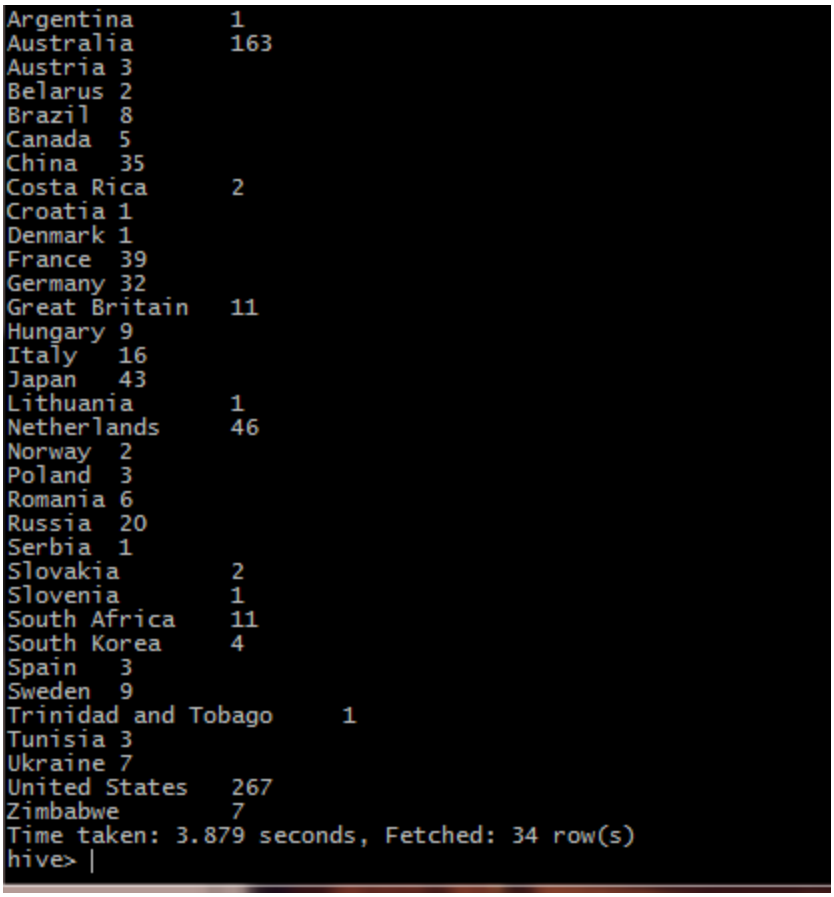
Task 1

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

Program:

```
hive> select country, sum(total) from olympics where sport='Swimming' group by c
ountry order by country;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the futu
re versions. Consider using a different execution engine (i.e. spark, tez) or us
ing Hive 1.X releases.
Query ID = acadgild_20181017220135_707c1d8c-2d2b-4033-bf2e-e371771f30f8
Total jobs = 2
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>
Job running in-process (local Hadoop)
2018-10-17 22:01:38,006 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local2133764090_0006
Launching Job 2 out of 2
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>
Job running in-process (local Hadoop)
2018-10-17 22:01:39,699 Stage-2 map = 100%,  reduce = 100%
Ended Job = job_local1818741112_0007
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 11410718 HDFS Write: 1037338 SUCCESS
Stage-Stage-2:  HDFS Read: 11410718 HDFS Write: 1037338 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
```

Results:



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

Program:

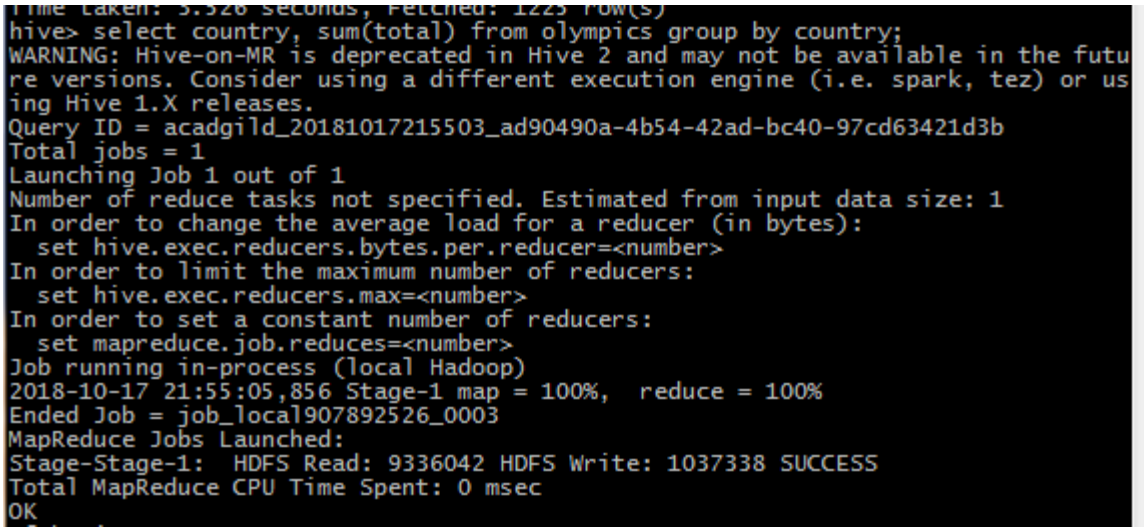
```
hive>
    >
    >
    >
    > select myyear, sum(total) from olympics where country = 'India'
    > group by myyear order by myyear;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the futu
re versions. Consider using a different execution engine (i.e. spark, tez) or us
ing Hive 1.X releases.
Query ID = acadgild_20181017215759_a37a75ab-3efa-498b-ba58-36707713decb
Total jobs = 2
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>
Job running in-process (local Hadoop)
2018-10-17 21:58:02,157 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local1058517218_0004
Launching Job 2 out of 2
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>
Job running in-process (local Hadoop)
2018-10-17 21:58:03,869 Stage-2 map = 100%,  reduce = 100%
Ended Job = job_local1961451449_0005
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 10373380 HDFS Write: 1037338 SUCCESS
Stage-Stage-2:  HDFS Read: 10373380 HDFS Write: 1037338 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
```

Result:

```
OK
2000    1
2004    1
2008    3
2012    6
Time taken: 4.261 seconds, Fetched: 4 row(s)
hive>
```

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

Program:


Time taken: 3.526 seconds, Fetched: 1223 row(s)
hive> select country, sum(total) from olympics group by country;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the futu
re versions. Consider using a different execution engine (i.e. spark, tez) or us
ing Hive 1.X releases.
Query ID = acadgild_20181017215503_ad90490a-4b54-42ad-bc40-97cd63421d3b
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>
Job running in-process (local Hadoop)
2018-10-17 21:55:05,856 Stage-1 map = 100%, reduce = 100%
Ended Job = job_local907892526_0003
MapReduce Jobs Launched:
Stage-Stage-1: HDFS Read: 9336042 HDFS Write: 1037338 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK

Result:

```
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
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
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
```

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

Program:

```
hive> select country, sum(gold) from olympics group by country;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider usin
g a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
Query ID = acadgild_20181017220454_63851d89-77bd-4f55-8152-8196720ee39e
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>
Job running in-process (local Hadoop)
2018-10-17 22:04:56,375 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local768621700_0009
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 14522732 HDFS Write: 1037338 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
```

Results:

```
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
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
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
```

Task 2

Write a hive UDF that implements functionality of string concat_ws(string SEP, array<string>).
This UDF will accept two arguments, one string and one array of string.
It will return a single string where all the elements of the array are separated by the SEP.

Task 3

Link: https://acadgild.com/blog/transactions-in-hive/
Refer the above given link for transactions in Hive and implement the operations given in the blog using your own sample data set and send us the screenshot.