# ASSIGNMENT- 4

## Hive Commands Use Case – Petrol


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

## Creation of Petrol Table in Hive and Loading of data


- 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 distributer_name, SUM(vol_OUT) FROM petrol GROUP BY distributer_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
Distributer 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 distributer_name and total amount of petrol sold from the table and display output grouped by distributer_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 distributer_id, vol_OUT FROM petrol order by vol_OUT desc limit 10;
Query ID = cloudera_20220210131414_7585b3fc-ed1d-45b3-a0da-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
T1A 9W4 899
S8W 0P4 899
V8U 2T6 898
08A 6Z5 897
09P 9S3 897
F6W 6H3 896
N5Q 8E5 895
M6S 1P4 895
E6O 9P1 895
J4M 4G3 895
Time taken: 34.609 seconds, Fetched: 10 row(s)
hive>
```

- Here I have used select command to select and display distributer_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 distributer_id, vol_OUT FROM petrol order by vol_OUT limit 10;
Query ID = cloudera_20220210132121_fb3dabd2-a582-4cff-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.reduces=<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 13:21:35,862 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 13:21:54,409 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
O5D 2R6 610
W0M 8R7 612
O0D 0L1 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 distributer_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 distributer_id,year,(vol_IN-vol_OUT) as difference from petrol where (vol_IN-vol_OUT)>400;
OK
S1J 8B8 1627    409
D2O 2X9 1644    401
E6U 0I2 1646    403
H5N 9W3 1662    428
H7W 3Q3 1670    406
M1J 2H6 1693    455
W0M 8R7 1697    435
V0Z 0F6 1708    422
K3Z 9G3 1726    443
F9H 4S7 1727    409
V8S 4P6 1732    464
O4W 8F6 1741    409
H8W 3U0 1745    417
R6N 9L2 1746    419
J7Z 9V4 1748    446
N0D 4G5 1749    407
R3W 2E3 1767    480
N3N 0W9 1769    440
M2B 0Q2 1779    401
C9B 3Q8 1782    457
A0X 9H9 1784    423
E3O 3Y7 1794    421
A7Z 3L9 1795    474
N2I 6C8 1823    431
C1N 6M7 1832    402
Y5F 0S2 1833    434
C0G 8G1 1843    460
M4V 0E1 1847    423
G7K 1D3 1852    424
P1B 7R8 1930    406
J1C 3G1 1934    435
I2N 7S1 1935    478
C8L 5T7 1946    406
I6D 1H5 1955    403
C4L 9Y5 1995    428
T4L 8D0 2000    452
J1O 5K1 2019    408
Time taken: 0.255 seconds, Fetched: 37 row(s)
hive>
```

- I have used this command to select and display the distributer_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”.

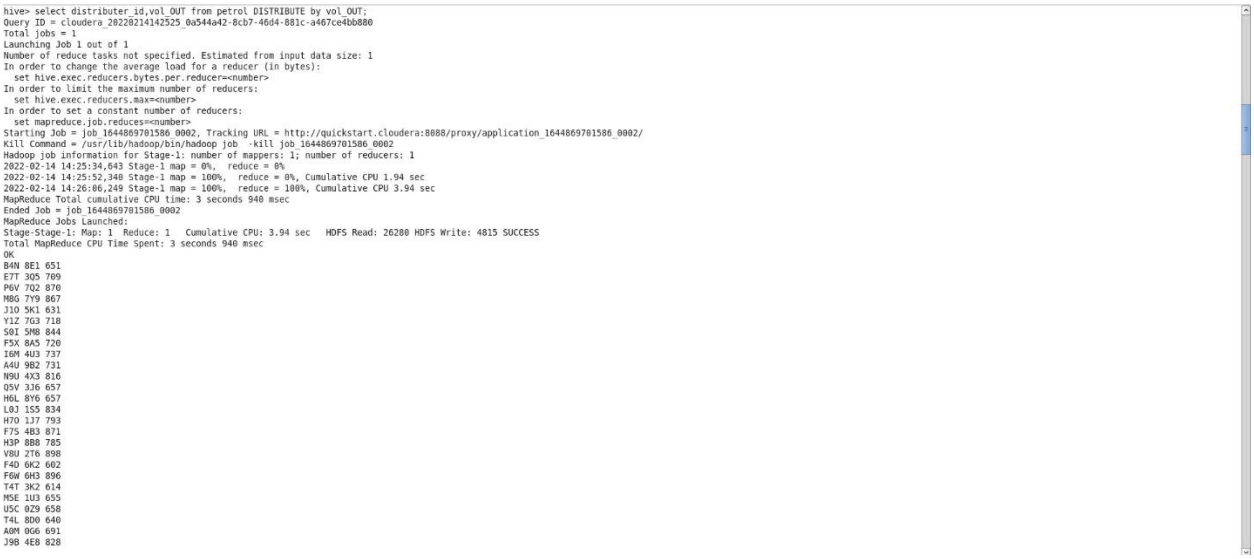
```
hive> select distributer_id,vol_OUT from petrol CLUSTER by vol_OUT;
Query ID = cloudera_20220214141818_15efb102-a6c2-4c06-9d5e-277f23786644
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (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.reduces=<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
F4O 6K2 602
H7M 4M4 603
G9F 6U7 607
R3W 2E3 608
H4P 6A9 610
O5D 2M6 610
W0M 8R7 612
O0D 0L1 612
V0Z 0F6 612
L9H 1K6 613
E6U 0I2 613
K9N 3H4 614
H8W 3U0 614
T4T 3K2 614
I3U 6H8 615
M1J 2H6 615
S2Z 0C3 616
E5T 2A9 617
C7W 9P0 617
N3N 0W9 617
P3Q 0Q9 618
K3E 9F5 619
B0B 0W6 619
P5C 4O8 619
Y8I 6V4 619
```



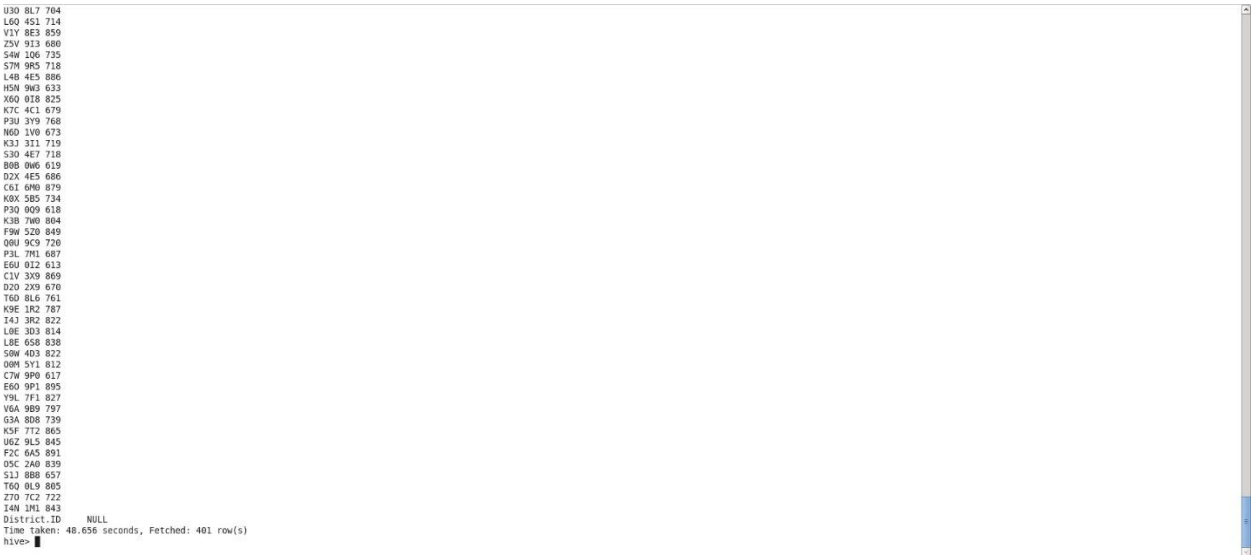
```
D9J 0N0 868
C1V 3X9 869
A5Q 3H2 870
P6V 7Q2 870
T9V 0E2 870
P1A 3W8 870
F7S 4B3 871
U2M 9W3 872
G4P 5B4 872
H5V 0V1 873
J3A 6G9 873
E7Q 2O3 873
I6U 1K6 874
O6Y 2C9 877
W4T 6K2 877
A9P 7L9 878
T1A 9O4 878
C6I 6M0 879
N2O 9Y7 879
M9N 1E7 879
A0V 7Q3 880
Y8W 8G6 881
F7B 7Y6 882
X8V 0F5 883
K5K 3S5 884
L4B 4E5 886
U5T 9M6 887
O9A 5G0 888
C0Z 5S4 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
E6O 9P1 895
M6S 1P4 895
N5Q 8E5 895
J4M 4G3 895
F6W 6H3 896
O8A 6Z5 897
O9P 9S3 897
V8U 2T6 898
S8W 8P4 899
T1A 9W4 899
Time taken: 87.033 seconds, Fetched: 401 row(s)
hive>
```

- Here I have used this command to select and output the values of distributer_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 distributer_id,vol_OUT from petrol DISTRIBUTE by vol_OUT;
Query ID = cloudera_20220214142525_0a544a42-8cb7-46d4-881c-a467ce4bb880
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_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 8E1 651
E7T 3Q5 709
P6V 7Q2 870
M8G 7Y9 867
J1O 5K1 631
Y1Z 7G3 718
S0I 5M8 844
F5X 8A5 720
I6M 4U3 737
A4U 9B2 731
N9U 4X3 816
Q5V 3J6 657
H6L 8Y6 657
L0J 1S5 834
H7O 1J7 793
F7S 4B3 871
H3P 8B8 785
V8U 2T6 898
F4D 6K2 602
F6W 6H3 896
T4T 3K2 614
M5E 1U3 655
U5C 0Z9 658
T4L 8D0 640
A0M 0G6 691
J9B 4E8 828
```



```
U3O 8L7 704
L6Q 4S1 714
Y1Y 8E3 859
Z5V 9I3 680
S4W 1Q6 735
S7M 9R5 718
L4B 4E5 886
H5N 9W3 633
X6Q 0I8 825
K7C 4C1 679
P3U 3Y9 768
N6O 1V0 673
K3J 3I1 719
S3O 4E7 718
B0B 0W6 619
D2X 4E5 686
C6I 6M0 879
K0X 5B5 734
P3Q 0Q9 618
K3B 7W0 804
F9W 5Z0 849
Q0U 9C9 720
P3L 7M1 687
E6U 0I2 613
C1V 3X9 869
D2O 2X9 670
T6D 8L6 761
K9E 1R2 787
I4J 3R2 822
L0E 3D3 814
L8E 6S8 838
S0W 4D3 822
O0M 5Y1 812
C7W 9P0 617
E6O 9P1 895
Y9L 7F1 827
V6A 9B9 797
G3A 8D8 739
K5F 7T2 865
U6Z 9L5 845
F2C 6A5 891
O5C 2A0 839
S1J 8B8 657
T6Q 0L9 805
Z7O 7C2 722
I4N 1M1 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 distributer_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 distributer_id,vol_OUT from petrol SORT by vol_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_1644869701586_0003, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1644869701586_0003/
Kill Command = /usr/lib/hadoop/bin/hadoop job  -kill job_1644869701586_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 14:29:28,831 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_1644869701586_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
F4D 6K2 602
H7M 4M4 603
G9F 6U7 607
R3W 2E3 608
H4P 6A9 610
O5D 2R6 610
W0M 8R7 612
O0D 0L1 612
V0Z 0F6 612
L9H 1K6 613
E6U 0I2 613
K9N 3H4 614
H8W 3U0 614
T4T 3K2 614
I3U 6H8 615
M1J 2H6 615
S2Z 0C3 616
E5T 2A9 617
C7W 9P0 617
N3N 0W9 617
P3Q 0Q9 618
K3E 9F5 619
B0B 0W6 619
P5C 4O8 619
Y8I 6V4 619
```

```
D9J 0N0 868
C1V 3X9 869
A5Q 3H2 870
P6V 7Q2 870
T9V 0E2 870
P1A 3W8 870
F7S 4B3 871
U2M 9W3 872
G4P 5B4 872
H5V 0V1 873
J3A 6G9 873
E7Q 2O3 873
I6U 1K6 874
O6Y 2C9 877
W4T 6K2 877
A9P 7L9 878
T1A 9O4 878
C6I 6M0 879
N2O 9Y7 879
M9N 1E7 879
A0V 7Q3 880
Y8W 8G6 881
F7B 7Y6 882
X0V 0F5 883
K5K 3S5 884
L4B 4E5 886
U5T 9M6 887
O9A 5G0 888
C0Z 5S4 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
E6O 9P1 895
M6S 1P4 895
N5Q 8E5 895
J4M 4G3 895
F6W 6H3 896
O8A 6Z5 897
O9P 9S3 897
V8U 2T6 898
S8W 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 distributer_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 (athelete 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 athelete, age, country, closing, sport, gold, silver, bronze and total and then also loaded the olypics dataset into the table.

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

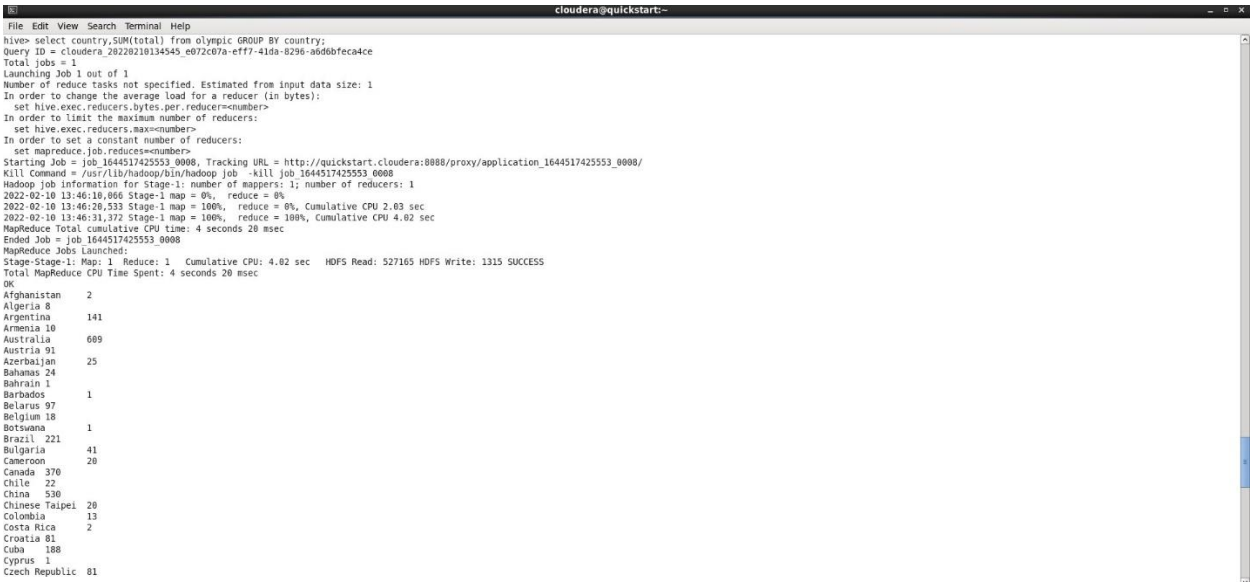
- Here I have used select command to select and output the values of country and total sum from the table based on sport swimming.

**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 = cloudera_20220210134343_5b290d21-9020-44ed-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 13:44:03,032 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 13:44:27,800 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 4.03 sec
MapReduce Total cumulative CPU time: 4 seconds 30 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 30 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.

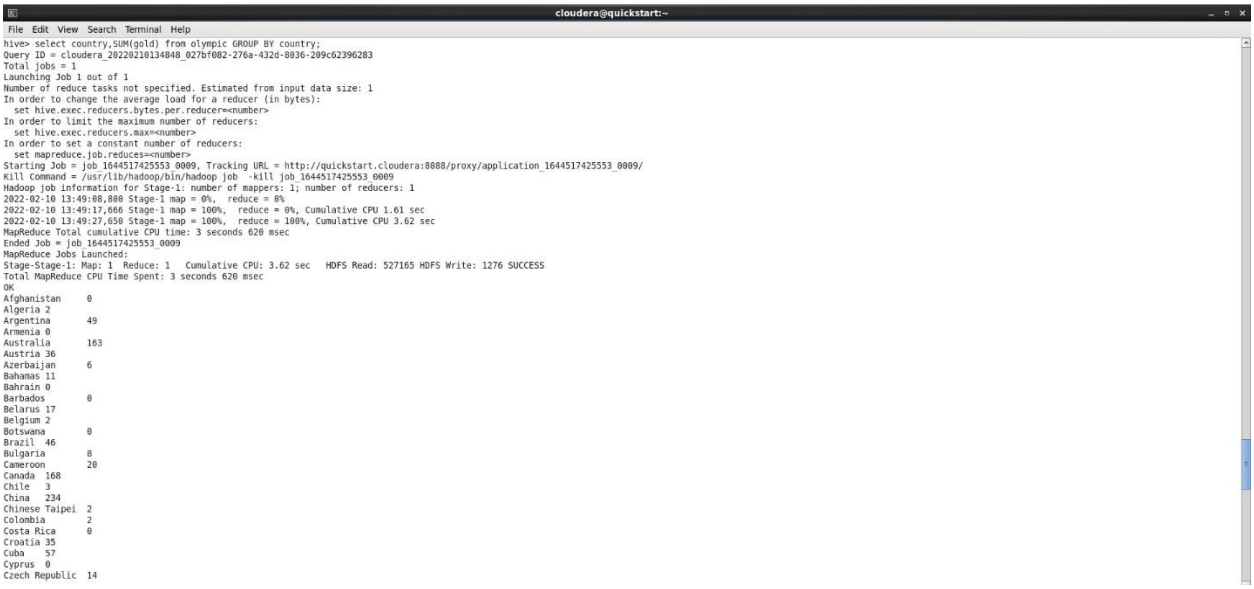
**3) Find the total number of medals each country won display the name along with total medals.**

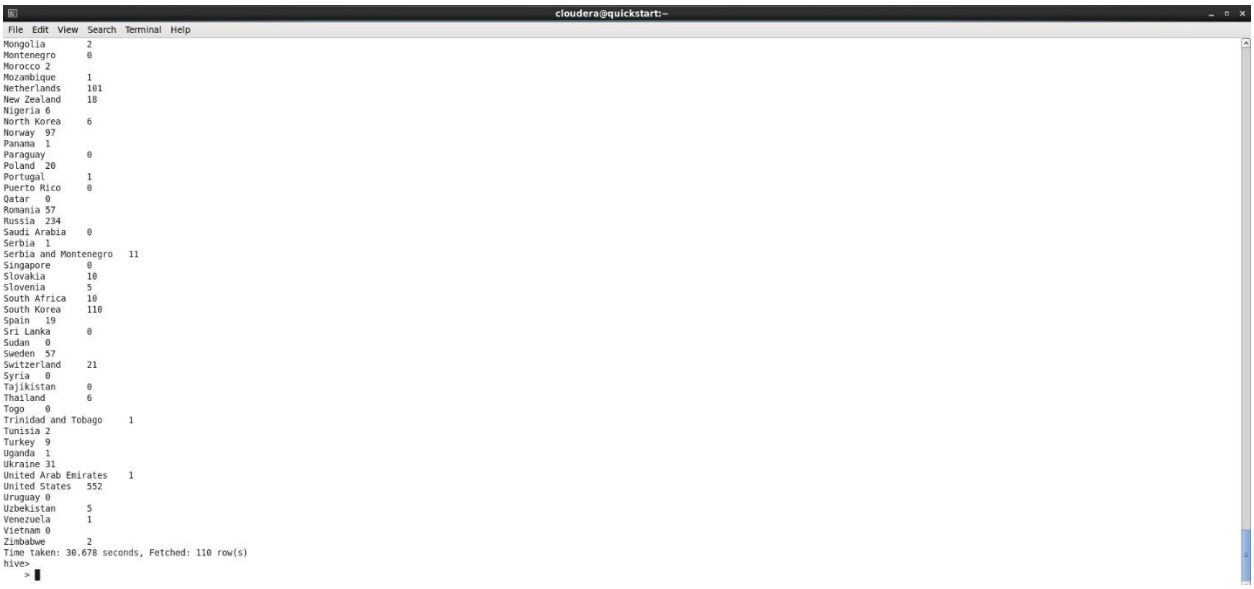


```
cloudera@quickstart:~
File  Edit  View  Search  Terminal  Help
hive> select country,SUM(total) from olympic GROUP BY country;
Query ID = cloudera_20220210134545_e072c07a-eff7-41da-8296-a6d6bfeca4ce
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input 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 13:46:10,066 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       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
```


- 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.

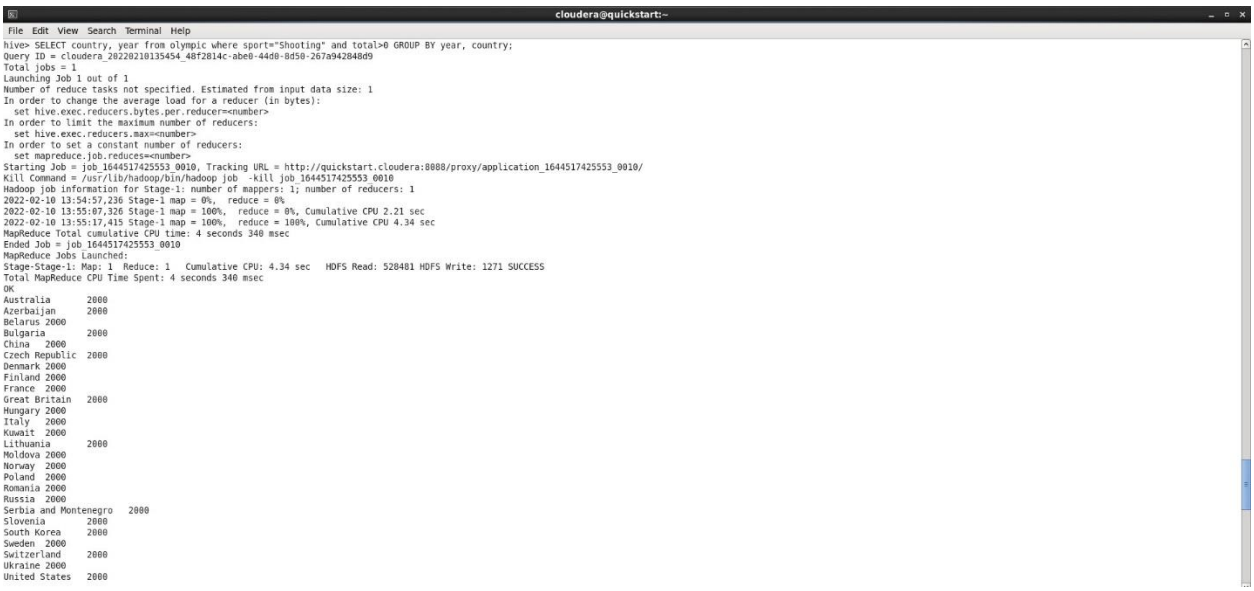
## 4) Find the real life number of gold medals each country won.

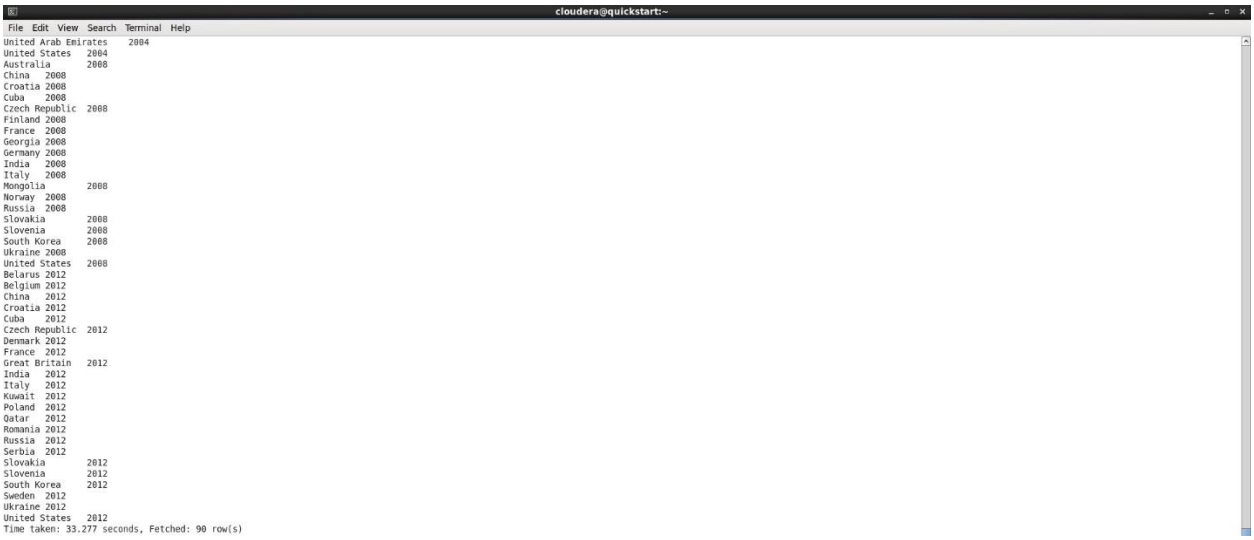





- Here I have used select, sum to select and output the values of country names and their total sum of golds and group by is used to order result by country from the Olympics table.

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



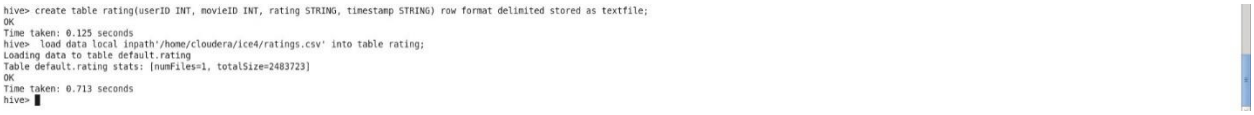



- Here I have used above command to select and display the country name, medals and year in shooting sport and group by is used to order result by year and country.

## Tasks

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


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




- 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: 12.807 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.