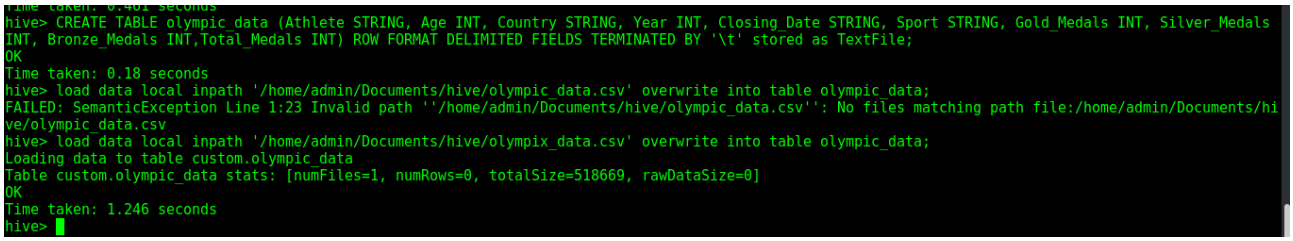
Advance Hive

Task 1:

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

Solution:

CREATE TABLE olympic_data (Athlete 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' stored as TextFile;
load data local inpath '/home/admin/Documents/hive/olympix_data.csv' overwrite into table olympic_data;



hive> select country,sum(total_medals) from olympic_data where sport = 'Swimming' Group by country;

```
hive> select country,sum(total_medals) from olympic_data where sport = 'Swimming' Group by country;
Query ID = root_20181022150451_2b2ed32b-7e15-4930-ae84-647d13387099
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1539070359797_0013)

--------------------------------------------------------------------------------
        VERTICES      STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 ..........   SUCCEEDED      1          1        0        0       0       0
Reducer 2 ......   SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 5.11 s
--------------------------------------------------------------------------------
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
```

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

Solution:

hive> select year,sum(total_medals) from olympic_data where country='India' group by year;

```
hive> select year,sum(total_medals)  from olympic_data where country='India' group by year;
Query ID = root_20181022151106_b2eb0512-48ed-4da5-83c3-954a9f916ca6
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1539070359797_0013)

--------------------------------------------------------------------------------
        VERTICES      STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 ..........   SUCCEEDED      1          1        0        0       0       0
Reducer 2 ......   SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 4.61 s
--------------------------------------------------------------------------------
OK
2000    1
2004    1
2008    3
2012    6
Time taken: 5.262 seconds, Fetched: 4 row(s)
hive> \
```

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

Solution:

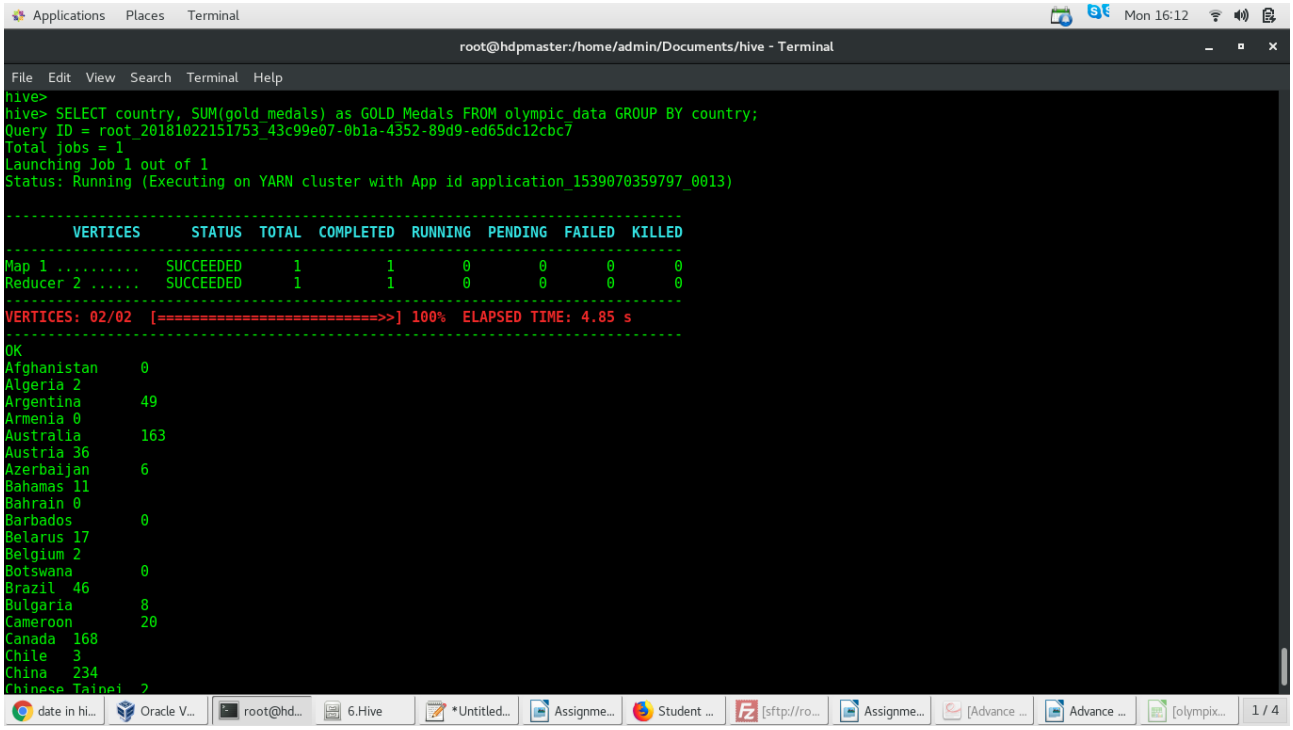
hive> select country,sum(total_medals) from olympic_data group by country;

```
hive> select country,sum(total_medals) from olympic_data group by country;
Query ID = root_20181022151350_f5202efb-6d7d-44f2-8293-a52025c94aeb
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1539070359797_0013)

--------------------------------------------------------------------------------
        VERTICES      STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 ..........   SUCCEEDED      1          1        0        0       0       0
Reducer 2 ......   SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 5.45 s
--------------------------------------------------------------------------------
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
```

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

Solution:

hive> SELECT country, SUM(gold_medals) as GOLD_Medals FROM olympic_data GROUP BY country;



Task 2:

Creae UDF Function which concatenate arguments that are passed.

```
hive> CREATE TABLE company(rank int, company_name string,website string, protocal string)
    > ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t';
OK
Time taken: 0.194 seconds
hive> load data local inpath 'hiveUdf.txt' overwrite into table company;
Loading data to table custom.company
Table custom.company stats: [numFiles=1, numRows=0, totalSize=520, rawDataSize=0]
OK
Time taken: 1.169 seconds
hive> select * from company limit 5;
OK
1       Sofia   Browsedrive     vk.com
2       Helaina Babblestorm     blogs.com
3       Worth   Mycat   usgs.gov
4       Glen    DabZ    xrea.com
5       Natalee Yadel   rakuten.co.jp
Time taken: 0.068 seconds, Fetched: 5 row(s)
hive>
```

CREATE TABLE company(rank int, company_name string,website string, protocal string)
    > ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t';

load data local inpath 'hiveUdf.txt' overwrite into table company;

select * from company limit 5;

o/p:

OK
1 Sofia Browsedrive vk.com
2 Helaina Babblestorm blogs.com
3 Worth Mycat usgs.gov
4 Glen DabZ xrea.com
5 Natalee Yadel rakuten.co.jp

adding JAR created from the JAVA class which is defining the UDF

add jar HiveUDFExample-0.0.1-SNAPSHOT.jar;

Create Temporary function:

hive> CREATE TEMPORARY FUNCTION hiveConcatws As 'Acadgild.HiveUDFExample.HiveConcatws';

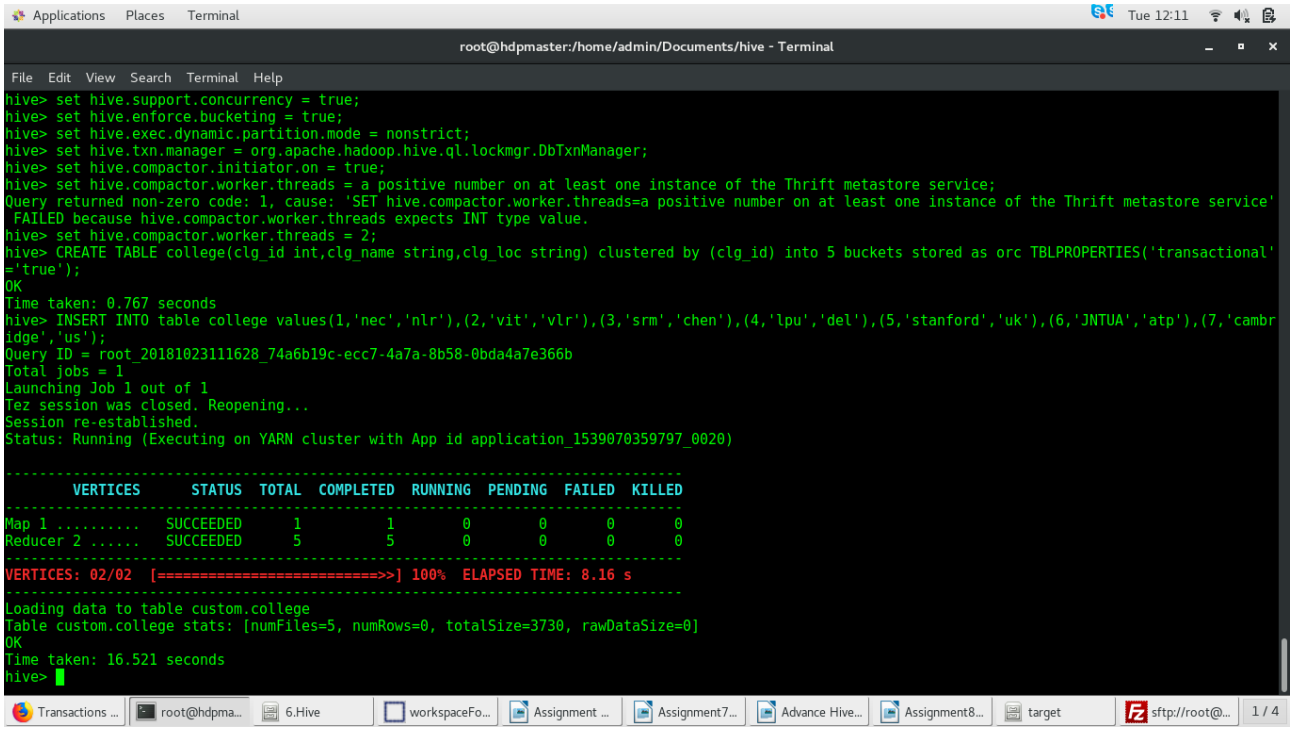
```
hive> SELECT hiveConcatws(website,'.',protocal) from company limit 5;
OK
Browsedrivevk.com
Babblestormblogs.com
Mycatusgs.gov
DabZxrea.com
Yadelrakuten.co.jp
Time taken: 0.09 seconds, Fetched: 5 row(s)
hive>
```

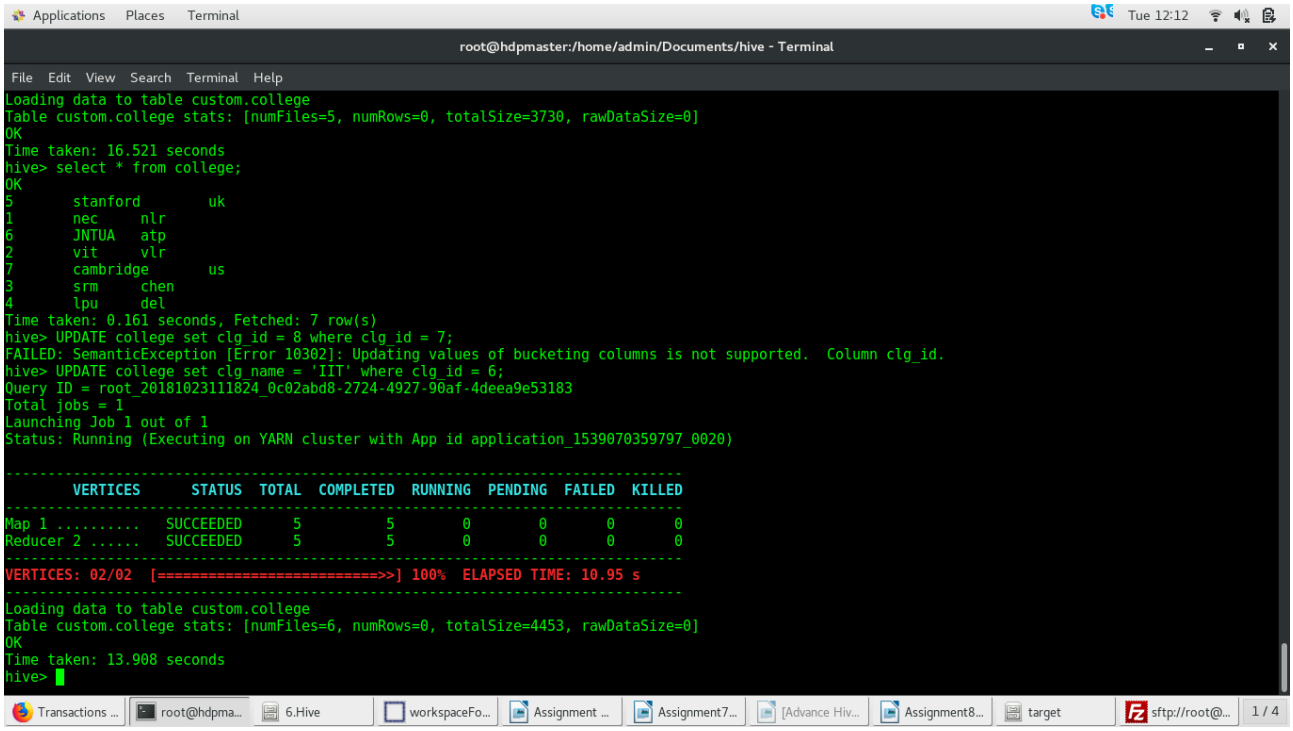
JAVA Class:

```
package Acadgild.HiveUDFExample;



import org.apache.hadoop.hive.ql.exec.Description;
import org.apache.hadoop.hive.ql.exec.UDF;

public class HiveConcatws extends UDF{

        @Description(name = "HiveConcatws", value = "_FUNC_(string SEP, array<string>) -
RETURN_TYPE(STRING)\n" + "Description: Concatenate two strings, separated by the
seperator",
                        extended = "Example:\n"
                                        + "  > SELECT HiveConcatws (website,'.',protocal) FROM
src;\n"
                                        + "www.walmart.com")



                public String evaluate(String param1, String[] param2)
```

```
        {
                String Output = "";
                if(param1==null && param2==null)
                {
                        return null;
                }
                for(int i = 0; i < param2.length; i++)
                {
                        Output+= param2[i];
                }
                return(param1.concat(Output));
        }
}
```

Task 3:





delete:

```
hive> select * from college;
OK
5       stanford        uk
1       nec     nlr
6       IIT     atp
2       vit     vlr
7       cambridge       us
3       srm     chen
4       lpu     del
Time taken: 0.151 seconds, Fetched: 7 row(s)
hive> delete from college where clg_id=5;
Query ID = root_20181023111957_bb7c4fc2-1bb2-4366-a555-7c80dce6ccfe
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1539070359797_0020)

--------------------------------------------------------------------------------
        VERTICES      STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
hive> delete from college where clg_id=5;
Query ID = root_20181023111957_bb7c4fc2-1bb2-4366-a555-7c80dce6ccfe
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1539070359797_0020)

--------------------------------------------------------------------------------
        VERTICES      STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 ..........   SUCCEEDED      5          5        0        0       0       0
Reducer 2 ......   SUCCEEDED      5          5        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 7.76 s
--------------------------------------------------------------------------------
Loading data to table custom.college
Table custom.college stats: [numFiles=7, numRows=0, totalSize=4987, rawDataSize=0]
OK
Time taken: 9.595 seconds
hive> select * from college;
OK
1       nec     nlr
6       IIT     atp
2       vit     vlr
7       cambridge       us
3       srm     chen
4       lpu     del
Time taken: 0.152 seconds, Fetched: 6 row(s)
```