

ASSIGNMENT- 4

Hive Commands Use Case – Petrol



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ mkdir ice4  
[cloudera@quickstart ~]$ su  
Password:  
su: incorrect password  
[cloudera@quickstart ~]$ su  
Password:  
[root@quickstart cloudera]# mount -t vboxsf datasets /home/cloudera/ice4  
[root@quickstart cloudera]# exit  
exit  
[cloudera@quickstart ~]$
```

- Firstly I have created a folder named ‘ice4’ to store the datasets.

Creation of Petrol Table in Hive and Loading of data



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ hive  
Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j.p  
roperties  
WARNING: Hive CLI is deprecated and migration to Beeline is recommended.  
hive> create table petrol (distributor_id STRING, distributor_name STRING, amt_IN String, amt_OUT STRING, vol_IN INT, vol_OUT INT, year INT) row format delimited fields terminated by ',' stored as textfile;  
OK  
Time taken: 5.37 seconds  
hive> load data local inpath '/home/cloudera/ice4/petrol.txt' into table petrol;  
Loading data to table default.petrol  
Table default.petrol stats: [numFiles=1, totalSize=19215]  
OK  
Time taken: 2.215 seconds  
hive>
```

- Then I have accessed hive using ‘hive’ command.
- Then I have used create command to create a table named petrol with columns distributor_name, amount_in, amount_out, volume_in, volume_out and year.
- Then I have loaded petrol.txt file into the petrol table.

1) In real life what is the total amount of petrol in volume sold by every distributor?

```
hive> SELECT distributor_name, SUM(vol OUT) FROM petrol GROUP BY distributor_name;
Query ID = cloudera.20220210130909.2b3b4c72-bdcf-4ac3-b793-4c4d3551b2c3
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_1644517425553_0001, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1644517425553_0001/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1644517425553_0001
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-02-10 13:09:45,616 Stage-1 map = 0%, reduce = 0%
2022-02-10 13:09:56,765 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.05 sec
2022-02-10 13:10:11,249 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 4.14 sec
MapReduce Total cumulative CPU time: 4 seconds 140 msec
Ended Job = job_1644517425553_0001
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 4.14 sec HDFS Read: 27486 HDFS Write: 76 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 140 msec
OK
Bharat 83662
Distributor name NULL
hindustan 71767
reliance 76558
shell 69266
Time taken: 51.148 seconds, Fetched: 5 row(s)
hive>
```

- Here I have used select command to select the distributor_name and total amount of petrol sold from the table and display output grouped by distributor_name.

2) Which are the top 10 distributors ID's for selling petrol and also display the amount of petrol sold in volume by them individually?

```
hive> SELECT distributor_id, vol OUT FROM petrol order by vol OUT desc limit 10;
Query ID = cloudera.20220210131414.7585b3fc-edid-45b3-a8da-d475153ff265
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks determined at compile time: 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_1644517425553_0002, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1644517425553_0002/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1644517425553_0002
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-02-10 13:14:32,642 Stage-1 map = 0%, reduce = 0%
2022-02-10 13:14:42,671 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.27 sec
2022-02-10 13:14:52,533 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 4.25 sec
MapReduce Total cumulative CPU time: 4 seconds 250 msec
Ended Job = job_1644517425553_0002
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 4.25 sec HDFS Read: 26504 HDFS Write: 120 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 250 msec
OK
TJA 944 899
SBW 094 899
VBU 276 898
DGA 625 897
QSP 953 897
FWW 043 896
NSD 085 895
NBS 194 895
EEO 991 895
JAN 463 895
Time taken: 34.689 seconds, Fetched: 10 row(s)
hive>
```

- Here I have used select command to select and display distributor_id and volume_out values from the petrol table depending on volume_out values with a limit of 10 and as we need top 10 distributor id we have used 'desc'(descending) here.

3) Find real life 10 distributor name who sold petrol in the least amount.

```
hive> SELECT distributor_id, vol_OUT FROM petrol order by vol_OUT limit 10;
Query ID = cloudera.20220210132121_fb3dabd2-a582-4c9f-9243-fa256f7fbc9d
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks determined at compile time: 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.reducers=<number>
Starting Job = job_1644517425553_0004, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1644517425553_0004/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1644517425553_0004
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-02-10 13:21:35,882 Stage-1 map = 0%, reduce = 0%
2022-02-10 13:21:44,672 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 1.58 sec
2022-02-10 13:21:54,489 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 3.45 sec
MapReduce Total cumulative CPU time: 3 seconds 450 msec
Ended Job = job_1644517425553_0004
MapReduce Jobs Launched:
  Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 3.45 sec HDFS Read: 26504 HDFS Write: 123 SUCCESS
Total MapReduce CPU Time Spent: 3 seconds 450 msec
OK
District_ID      NULL
F4D 6K2 602
H7M 4M4 603
G9F 6U7 607
R3W 2E3 608
H4P 6A9 610
Q5D 2R6 610
W0M 0R7 612
0B0 8L1 612
V0Z 0F6 612
Time taken: 29.588 seconds, Fetched: 10 row(s)
hive>
```

- Here I have used this command to select and output the values of distributor_id and volume_out values from the table with a limit of 10.
- Here we have used order by which is used to sort output values in ascending or descending order. If we don't specify any the order by sorts values in ascending order.

4) List all distributors who have this difference, along with the year and the difference which they have in that year.

```
hive> select distributor_id,year,(vol_IN-vol_OUT) as difference from petrol where (vol_IN-vol_OUT)>400;
OK
S1J 888 1627 489
D2D 229 1644 481
E6U 012 1646 483
H5N 963 1662 428
H7W 303 1670 486
N1J 2H6 1693 455
W0M 0R7 1697 435
V0Z 0F6 1708 422
K3Z 963 1726 443
F9H 457 1727 489
V8S 4P6 1732 464
Q4W 0F6 1741 489
H8M 3U0 1745 417
R6N 9L2 1746 419
J7Z 9V4 1748 446
N0D 4G5 1749 487
R3W 2E3 1767 488
N3N 0W9 1769 448
H2B 002 1779 481
C9B 308 1782 457
A0X 9H9 1784 423
E30 3Y7 1794 421
A7Z 3L9 1795 474
N2I 6C8 1823 431
C1N 0H7 1832 482
Y3F 0S2 1833 434
C0G 8G1 1843 468
H4V 0E1 1847 423
G7K 1D3 1852 424
P1B 7R8 1930 486
J1C 3G1 1934 435
I2N 751 1935 478
C8L 577 1946 466
I6D 1H5 1955 483
C4L 9Y5 1995 428
T4L 8D0 2000 452
J10 5K1 2019 488
Time taken: 0.255 seconds, Fetched: 37 row(s)
hive>
```

- I have used this command to select and display the distributor_name and respective year whose difference between volume_in and volume_out is greater than 500.
- In the given dataset as there are no data more with volume out more than 500 so I have used value as 400 and displayed the data.

5) Find the people who sell least amount of petrol using “cluster by”.

```
hive> select distributor_id,vol OUT from petrol CLUSTER by vol OUT;
Query ID = cloudera.28228214141818_15efb102-a6c2-4c86-9d5e-277f23786644
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.reducers=<number>
Starting Job = job_1644869701586_0001, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1644869701586_0001/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1644869701586_0001
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-02-14 14:19:13,085 Stage-1 map = 0%, reduce = 0%
2022-02-14 14:19:35,237 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 1.95 sec
2022-02-14 14:19:52,395 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 4.21 sec
MapReduce Total cumulative CPU time: 4 seconds 210 msec
Ended Job = job_1644869701586_0001
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 4.21 sec HDFS Read: 26343 HDFS Write: 4815 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 210 msec
OK
District_ID      NULL
F4D 0K2 602
H7M 4M4 603
G9F 6U7 607
R3W 2E3 608
H4P 6A9 610
Q5D 2R6 610
W0P 8R7 612
08D 0L1 612
V0Z 0F6 612
L9H 3K6 613
E6U 0T2 613
K9N 3H4 614
H6W 3U0 614
T4T 3K2 614
I3U 0H0 615
M1J 2M6 615
S2Z 0C3 616
E5T 2A9 617
C7W 9P0 617
N3M 0M9 617
P3Q 0Q9 618
K3E 9F5 619
B0B 0M6 619
P5C 408 619
Y8I 0V4 619
```

```
D9J 0N0 868
C1V 3X9 869
A5U 3K2 870
P6V 7Q2 870
T9V 8E2 870
T1A 3M8 870
F7S 4B3 871
U2M 9K3 872
G4P 5B4 872
H5V 0V1 873
J3A 6G9 873
E7D 2O3 873
L6U 3K6 874
06Y 2C9 877
W4T 0K2 877
A9P 7L3 878
T1A 9Q4 878
C0I 0M0 879
N2D 9Y7 879
M0N 1E7 879
ABV 7Q3 880
Y0M 8G6 881
F7B 7Y6 882
X6V 0F5 883
K5K 3S5 884
L4B 4E5 886
U5T 9M6 887
09A 5G8 888
C8Z 5S4 888
Q7T 4K7 889
M6V 9G8 889
F9Y 5E6 890
B7U 3Q4 891
J3M 0B4 891
F2C 6A5 891
U5N 7L7 894
H5Z 2W0 894
E6D 9P1 895
N6S 1P4 895
N5Q 8E5 895
J4M 4G3 895
F6W 6H3 896
08A 6Z5 897
09P 9S3 897
V8U 2T6 898
S8W 0P4 899
T3A 9M4 899
Time taken: 87.033 seconds, Fetched: 401 row(s)
hive>
```

- Here I have used this command to select and output the values of distributor_id and volume_out values from the table based on values of volume_out.
- Here I have used cluster by clause which is an alternative for sort by and distribute by clauses.
- We use cluster by if we want to store results into multiple reducers. But at the front end it is identical to sort by and distribute by.

6) Find the people who sell least amount of petrol using “distribute by”.

```
hive> select distributor_id,vol_out from petrol DISTRIBUTE BY vol_out;
Query ID = cloudera.28228214142529_0a544a42-8cb7-46b4-881c-a467ce4b6880
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.reducers=<number>
Starting Job = job_1644869701586_0002, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1644869701586_0002/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1644869701586_0002
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-02-14 14:25:34,643 Stage-1 map = 0%, reduce = 0%
2022-02-14 14:25:52,340 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 1.94 sec
2022-02-14 14:26:06,249 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 3.94 sec
MapReduce Total cumulative CPU time: 3 seconds 940 msec
Ended Job = job_1644869701586_0002
MapReduce Jobs Launched:
Stage:Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 3.94 sec HDFS Read: 26280 HDFS Write: 4815 SUCCESS
Total MapReduce CPU Time Spent: 3 seconds 940 msec
OK
B4N 861 651
E7T 305 709
P6V 702 870
N8S 719 867
J10 5K1 631
Y1Z 763 718
501 598 844
F3X 845 720
16M 403 737
A4U 982 731
N9U 423 816
Q5V 316 657
H6L 8Y6 657
L8J 155 834
H70 137 793
F75 483 871
H3P 888 785
V8U 276 898
F4D 6K2 602
F6W 6H3 896
T4T 3K2 614
M5E 1U3 655
J5C 829 658
T4L 800 640
A8M 966 691
J9B 4E8 828
```

```
U30 8L7 704
L60 451 714
V1Y 8E3 859
Z5V 913 680
54W 106 735
57H 985 718
L4B 4E3 886
H5N 9K3 633
X6Q 018 825
K7C 4C1 679
P3U 3Y9 768
N6D 1V0 673
K31 311 719
S30 4E7 718
B0B 0M6 619
D2X 4E5 686
C5I 0M0 879
K0X 585 734
P3Q 009 618
K3B 760 864
F9W 520 849
Q0U 9C9 720
P3L 7H1 687
E6U 012 613
C1V 3X9 869
D2D 2X9 670
T6B 8L6 761
K9E 1R2 787
I4J 3R2 822
L0E 3D3 814
L8E 658 838
50W 4D3 822
Q0N 5Y1 812
C7W 9P0 617
E6Q 9P1 895
Y9L 7F1 827
V6A 989 797
G3A 8D8 739
K5F 712 865
U6Z 9L5 845
F2C 6A5 891
05C 2A0 839
51J 888 657
T6Q 0L9 805
Z7Q 7C2 722
I4W 1H1 843
District_ID NULL
Time taken: 48.656 seconds, Fetched: 401 row(s)
hive>
```

- Here I have used this command to output the values of distributor_id and volume_out values from the table based on values of volume_out.
- Here I have used distribute by clause and it is used to distribute the rows among reducers. All Distribute BY columns will go to the same reducer.

7) Find the people who sell least amount of petrol using “sort by”.

```
hive> select distributor_id,volume_out from petrol SORT BY volume_out;
Query ID = cloudera.20220214142929_3d6e60fe-13a2-44b7-81df-e107e857ca8f
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_1644889701586_0003, Tracking URL = http://quickstart.cloudera:8888/proxy/application_1644889701586_0003/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1644889701586_0003
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-02-14 14:29:19,002 Stage-1 map = 0%, reduce = 0%
2022-02-14 14:29:20,011 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 1.51 sec
2022-02-14 14:29:39,694 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 3.59 sec
MapReduce Total cumulative CPU time: 3 seconds 590 msec
Ended Job = job_1644889701586_0003
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 3.59 sec HDFS Read: 26386 HDFS Write: 4815 SUCCESS
Total MapReduce CPU Time Spent: 3 seconds 590 msec
OK
District_ID NULL
F40 6K2 602
H7N 4M4 603
G9F 6U7 607
R3W 2E3 608
H4P 6A9 610
O5D 2R6 610
W0M 8R7 612
O80 0L1 612
V0Z 0F6 612
L9H 1K6 613
E6U 012 613
K9N 3H4 614
H8W 3U0 614
T4T 3K2 614
L3U 0H8 615
M1J 2H6 615
S2Z 0C3 616
E5T 2A9 617
C7W 9P0 617
N3N 069 617
P3D 009 618
K3E 9F5 619
B0B 0W6 619
P5C 408 619
Y8I 0V4 619
```

```
D9J 0N0 868
C1V 3X9 869
A5Q 3H2 870
R6V 7Q2 870
T9V 0E2 870
P1A 3W8 870
F7S 4B3 871
U2M 963 872
G4P 5B4 872
H5V 0V1 873
J3A 6G9 873
E7Q 2Q3 873
I6U 1K6 874
00Y 2C9 877
W4T 6K2 877
A9P 7L9 878
T1A 9Q4 878
C6I 6N0 879
N2Q 917 879
W0N 1E7 879
A6V 7Q3 880
Y8W 8G6 881
F7B 7Y6 882
X0V 0F5 883
K5K 355 884
L4B 4E3 886
U5T 9N6 887
09A 5G0 888
C8Z 554 888
Q7T 4K7 889
M0V 9G8 889
F9Y 5E6 890
R7U 3Q4 891
J1M 6B4 891
F2C 6A5 891
U5N 7L7 894
H5Z 2W0 894
E6Q 9P1 895
N6S 1P4 895
N5Q 8E5 895
J4M 4G3 895
F6W 0H3 896
08A 6Z5 897
09P 953 897
V8U 2T6 898
S0W 0P4 899
T1A 9W4 899
Time taken: 35.169 seconds, Fetched: 401 row(s)
hive>
```

- Here I have used this command to output the values of distributor_id and volume_out values from the table based on sorted values of volume_out.

Creation of Olympics Table in Hive and Loading of data

```
hive> CREATE table olympic (athlete STRING, age INT, country STRING, year STRING, closing STRING, sport STRING, gold INT, silver INT, bronze INT, total INT) row format delimited fields terminated by '\t' stored as textfile;
OK
Time taken: 0.266 seconds
hive> load data local inpath '/home/cloudera/ice4/olympic_data.csv' into table olympic;
Loading data to table default.olympic
Table default.olympic stats: [numFiles=1, totalSize=518669]
OK
Time taken: 0.496 seconds
hive>
```

- Firstly I have created olympics table with columns like athlete, age, country, closing, sport, gold, silver, bronze and total and then also loaded the olympics dataset into the table.

1) Using the dataset list the total number of medals won by each country in swimming.

```
cloudera@quickstart:~$
File Edit View Search Terminal Help
hive> select country,SUM(total) from olympic where sport = "Swimming" GROUP BY country;
Query ID = cloudera.20220210133535_e24f2122-f3a8-4b1d-bf78-b37a826dc7a8
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.reducers=<number>
Starting Job = job_1644517425553_0005, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1644517425553_0005/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1644517425553_0005
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-02-10 13:35:51,590 Stage-1 map = 0%, reduce = 0%
2022-02-10 13:36:00,489 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.14 sec
2022-02-10 13:36:10,203 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 4.12 sec
MapReduce Total cumulative CPU time: 4 seconds 120 msec
Ended Job = job_1644517425553_0005
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 4.12 sec HDFS Read: 528176 HDFS Write: 386 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 120 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

cloudera@quickstart:~$
File Edit View Search Terminal Help
Starting Job = job_1644517425553_0005, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1644517425553_0005/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1644517425553_0005
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-02-10 13:35:51,590 Stage-1 map = 0%, reduce = 0%
2022-02-10 13:36:00,489 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.14 sec
2022-02-10 13:36:10,203 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 4.12 sec
MapReduce Total cumulative CPU time: 4 seconds 120 msec
Ended Job = job_1644517425553_0005
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 4.12 sec HDFS Read: 528176 HDFS Write: 386 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 120 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: 31.429 seconds, Fetched: 34 row(s)
hive>
```

- Here I have used select command to select and output the values of country and total sum from the table based on sport swimming.

2) Display real life number of medals India won year wise.

```
hive>
>
> select year,SUM(total) from olympic where country = "India" GROUP BY year;
Query ID = cloadera_28228210134343_5b296d21-9b2b-44ed-8ae4-409c91b44c92
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_1644517425553_0007, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1644517425553_0007/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1644517425553_0007
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-02-10 13:44:03,932 Stage-1 map = 0%, reduce = 0%
2022-02-10 13:44:12,982 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.03 sec
2022-02-10 13:44:27,088 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 4.03 sec
MapReduce Total cumulative CPU time: 4 seconds 38 msec
Ended Job = job_1644517425553_0007
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 4.03 sec HDFS Read: 528221 HDFS Write: 28 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 38 msec
OK
2000 1
2004 1
2008 3
2012 6
Time taken: 36.123 seconds, Fetched: 4 row(s)
hive>
```

- Here I have used select clause to select and display the values of year and total sum of medals and 'where' is used to specify condition i.e the country is India.

3) Find the total number of medals each country won display the name along with total medals.

```
cloudera@quickstart:~$
File Edit View Search Terminal Help
hive> select country,SUM(total) from olympic GROUP BY country;
Query ID = cloadera_28228210134545_e072c07a-eff7-41da-8296-a6d6b7eca4ce
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_1644517425553_0008, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1644517425553_0008/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1644517425553_0008
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-02-10 13:46:10,966 Stage-1 map = 0%, reduce = 0%
2022-02-10 13:46:20,533 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.03 sec
2022-02-10 13:46:31,372 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 4.02 sec
MapReduce Total cumulative CPU time: 4 seconds 20 msec
Ended Job = job_1644517425553_0008
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 4.02 sec HDFS Read: 527165 HDFS Write: 1315 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 20 msec
OK
Afghanistan 2
Algeria 8
Argentina 141
Armenia 10
Australia 669
Austria 91
Azerbaijan 25
Bahamas 24
Bahrain 1
Barbados 1
Belarus 97
Belgium 18
Botswana 1
Brazil 221
Bulgaria 41
Cameroon 28
Canada 370
Chile 22
China 530
Chinese Taipei 28
Colombia 13
Costa Rica 2
Croatia 81
Cuba 188
Cyprus 1
Czech Republic 81
```



```

cloudera@quickstart:~$
File Edit View Search Terminal Help
Mexico 38
Moldova 5
Mongolia 10
Montenegro 14
Morocco 11
Mozambique 1
Netherlands 318
New Zealand 52
Nigeria 39
North Korea 21
Norway 192
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 388
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: 36.618 seconds, Fetched: 110 row(s)

```

- Here I have used select, sum to select and display the country names and total sum value and group by is used to order results based on country from the Olympics table.

4) Find the real life number of gold medals each country won.

```

cloudera@quickstart:~$
File Edit View Search Terminal Help
hive> select country,sum(gold) from olympic GROUP BY country;
Query ID = cloudera_28228218134848_027bf082-276a-4326-8836-209c62396283
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_1644517425553_0009, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1644517425553_0009/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1644517425553_0009
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-02-10 13:49:08,000 Stage-1 map = 0%, reduce = 0%
2022-02-10 13:49:17,686 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 1.61 sec
2022-02-10 13:49:27,638 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 3.62 sec
MapReduce Total cumulative CPU time: 3 seconds 628 msec
Ended Job = job_1644517425553_0009
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 3.62 sec HDFS Read: 527165 HDFS Write: 1276 SUCCESS
Total MapReduce CPU Time Spent: 3 seconds 628 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 28
Canada 168
Chile 3
China 234
Chinese Taipei 2
Colombia 2
Costa Rica 0
Croatia 35
Cuba 57
Cyprus 0
Czech Republic 14

```

```

cloudera@quickstart:~
File Edit View Search Terminal Help
Mongolia 2
Montenegro 0
Morocco 2
Mozambique 1
Netherlands 101
New Zealand 18
Nigeria 6
North Korea 6
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: 30.678 seconds, Fetched: 110 row(s)
hive>
>

```

- Here I have used select, sum to select and output the values of country names and their total sum of golds and group by is used to order result by country from the Olympics table.

5) Which country got medals for Shooting, year wise classification?

```

cloudera@quickstart:~
File Edit View Search Terminal Help
hive> SELECT country, year from olympic where sport="Shooting" and total>0 GROUP BY year, country;
Query ID = cloudera_26220218135454_48f2814c-abe0-4406-8d50-267a942848d9
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.reducers=<number>
Starting Job = job_1644517425553_0010, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1644517425553_0010/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1644517425553_0010
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-02-10 13:54:57,226 Stage-1 map = 0%, reduce = 0%
2022-02-10 13:55:07,326 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.21 sec
2022-02-10 13:55:17,415 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 4.34 sec
MapReduce Total cumulative CPU time: 4 seconds 340 msec
Ended Job = job_1644517425553_0010
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 4.34 sec HDFS Read: 528461 HDFS Write: 1271 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 340 msec
OK
Australia 2000
Azerbaijan 2000
Belarus 2000
Bulgaria 2000
China 2000
Czech Republic 2000
Denmark 2000
Finland 2000
France 2000
Great Britain 2000
Hungary 2000
Italy 2000
Kuwait 2000
Lithuania 2000
Moldova 2000
Norway 2000
Poland 2000
Romania 2000
Russia 2000
Serbia and Montenegro 2000
Slovenia 2000
South Korea 2000
Sweden 2000
Switzerland 2000
Ukraine 2000
United States 2000

```

```

cloudera@quickstart:~
File Edit View Search Terminal Help
United Arab Emirates 2004
United States 2004
Australia 2008
China 2008
Croatia 2008
Cuba 2008
Czech Republic 2008
Finland 2008
France 2008
Georgia 2008
Germany 2008
India 2008
Italy 2008
Mongolia 2008
Norway 2008
Russia 2008
Slovakia 2008
Slovenia 2008
South Korea 2008
Ukraine 2008
United States 2008
Belarus 2012
Belgium 2012
China 2012
Croatia 2012
Cuba 2012
Czech Republic 2012
Denmark 2012
France 2012
Great Britain 2012
India 2012
Italy 2012
Kuwait 2012
Poland 2012
Qatar 2012
Romania 2012
Russia 2012
Serbia 2012
Slovakia 2012
Slovenia 2012
South Korea 2012
Sweden 2012
Ukraine 2012
United States 2012
Time taken: 33.277 seconds, Fetched: 90 row(s)

```

- Here I have used above command to select and display the country name, medals and year in shooting sport and group by is used to order result by year and country.

Tasks

1) Create 3 tables called movies, ratings and users. Load the data into tables.

```

hive> create table movies(movieID INT, title STRING, genres STRING) row format delimited fields terminated by ',' stored as textfile;
OK
Time taken: 0.314 seconds
hive> load data local inpath '/home/cloudera/ice4/movies.csv' into table movies;
Loading data to table default.movies
Table default.movies stats: [numFiles=1, totalSize=494431]
OK
Time taken: 0.618 seconds
hive>

```

- I have created movies table with columns like movieId, title and genres and then loaded the dataset into the table.

```

hive> create table rating(userID INT, movieID INT, rating STRING, timestamp STRING) row format delimited stored as textfile;
OK
Time taken: 0.125 seconds
hive> load data local inpath '/home/cloudera/ice4/ratings.csv' into table rating;
Loading data to table default.rating
Table default.rating stats: [numFiles=1, totalSize=2483723]
OK
Time taken: 0.713 seconds
hive>

```

- I have created rating table with columns being userId, movieId, rating and timestamp and then loaded the dataset into the table.

```
hive> create table users(userid int,gender string,id int,ratingsgiven int,zip string) row format delimited fields terminated by ',' stored as textfile;
OK
Time taken: 12.887 seconds
hive> load data local inpath '/home/cloudera/ice4/users.txt' into table users;
Loading data to table default.users
Table default.users stats: [numFiles=1, totalSize=0]
OK
Time taken: 0.968 seconds
hive>
```

- I have created users table with columns being userId, gender, id, ratingsgiven and zip and then loaded the dataset into the table.

2) List all movies with genre of movie is “Action” and “Drama”.

```
hive> select * from movies where genre like '%Drama%' and genre like '%Action%';
OK
20      Money Train (1995)      Action|Comedy|Crime|Drama|Thriller
42      Dead Presidents (1995) Action|Crime|Drama
86      White Squall (1996)     Action|Adventure|Drama
110     Braveheart (1995)      Action|Drama|War
145     Bad Boys (1995) Action|Comedy|Crime|Drama|Thriller
151     Rob Roy (1995)  Action|Drama|Romance|War
168     First Knight (1995)  Action|Drama|Romance
198     Strange Days (1995)  Action|Crime|Drama|Mystery|Sci-Fi|Thriller
292     Outbreak (1995) Action|Drama|Sci-Fi|Thriller
293     Léon: The Professional (a.k.a. The Professional) (Léon) (1994) Action|Crime|Drama|Thriller
349     Clear and Present Danger (1994) Action|Crime|Drama|Thriller
384     Bad Company (1995)   Action|Crime|Drama
390     Faster Pussycat! Kill! Kill! (1965)   Action|Crime|Drama
493     Menace II Society (1993)   Action|Crime|Drama
504     No Escape (1994)   Action|Drama|Sci-Fi
517     Rising Sun (1993)   Action|Drama|Mystery
519     RoboCop 3 (1993)   Action|Crime|Drama|Sci-Fi|Thriller
522     Romper Stomper (1992) Action|Drama
553     Tombstone (1993)   Action|Drama|Western
647     Courage Under Fire (1996) Action|Crime|Drama|War
786     Eraser (1996)   Action|Drama|Thriller
798     Daylight (1996) Action|Adventure|Drama|Thriller
875     Nothing to Lose (1994) Action|Crime|Drama
996     Last Man Standing (1996)   Action|Crime|Drama|Thriller
1100    Days of Thunder (1990) Action|Drama|Romance
1112    Palookaville (1996)   Action|Comedy|Drama
1208    Apocalypse Now (1979) Action|Drama|War
1209    Once Upon a Time in the West (C'era una volta il West) (1968) Action|Drama|Western
1224    Henry V (1989)   Action|Drama|Romance|War
1264    Diva (1981)   Action|Drama|Mystery|Romance|Thriller
```

```

138632 Tokyo Tribe (2014) Action|Crime|Drama|Sci-Fi
139130 Afro Samurai (2007) Action|Adventure|Animation|Drama|Fantasy
139642 Southpaw (2015) Action|Drama
142420 High Rise (2015) Action|Drama|Sci-Fi
144352 Unforgiven (2013) Action|Crime|Drama
146730 Lost in the Sun (2015) Action|Drama|Thriller
149612 Swelter (2014) Action|Drama|Thriller
150548 Sherlock: The Abominable Bride (2016) Action|Crime|Drama|Mystery|Thriller
156607 The Huntsman Winter's War (2016) Action|Adventure|Drama|Fantasy
157407 I Am Wrath (2016) Action|Crime|Drama|Thriller
158874 Karate Bullfighter (1975) Action|Drama
160527 Sympathy for the Underdog (1971) Action|Crime|Drama
160730 The Adderall Diaries (2015) Action|Drama|Thriller
160836 Hazard (2005) Action|Drama|Thriller
161032 The Grandmother (1970) Action|Drama
161354 Batman: The Killing Joke (2016) Action|Animation|Crime|Drama
161594 Kingsglaive: Final Fantasy XV (2016) Action|Adventure|Animation|Drama|Fantasy|Sci-Fi
165347 Jack Reacher: Never Go Back (2016) Action|Crime|Drama|Mystery|Thriller
168456 Mercury Plains (2016) Action|Adventure|Drama
168612 Ghost in the Shell (2017) Action|Drama|Sci-Fi
169992 Free Fire (2017) Action|Crime|Drama
170399 CHiPS (2017) Action|Comedy|Drama
170705 Band of Brothers (2001) Action|Drama|War
170875 The Fate of the Furious (2017) Action|Crime|Drama|Thriller
171765 Okja (2017) Action|Adventure|Drama|Sci-Fi
173145 War for the Planet of the Apes (2017) Action|Adventure|Drama|Sci-Fi
174055 Dunkirk (2017) Action|Drama|Thriller|War
175585 Shot Caller (2017) Action|Crime|Drama|Thriller
184931 Death Wish (2018) Action|Crime|Drama|Thriller
187031 Jurassic World: Fallen Kingdom (2018) Action|Adventure|Drama|Sci-Fi|Thriller
Time taken: 38.665 seconds, Fetched: 427 row(s)
hive> █

```

- Here I have used select in this command to select and display the names of all movies and where is use to specify condition i.e genre is 'Action' and 'Drama'.

3) List movie ids of all movies with rating equal to 5.

```

hive> select M.id,MR.rating from movies M join ratings MR on (M.id=MR.id) where MR.rating = 5.0;
Query ID = cloudera_20220213212727_b5058f98-afee-45db-aab9-e6037126f4a4
Total jobs = 1
Execution log at: /tmp/cloudera/cloudera_20220213212727_b5058f98-afee-45db-aab9-e6037126f4a4.log
2022-02-13 09:28:05 Starting to launch local task to process map join; maximum memory = 1013645312
2022-02-13 09:28:31 Dump the side-table for tag: 0 with group count: 9742 into file: file:/tmp/cloudera/80315358-fcce-4f6d-bf60-2f72af78ec65/hive_2022-02-13_21-27-0
6_569_6829315296793445164-1/-local-10003/HashTable-Stage-3/MapJoin-mapfile10--.hashtable
2022-02-13 09:28:32 Uploaded 1 File to: file:/tmp/cloudera/80315358-fcce-4f6d-bf60-2f72af78ec65/hive_2022-02-13_21-27-06_569_6829315296793445164-1/-local-10003/Hash
Table-Stage-3/MapJoin-mapfile10--.hashtable (198527 bytes)
2022-02-13 09:28:32 End of local task; Time Taken: 26.859 sec.
Execution completed successfully
MapredLocal task succeeded
Launching Job 1 out of 1
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1644803028425_0002, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1644803028425_0002/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1644803028425_0002
Hadoop job information for Stage-3: number of mappers: 1; number of reducers: 0
2022-02-13 21:29:20,845 Stage-3 map = 0%, reduce = 0%
█

```

```

78499
78836
81834
86142
86898
88129
89745
92420
93838
93840
96610
96832
100906
102125
103341
106920
107406
107771
109968
110501
112175
112183
112290
115149
115727
121231
122882
122920
138632
156371
158238
164179
168248
168250
168252
Time taken: 40.026 seconds, Fetched: 13211 row(s)

```

- Here I have used select clause to select movie_id and movie_rating from movies and ratings table and display all movie id's whose rating is equal to 5.
- Here I have used join operation as we are retrieving data from two tables named 'movies' and 'rating' based on a common column named movie_id and to return movie_id's for the rating of 5.

4) Find top 11 average rated "Action" movies with descending order of rating.

```

hive> select M.title,avg(MR.rating) as a from movies M join ratings MR on (M.id=MR.id) where M.genre like '%Action%' group by M.title order by a desc limit 11;
Query ID = cloudera_20220213213737_237a9bad-8379-42e7-9cd3-09a674f420eb
Total jobs = 2
Execution log at: /tmp/cloudera/cloudera_20220213213737_237a9bad-8379-42e7-9cd3-09a674f420eb.log
2022-02-13 09:37:55 Starting to launch local task to process map join; maximum memory = 1013645312
2022-02-13 09:38:05 Dump the side-table for tag: 0 with group count: 1499 into file: file:/tmp/cloudera/80315358-fcce-4f6d-bf60-2f72af78ec65/hive_2022-02-13_21-37-23_542_8526554827072050210-1/-local-10005/HashTable-Stage-2/MapJoin-mapfile20-..hashtable
2022-02-13 09:38:05 Uploaded 1 File to: file:/tmp/cloudera/80315358-fcce-4f6d-bf60-2f72af78ec65/hive_2022-02-13_21-37-23_542_8526554827072050210-1/-local-10005/HashTable-Stage-2/MapJoin-mapfile20-..hashtable (73685 bytes)
2022-02-13 09:38:05 End of local task; Time Taken: 10.585 sec.
Execution completed successfully
MapredLocal task succeeded
Launching Job 1 out of 2
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_1644803028425_0003, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1644803028425_0003/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1644803028425_0003

```

```

OK
Reform School Girls (1986)      5.0
Max Manus (2008)                5.0
Justice League: Doom (2012)     5.0
Branded to Kill (Koroshi no rakuin) (1967)  5.0
Alien Contamination (1980)      5.0
Crippled Avengers (Can que) (Return of the 5 Deadly Venoms) (1981)  5.0
Supercop 2 (Project S) (Chao ji ji hua) (1993)  5.0
On the Other Side of the Tracks (De l'autre côté du périph) (2012)  5.0
Maniac Cop 2 (1990)             5.0
Sonatine (Sonachine) (1993)     5.0
Galaxy of Terror (Quest) (1981)  5.0
Time taken: 290.01 seconds, Fetched: 11 row(s)
hive>

```

- Here I have used select clause to select movie title and the average of rating given for movies which are of genre 'Action' and group by is used to group them by movie title on a limit of 11 in descending order. Here we used desc to get values in descending order.
- Here I have used join operation as we are retrieving data from two tables named 'movies' and 'rating' based on a common column named movie_id and to return movie_rating for that appropriate movie_id.