRING,  
  age INT,  
  country STRING,  
  year INT,  
  closing_date STRING,  
  sport STRING,  
  gold_medals INT,  
  silver_medals INT,  
  bronze_medals INT,  
  total_medals INT  
)
```

```
ROW FORMAT DELIMITED
```

```
FIELDS TERMINATED BY '\t';
```

```

hive> CREATE TABLE IF NOT EXISTS olympics_data
> (
>   athlete_name STRING,
>   age INT,
>   country STRING,
>   year INT,
>   closing_date STRING,
>   sport STRING,
>   gold_medals INT,
>   silver_medals INT,
>   bronze_medals INT,
>   total_medals INT
> )
> ROW FORMAT DELIMITED
> FIELDS TERMINATED BY '\t';

```

```

OK
Time taken: 0.131 seconds
hive> SHOW TABLES;
OK
employee
olympics_data
temperature_data
temperature_data_new
Time taken: 0.042 seconds, Fetched: 4 row(s)

```

Let's load the data from given CSV file into the table we created above.

Data load query:

```

LOAD DATA LOCAL INPATH '/home/acadgild/olympix_data.csv' INTO TABLE
olympics_data;

```

```

hive> LOAD DATA LOCAL INPATH '/home/acadgild/olympix_data.csv' INTO TABLE olympics_data;
Loading data to table custom.olympics_data
Table custom.olympics_data stats: [numFiles=1, totalSize=518669]
OK
Time taken: 0.609 seconds

```

```

SELECT * FROM olympics_data;

```

```

Xue Haifeng      28      China      2008      08-24-08      Archery      0      0      1      1
Chen Li-Ju       23      Chinese Taipei 2004      08-29-04      Archery      0      0      1      1
Chen Szu-Yuan    23      Chinese Taipei 2004      08-29-04      Archery      0      1      0      1
Tim Cuddihy      17      Australia     2004      08-29-04      Archery      0      0      1      1
Marco Gialazzo   21      Italy         2004      08-29-04      Archery      1      0      0      1
He Ying 27      China         2004      08-29-04      Archery      0      1      0      1
Dmytro Hrachov  20      Ukraine       2004      08-29-04      Archery      0      0      1      1
Im Dong-Hyeon   19      South Korea   2004      08-29-04      Archery      1      0      0      1
Jang Yong-Ho    28      South Korea   2004      08-29-04      Archery      1      0      0      1
Lin Sang        26      China         2004      08-29-04      Archery      0      1      0      1
Liu Ming-Huang  19      Chinese Taipei 2004      08-29-04      Archery      0      1      0      1
Park Gyeong-Mo  28      South Korea   2004      08-29-04      Archery      1      0      0      1
Viktor Ruban    23      Ukraine       2004      08-29-04      Archery      0      1      1      1
Oleksandr Serdiuk 26      Ukraine       2004      08-29-04      Archery      0      0      1      1
Wang Cheng-Pang 17      Chinese Taipei 2004      08-29-04      Archery      0      1      0      1
Alison Williamson 32      Great Britain 2004      08-29-04      Archery      0      0      1      1
Wu Hui-Ju       21      Chinese Taipei 2004      08-29-04      Archery      0      0      1      1
Hiroshi Yamamoto 41      Japan         2004      08-29-04      Archery      0      1      0      1
Yuan Shu-Chi    19      Chinese Taipei 2004      08-29-04      Archery      0      0      1      1
Yun Mi-Jin      21      South Korea   2004      08-29-04      Archery      1      0      0      1
Zhang Juanjuan  23      China         2004      08-29-04      Archery      0      1      0      1
Matteo Bisiani  24      Italy         2000      10-01-00      Archery      0      1      0      1
Nataliya Burdeina 26      Ukraine       2000      10-01-00      Archery      0      1      0      1
Ilario Di Buò    43      Italy         2000      10-01-00      Archery      0      1      0      1
Simon Fairweather 30      Australia     2000      10-01-00      Archery      1      0      0      1
Michele Frangilli 24      Italy         2000      10-01-00      Archery      0      1      0      1
Jang Yong-Ho    24      South Korea   2000      10-01-00      Archery      1      0      0      1
Butch Johnson    45      United States 2000      10-01-00      Archery      0      0      1      1
Kim Cheong-Tae  20      South Korea   2000      10-01-00      Archery      1      0      0      1
Barbara Mensing 39      Germany       2000      10-01-00      Archery      0      1      1      1
O Gyo-Mun       28      South Korea   2000      10-01-00      Archery      1      0      0      1
Cornelia Pfohl  29      Germany       2000      10-01-00      Archery      0      1      1      1
Olena Sadovnycha 32      Ukraine       2000      10-01-00      Archery      0      1      0      1
Kateryna Serdiuk 17      Ukraine       2000      10-01-00      Archery      0      1      0      1
Wietse van Alten 21      Netherlands   2000      10-01-00      Archery      0      0      1      1
Sandra Wagner-Sachse 31      Germany       2000      10-01-00      Archery      0      0      1      1
Rod White       23      United States 2000      10-01-00      Archery      0      0      1      1
Time taken: 0.127 seconds, Fetched: 8618 row(s)
hive>

```

Problem Statement:

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

Hive query:

```
SELECT country, SUM(total_medals)
```

```
FROM olympics_data
```

```
WHERE sport = 'Swimming'
```

```
GROUP BY country;
```

Comments: The query uses SUM() function to calculate total medals won by each country, groups the result set country wise with the constraint of specific sport, swimming in this case.

```
hive> SELECT country, SUM(total_medals)
> FROM olympics_data
> WHERE sport = 'Swimming'
> GROUP BY country;
query ID = aca0g1l0_20171224225959_a2T38b9d-85a9-43f2-845e-9fdb59c33732
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_1514121655167_0007, Tracking URL = http://localhost:8088/proxy/application_1514121655167_0007/
```

Output:

```
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 7.73 sec HDFS Read: 518905 HDFS Write: 386 SUCCESS
Total MapReduce CPU Time Spent: 7 seconds 730 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: 105.318 seconds, Fetched: 34 row(s)
hive>
```

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

Hive query:

```
SELECT year, SUM(total_medals)
```

```
FROM olympics_data
```

```
WHERE country = 'India'
```

```
GROUP BY year;
```

Comments: The query uses SUM() function to calculate total medals won by India, groups the result set by its year of participation.

Query with output:

```
acadgild@localhost:usr/local/hadoop-2.6.0/sbin
File Edit View Search Terminal Help
Sweden 9
Trinidad and Tobago 1
Tunisia 3
Ukraine 7
United States 267
Zimbabwe 7
Time taken: 105.318 seconds. Fetched: 34 row(s)
hive> SELECT year, SUM(total_medals)
> FROM olympics_data
> WHERE country = 'India'
> GROUP BY year;
Query ID = acadgild_20171224232727_8ce5327f-8fde-43f8-b6bd-c99107fe8a40
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_1514121655167_0008, Tracking URL = http://localhost:8088/proxy/application_1514121655167_0008/
Kill Command = /home/acadgild/hadoop-2.6.0/bin/hadoop job -kill job_1514121655167_0008
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2017-12-24 23:28:38,025 Stage-1 map = 0%, reduce = 0%
2017-12-24 23:28:58,032 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 3.29 sec
2017-12-24 23:29:16,666 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 6.19 sec
MapReduce Total cumulative CPU time: 6 seconds 190 msec
Ended Job = job_1514121655167_0008
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 6.19 sec HDFS Read: 518905 HDFS Write: 28 SUCCESS
Total MapReduce CPU Time Spent: 6 seconds 190 msec
OK
2000 1
2004 1
2008 3
2012 6
Time taken: 84.352 seconds, Fetched: 4 row(s)
hive>
```

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

Hive query:

```
SELECT country, SUM(total_medals)
```

```
FROM olympics_data
```

```
GROUP BY country;
```

Comments: The query uses SUM() function to calculate total medals won and groups the result set by country.

```
hive> SELECT country, SUM(total_medals)
> FROM olympics_data
> GROUP BY country;
Query ID = acaug10_20171224233636_47511870-99f1-4fed-8796-53287749ea3a
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_1514121655167_0009, Tracking URL = http://localhost:8088/proxy/application_1514121655167_0009/
```

Output:

```
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 3.93 sec HDFS Read: 518905 HDFS Write: 1315 SUCCESS
Total MapReduce CPU Time Spent: 3 seconds 930 msec
OK
Afghanistan      2
Algeria          8
Argentina        141
Armenia          10
Australia        609
Austria          91
Azerbaijan       25
Bahamas          24
Bahrain          1
Barbados         1
Belarus          97
Belgium          18
Botswana         1
Brazil           221
Bulgaria         41
Cameroon         20
Canada           370
Chile            22
China            530
Chinese Taipei   20
Colombia         13
Costa Rica       2
Croatia          81
Cuba             188
Cyprus           1
Czech Republic   81
Denmark          89
Dominican Republic 5
Ecuador          1
Egypt            8
Eritrea          1
Estonia          18
Ethiopia         29
```

| | | |
|---------------|-----|-----|
| Ecuador | 1 | |
| Egypt | 8 | |
| Eritrea | 1 | |
| Estonia | 18 | |
| Ethiopia | | 29 |
| Finland | 118 | |
| France | 318 | |
| Gabon | 1 | |
| Georgia | 23 | |
| Germany | 629 | |
| Great Britain | | 322 |
| Greece | 59 | |
| Grenada | 1 | |
| Guatemala | | 1 |
| Hong Kong | | 3 |
| Hungary | 145 | |
| Iceland | 15 | |
| India | 11 | |
| Indonesia | | 22 |
| Iran | 24 | |
| Ireland | 9 | |
| Israel | 4 | |
| Italy | 331 | |
| Jamaica | 80 | |
| Japan | 282 | |
| Kazakhstan | | 42 |
| Kenya | 39 | |
| Kuwait | 2 | |
| Kyrgyzstan | | 3 |
| Latvia | 17 | |
| Lithuania | | 30 |
| Macedonia | | 1 |
| Malaysia | | 3 |
| Mauritius | | 1 |
| Mexico | 38 | |
| Moldova | 5 | |
| Mongolia | | 10 |
| Montenegro | | 14 |
| Morocco | 11 | |

| | | |
|-----------------------|------|-----|
| Panama | 1 | |
| Paraguay | | 17 |
| Poland | 80 | |
| Portugal | | 9 |
| Puerto Rico | | 2 |
| Qatar | 3 | |
| Romania | 123 | |
| Russia | 768 | |
| Saudi Arabia | | 6 |
| Serbia | 31 | |
| Serbia and Montenegro | | 38 |
| Singapore | | 7 |
| Slovakia | | 35 |
| Slovenia | | 25 |
| South Africa | | 25 |
| South Korea | | 308 |
| Spain | 205 | |
| Sri Lanka | | 1 |
| Sudan | 1 | |
| Sweden | 181 | |
| Switzerland | | 93 |
| Syria | 1 | |
| Tajikistan | | 3 |
| Thailand | | 18 |
| Togo | 1 | |
| Trinidad and Tobago | | 19 |
| Tunisia | 4 | |
| Turkey | 28 | |
| Uganda | 1 | |
| Ukraine | 143 | |
| United Arab Emirates | | 1 |
| United States | 1312 | |
| Uruguay | 1 | |
| Uzbekistan | | 19 |
| Venezuela | | 4 |
| Vietnam | 2 | |
| Zimbabwe | | 7 |

Time taken: 44.864 seconds, Fetched: 110 row(s)
hive> █

4. Write a Hive program to find the number of gold medals each country won.

Hive query:

```
SELECT country, SUM(total_medals)
```

```
FROM olympics_data
```

```
GROUP BY country;
```

Comments: The query uses SUM() function to calculate the number of gold medals won by each country and groups the result set by country.

```
hive> SELECT country, SUM(gold_medals)
> FROM olympics_data
> GROUP BY country;

Query ID = acaug10_20171224204141_007e350z-ud85-4af6-b91d-ffc16a90de2
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_1514121655167_0006, Tracking URL = http://localhost:8088/proxy/application_1514121655167_0006/
Kill Command = /home/acadgild/hadoop-2.6.0/bin/hadoop job -kill job_1514121655167_0006
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2017-12-24 20:41:52,494 Stage-1 map = 0%, reduce = 0%
2017-12-24 20:42:14,400 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.22 sec
2017-12-24 20:42:30,490 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 4.46 sec
```

Output:

```
Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 4.46 sec HDFS Read: 518905 HDFS Write: 1276 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 460 msec
OK
Afghanistan      0
Algeria          2
Argentina        49
Armenia          0
Australia        163
Austria          36
Azerbaijan       6
Bahamas          11
Bahrain          0
Barbados         0
Belarus          17
Belgium          2
Botswana         0
Brazil           46
Bulgaria         8
Cameroon         20
Canada           168
Chile            3
China            234
Chinese Taipei   2
Colombia         2
Costa Rica       0
Croatia          35
Cuba             57
Cyprus           0
Czech Republic   14
Denmark          46
Dominican Republic 3
Ecuador          0
Egypt            1
Eritrea          0
Estonia          6
Ethiopia         13
Finland          11
```

| | | |
|---------------|-----|-----|
| Eritrea | 0 | |
| Estonia | 6 | |
| Ethiopia | | 13 |
| Finland | 11 | |
| France | 108 | |
| Gabon | 0 | |
| Georgia | 6 | |
| Germany | 223 | |
| Great Britain | | 124 |
| Greece | 12 | |
| Grenada | 1 | |
| Guatemala | | 0 |
| Hong Kong | | 0 |
| Hungary | 77 | |
| Iceland | 0 | |
| India | 1 | |
| Indonesia | | 5 |
| Iran | 10 | |
| Ireland | 1 | |
| Israel | 1 | |
| Italy | 86 | |
| Jamaica | 24 | |
| Japan | 57 | |
| Kazakhstan | | 13 |
| Kenya | 11 | |
| Kuwait | 0 | |
| Kyrgyzstan | | 0 |
| Latvia | 3 | |
| Lithuania | | 5 |
| Macedonia | | 0 |
| Malaysia | | 0 |
| Mauritius | | 0 |
| Mexico | 19 | |
| Moldova | 0 | |
| Mongolia | | 2 |
| Montenegro | | 0 |
| Morocco | 2 | |
| Mozambique | | 1 |
| Netherlands | | 101 |

| | | |
|-----------------------|-----|-----|
| Norway | 97 | |
| Panama | 1 | |
| Paraguay | | 0 |
| Poland | 20 | |
| Portugal | | 1 |
| Puerto Rico | | 0 |
| Qatar | 0 | |
| Romania | 57 | |
| Russia | 234 | |
| Saudi Arabia | | 0 |
| Serbia | 1 | |
| Serbia and Montenegro | | 11 |
| Singapore | | 0 |
| Slovakia | | 10 |
| Slovenia | | 5 |
| South Africa | | 10 |
| South Korea | | 110 |
| Spain | 19 | |
| Sri Lanka | | 0 |
| Sudan | 0 | |
| Sweden | 57 | |
| Switzerland | | 21 |
| Syria | 0 | |
| Tajikistan | | 0 |
| Thailand | | 6 |
| Togo | 0 | |
| Trinidad and Tobago | | 1 |
| Tunisia | 2 | |
| Turkey | 9 | |
| Uganda | 1 | |
| Ukraine | 31 | |
| United Arab Emirates | | 1 |
| United States | 552 | |
| Uruguay | 0 | |
| Uzbekistan | | 5 |
| Venezuela | | 1 |
| Vietnam | 0 | |
| Zimbabwe | | 2 |

Time taken: 76.539 seconds, Fetched: 110 row(s)