Session 9 - ADVANCED HIVE

DATE SET DESCRIPTION

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

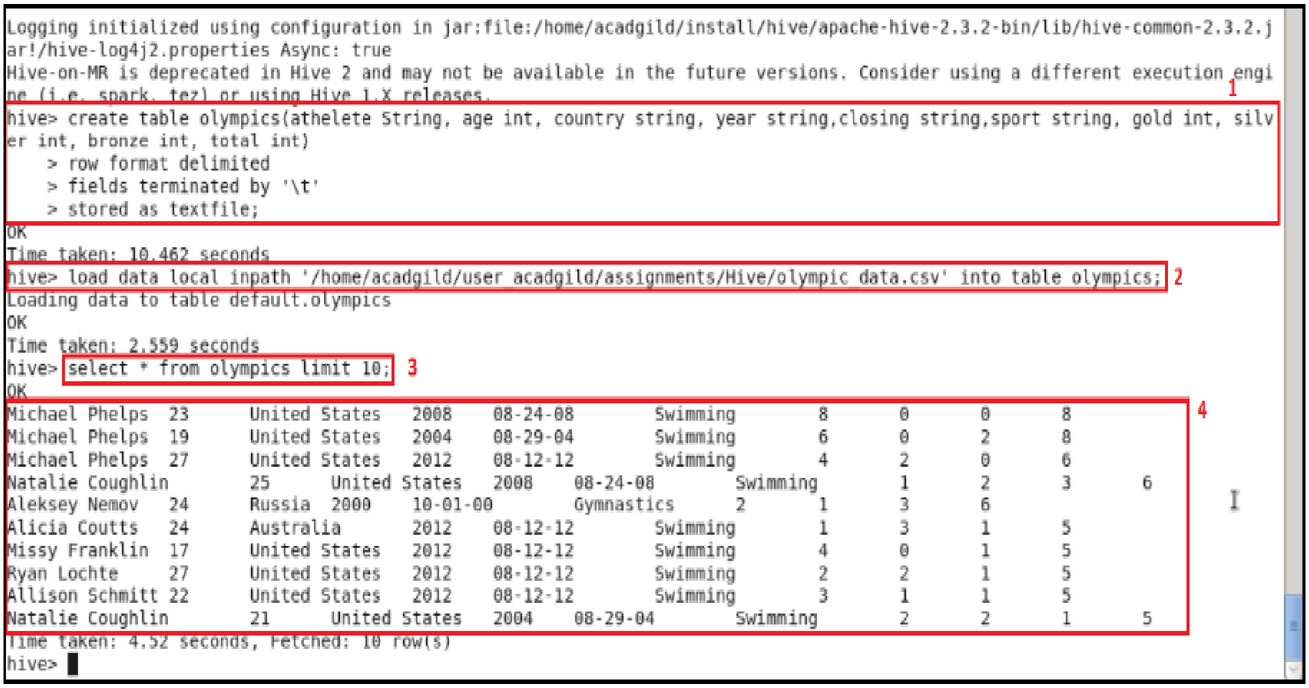
Silver Medals: No. of Silver medals

Bronze Medals: No. of Bronze medals

Total Medals: Consists of total no. of medals

==

Create a table ‘olympics’ using above mentioned information:



```
Logging initialized using configuration in jar:file:/home/acadgild/install/hive/apache-hive-2.3.2-bin/lib/hive-common-2.3.2.j
ar!/hive-log4j2.properties Async: true
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engi
ne (i.e. spark, tez) or using Hive 1.X releases.
hive> create table olympics(athelete String, age int, country string, year string,closing string,sport string, gold int, silv
er int, bronze int, total int)
    > row format delimited
    > fields terminated by '\t'
    > stored as textfile;
OK
Time taken: 10.462 seconds
hive> load data local inpath '/home/acadgild/user_acadgild/assignments/Hive/olympic_data.csv' into table olympics;
Loading data to table default.olympics
OK
Time taken: 2.559 seconds
hive> select * from olympics limit 10;
OK
Michael Phelps  23      United States   2008    08-24-08        Swimming        8       0       0       8
Michael Phelps  19      United States   2004    08-29-04        Swimming        6       0       2       8
Michael Phelps  27      United States   2012    08-12-12        Swimming        4       2       0       6
Natalie Coughlin        25      United States   2008    08-24-08        Swimming        1       2       3       6
Aleksey Nemov   24      Russia  2000    10-01-00        Gymnastics      2       1       3       6
Alicia Coutts   24      Australia       2012    08-12-12        Swimming        1       3       1       5
Missy Franklin  17      United States   2012    08-12-12        Swimming        4       0       1       5
Ryan Lochte     27      United States   2012    08-12-12        Swimming        2       2       1       5
Allison Schmitt 22      United States   2012    08-12-12        Swimming        3       1       1       5
Natalie Coughlin        21      United States   2004    08-29-04        Swimming        2       2       1       5
Time taken: 4.52 seconds, Fetched: 10 row(s)
hive>
```

1 : create a table using below:

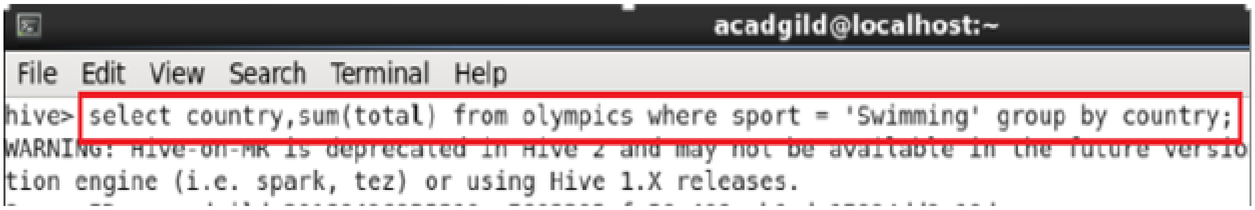
create table olympics(athelete string, age int,country string,year string,closing string,sport string, gold int, silver int, bronze int ,total int) row format delimited fields terminated by '\t' stored as textfile;

2: Loaded the data using below:

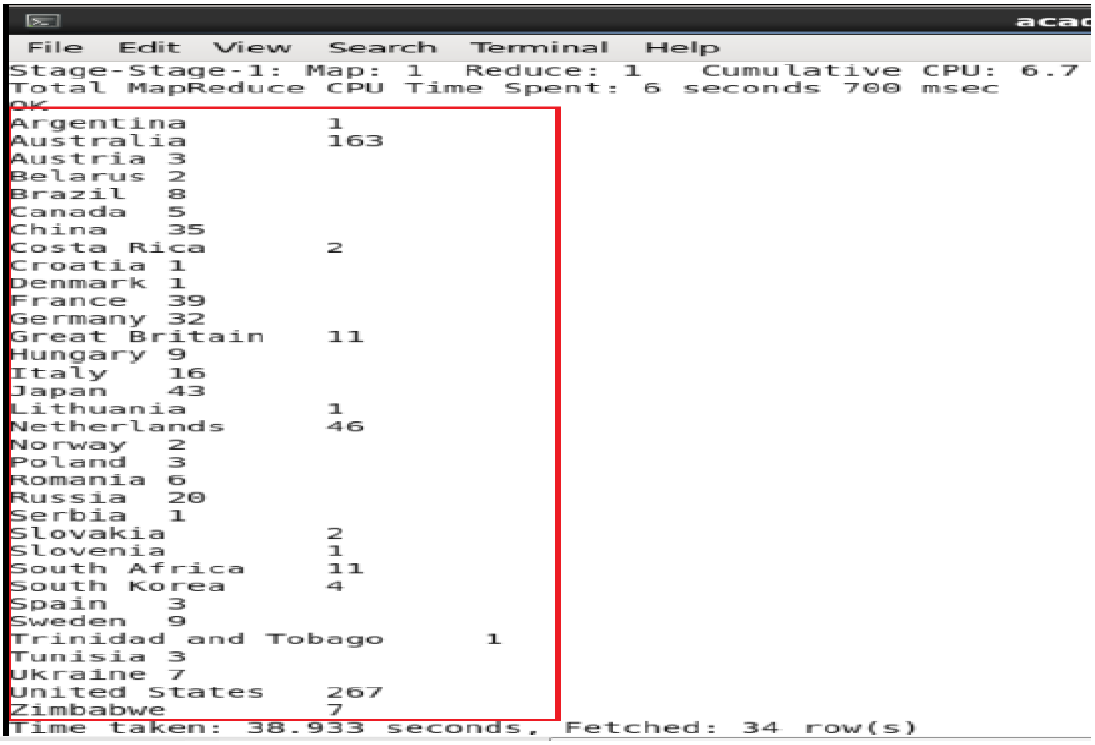
Load data local inpath '/home/acadgild/user_acadgild/assignments/Hive/olympic_data.csv' into table olympics;

3: display the contents of the table

Task: 1.1:
Write a Hive program to find the number of medals won by each country in swimming.
Solution: select country,sum(total) from olympics where sport = 'Swimming' group by country;



```
hive> select country,sum(total) from olympics where sport = 'Swimming' group by country;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versio
tion engine (i.e. spark, tez) or using Hive 1.X releases.
```

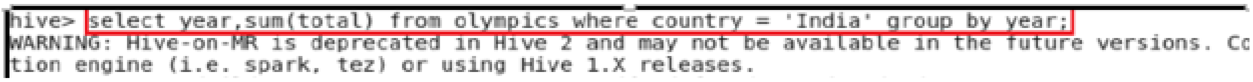


```
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 6.7
Total MapReduce CPU Time Spent: 6 seconds 700 msec
OK
Argentina       1
Australia       163
Austria 3
Belarus 2
Brazil  8
Canada  5
China   35
Costa Rica      2
Croatia 1
Denmark 1
France  39
Germany 32
Great Britain   11
Hungary 9
Italy   16
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: 38.933 seconds, Fetched: 34 row(s)
```

Task 1.2:

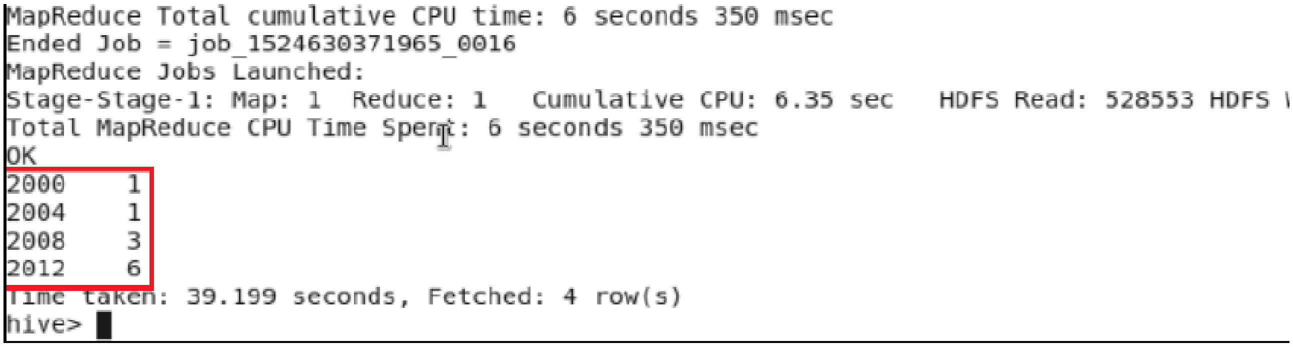
Write a Hive program to find the number of medals that India won year wise.

Solution: select year,sum(total) from olympics where country = ‘India’ group by year;



```
hive> select year,sum(total) from olympics where country = 'India' group by year;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Co
tion engine (i.e. spark, tez) or using Hive 1.X releases.
```

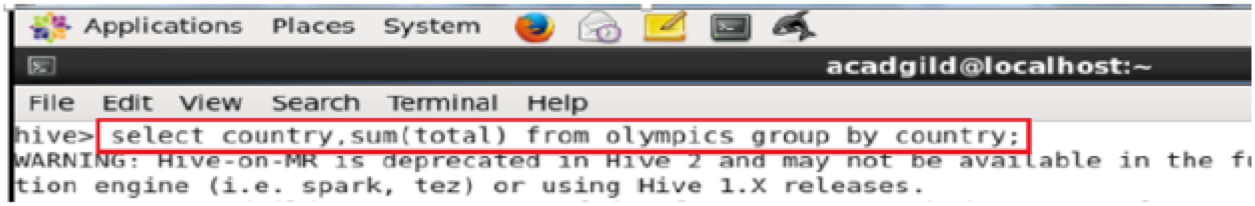
Output:



```
MapReduce Total cumulative CPU time: 6 seconds 350 msec
Ended Job = job_1524630371965_0016
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 6.35 sec   HDFS Read: 528553 HDFS
Total MapReduce CPU Time Spent: 6 seconds 350 msec
OK
2000    1
2004    1
2008    3
2012    6
Time taken: 39.199 seconds, Fetched: 4 row(s)
hive>
```

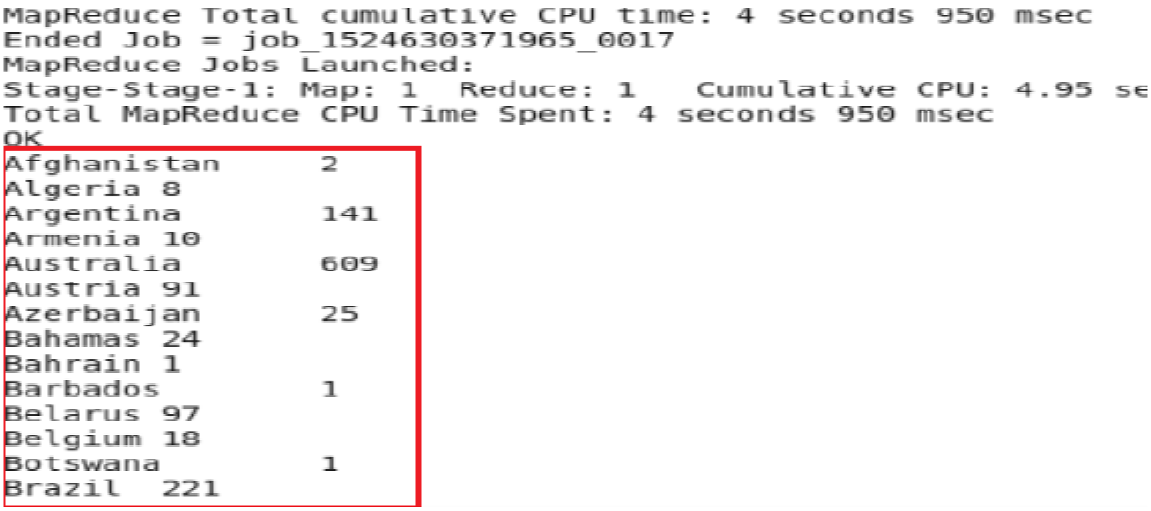
Task 1.3: Write a Hive Program to find the total number of medals each country won.

Solution: select country, sum(total) from olympics group by country;



```
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 f
tion engine (i.e. spark, tez) or using Hive 1.X releases.
```

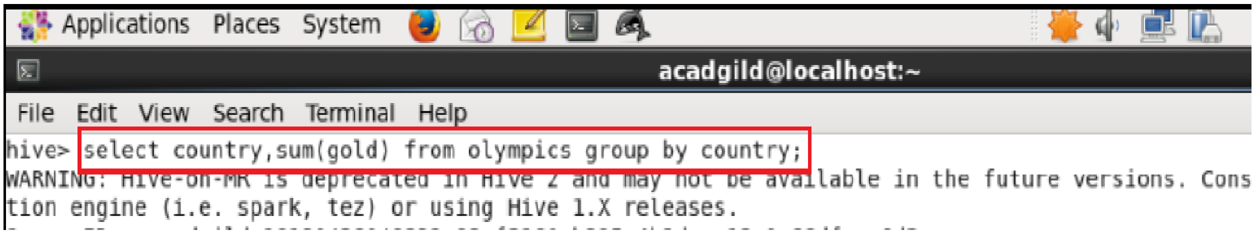
Output:



```
MapReduce Total cumulative CPU time: 4 seconds 950 msec
Ended Job = job_1524630371965_0017
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 4.95 se
Total MapReduce CPU Time Spent: 4 seconds 950 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
```

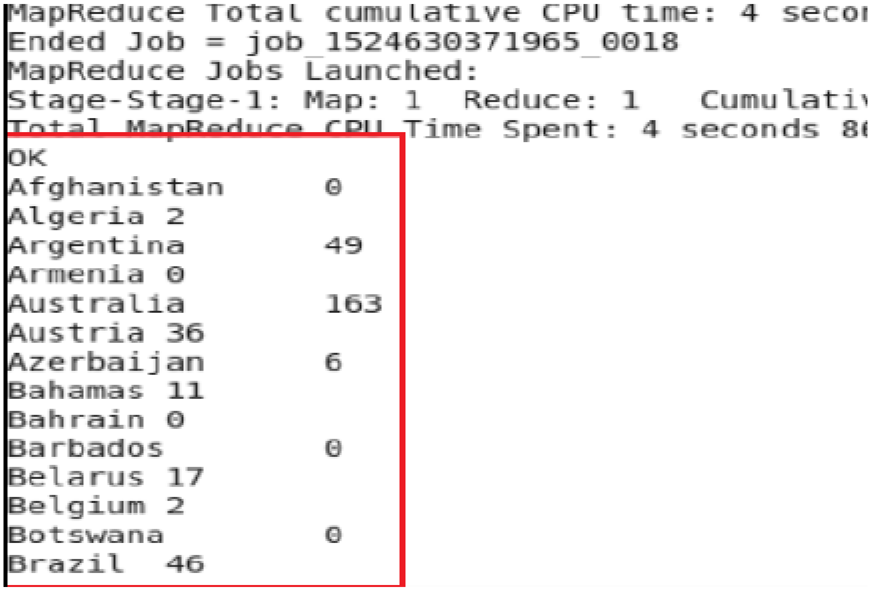
Task: 1.4 Write a Hive program to find the number of gold medals each country won

Solution: select country, sum(gold) from olympics group by country;


Applications Places System
acadgild@localhost:~
File Edit View Search Terminal Help

```
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. Cons
tion engine (i.e. spark, tez) or using Hive 1.X releases.
```

Output:



```
MapReduce Total cumulative CPU time: 4 seco
Ended Job = job_1524630371965_0018
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1   Reduce: 1   Cumulati
Total MapReduce CPU Time Spent: 4 seconds 8
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
```

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.

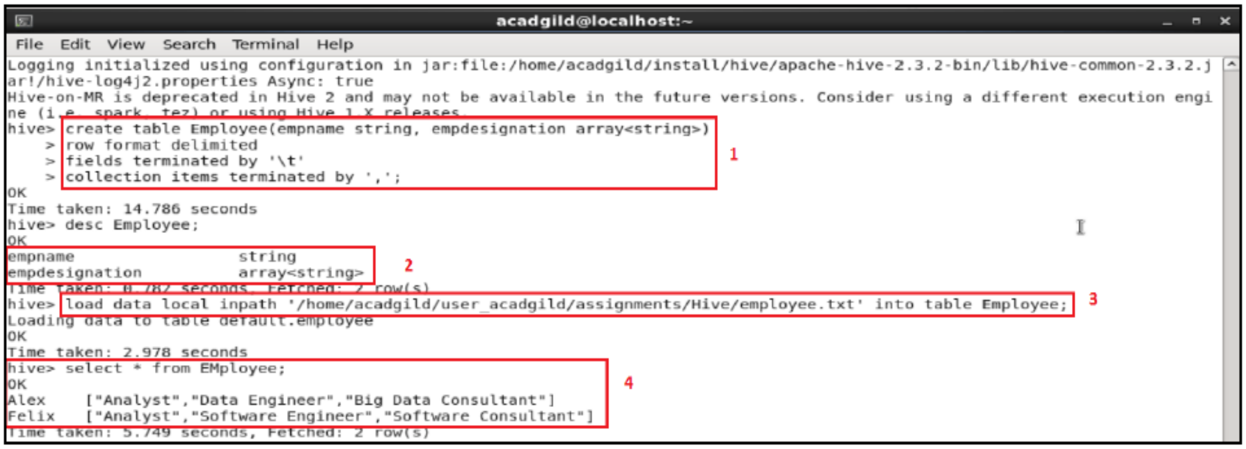
Solution: created by extending the *org.apache.hadoop.hive.ql.exec.UDF* class.

```
package com.acadgild.hiveudf;

import java.util.ArrayList;

import org.apache.commons.lang.StringUtils;
import org.apache.hadoop.hive.ql.exec.UDF;
import org.apache.hadoop.io.Text;

public class StringConcatUDF extends UDF {
    private Text result = new Text();
    public Text evaluate(String sep, ArrayList<String> stringChars) {
        if (sep == null) {
            return null;
        }
        String tempstr = "";
        for (int i = 0; i <= stringChars.size() - 1; i++) {
            tempstr = tempstr + (stringChars.get(i) + sep);
        }
        String finalstr = tempstr.substring(0, tempstr.length() - 1);
        result.set(finalstr);
        return result;
    }
    public Text evaluate(Text str) {
        if (str == null) {
            return null;
        }
        result.set(StringUtils.strip(str.toString()));
        return result;
    }
}
```

- Create a jar file for the java file.
- Add the jar in hive list of jars.

add jar '/location/of/the/jar/file'

Create a table with a column with array datatype.



```
Logging initialized using configuration in jar:file:/home/acadgild/install/hive/apache-hive-2.3.2-bin/lib/hive-common-2.3.2.j
ar!/hive-log4j2.properties Async: true
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engi
ne (i.e. spark, tez) or using Hive 1.X releases.
hive> create table Employee(empname string, empdesignation array<string>)
    > row format delimited
    > fields terminated by '\t'
    > collection items terminated by ',';
OK
Time taken: 14.786 seconds
hive> desc Employee;
OK
empname                 string
empdesignation          array<string>
Time taken: 0.782 seconds, Fetched: 2 row(s)
hive> load data local inpath '/home/acadgild/user_acadgild/assignments/Hive/employee.txt' into table Employee;
Loading data to table default.employee
OK
Time taken: 2.978 seconds
hive> select * from EMployee;
OK
Alex    ["Analyst","Data Engineer","Big Data Consultant"]
Felix   ["Analyst","Software Engineer","Software Consultant"]
Time taken: 5.749 seconds, Fetched: 2 row(s)
```

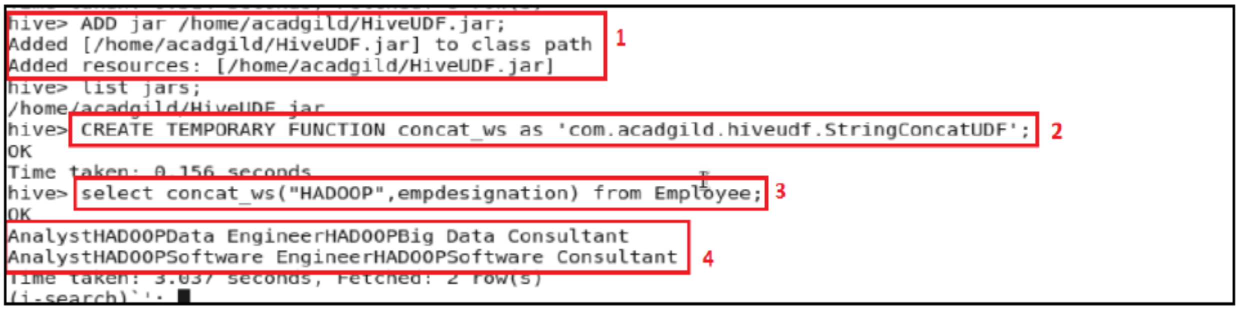
Using Above:

created a table employee where the fields are delimited using a tab space and the values in an array are separated using comma

where the datatype of the column is array.

sample data from a text file is loaded

The table is loaded with the data and the array can be seen



```
hive> ADD jar /home/acadgild/HiveUDF.jar;
Added [/home/acadgild/HiveUDF.jar] to class path
Added resources: [/home/acadgild/HiveUDF.jar]
hive> list jars;
/home/acadgild/HiveUDF.jar
hive> CREATE TEMPORARY FUNCTION concat_ws as 'com.acadgild.hiveudf.StringConcatUDF';
OK
Time taken: 0.156 seconds
hive> select concat_ws("HADOOP",empdesignation) from Employee;
OK
AnalystHADOOPData EngineerHADOOPBig Data Consultant
AnalystHADOOPSoftware EngineerHADOOPSoftware Consultant
Time taken: 3.037 seconds, Fetched: 2 row(s)
(i-search)`':
```

Next Steps:

* Adding jar to hive. Verifying the jar is added to hive, using 'list jars'.

* A temporary function is created with the classname to be used.

CREATE TEMPORARY FUNCTION concat_ws as 'com.acadgild.hiveudf.StringConcatUDF';

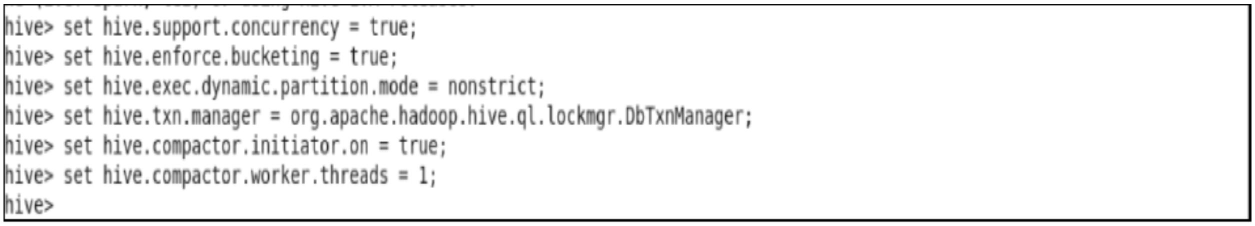
* Using the method.

select concat_ws("HADOOP",empdesignation) from Employee;

* The word HADOOP (1st arguement) is concatenated between each field in the array.

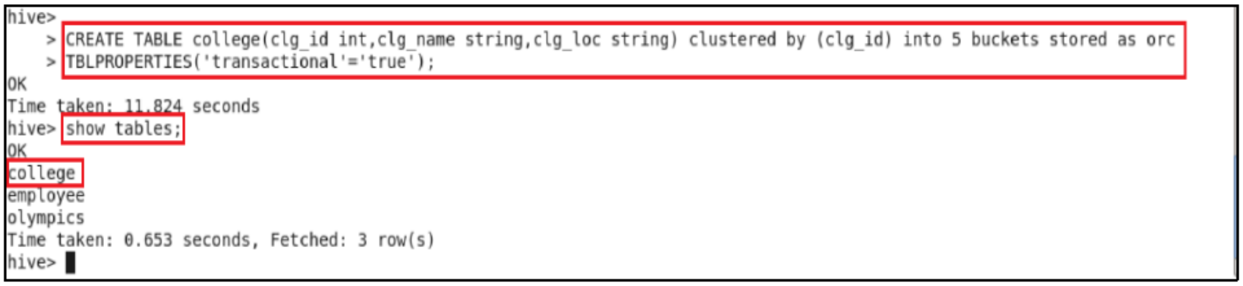
Task 3: Link: https://acadgild.com/blog/transactions-in-hive/
Refer the above given link for transactions in Hive and implement the operations given in the blog using your own sample data set and send us the screenshot.

The below properties needs to be set appropriately in hive shell , order-wise to work with transactions in Hive:
Creating



```
hive> set hive.support.concurrency = true;
hive> set hive.enforce.bucketing = true;
hive> set hive.exec.dynamic.partition.mode = nonstrict;
hive> set hive.txn.manager = org.apache.hadoop.hive.ql.lockmgr.DbTxnManager;
hive> set hive.compactor.initiator.on = true;
hive> set hive.compactor.worker.threads = 1;
hive>
```

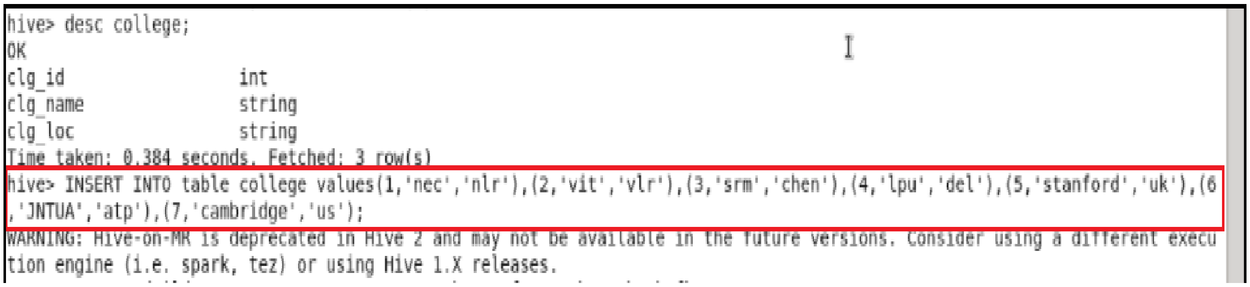
Created a table to support Hive Transactions :
CREATE TABLE college(clg_id int,clg_name string,clg_loc string) clustered by (clg_id) into 5 buckets stored as orc TBLPROPERTIES('transactional'='true');



```
hive>
    > CREATE TABLE college(clg_id int,clg_name string,clg_loc string) clustered by (clg_id) into 5 buckets stored as orc
    > TBLPROPERTIES('transactional'='true');
OK
Time taken: 11.824 seconds
hive> show tables;
OK
college
employee
olympics
Time taken: 0.653 seconds, Fetched: 3 row(s)
hive> 
```

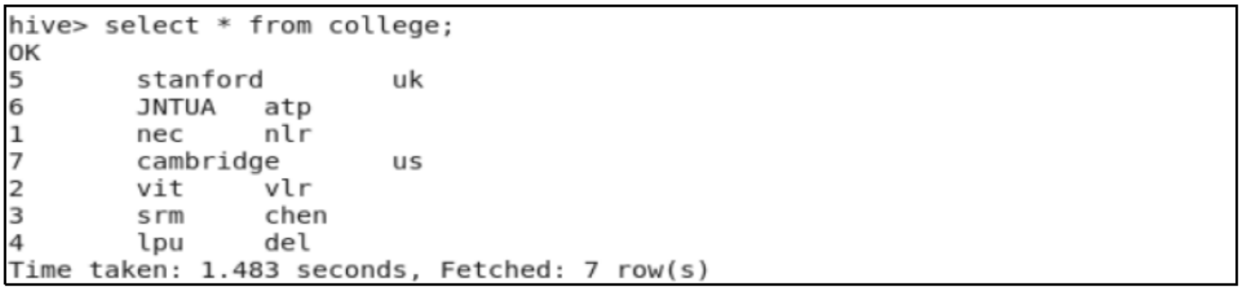
Insert Data into Hive Tables:

INSERT INTO table college values (1,'nec','nlr'),(2,'vit','vlr'),(3,'srm','chen'),(4,'lpu','del'),(5,'stanford','uk'),(6,'JNTUA','atp'),(7,'cambridge','us');



```
hive> desc college;
OK
clg_id                  int
clg_name                string
clg_loc                 string
Time taken: 0.384 seconds, Fetched: 3 row(s)
hive> INSERT INTO table college values(1,'nec','nlr'),(2,'vit','vlr'),(3,'srm','chen'),(4,'lpu','del'),(5,'stanford','uk'),(6
,'JNTUA','atp'),(7,'cambridge','us');
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execu
tion engine (i.e. spark, tez) or using Hive 1.X releases.
```

Date is now inserted into the table:



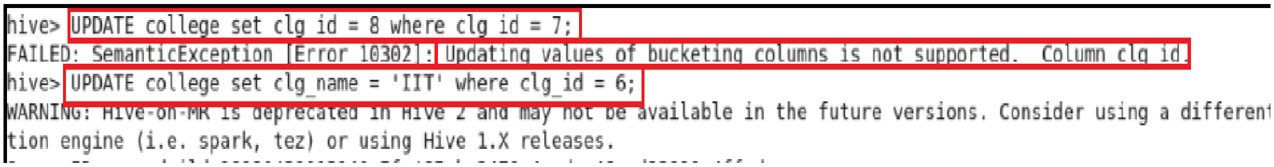
```
hive> select * from college;
OK
5       stanford        uk
6       JNTUA   atp
1       nec     nlr
7       cambridge       us
2       vit     vlr
3       srm     chen
4       lpu     del
Time taken: 1.483 seconds, Fetched: 7 row(s)
```

Update the Data in Hive Table:

UPDATE college set clg_id = 8 where clg_id = 7;

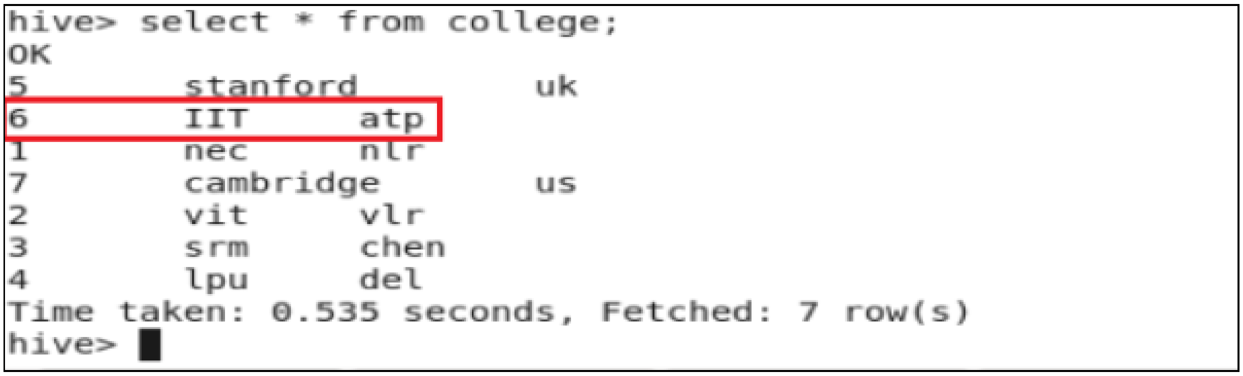
Bucketed column cannot be udpated. Only non bucketed columns can be updated.

UPDATE college set clg_name = 'IIT' where clg_id = 6;



```
hive> UPDATE college set clg_id = 8 where clg_id = 7;
FAILED: SemanticException [Error 10302]: Updating values of bucketing columns is not supported.  Column clg_id.
hive> UPDATE college set clg_name = 'IIT' where clg_id = 6;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different
tion engine (i.e. spark, tez) or using Hive 1.X releases.
```

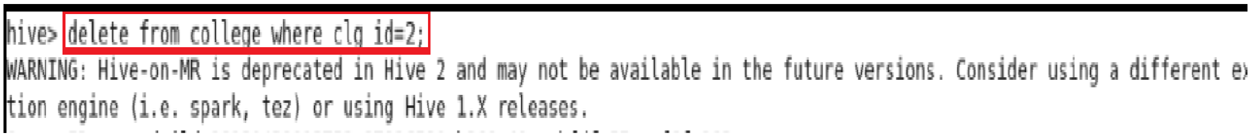
The updated values in tabele are as below:



```
hive> select * from college;
OK
5       stanford        uk
6       IIT     atp
1       nec     nlr
7       cambridge       us
2       vit     vlr
3       srm     chen
4       lpu     del
Time taken: 0.535 seconds, Fetched: 7 row(s)
hive> █
```

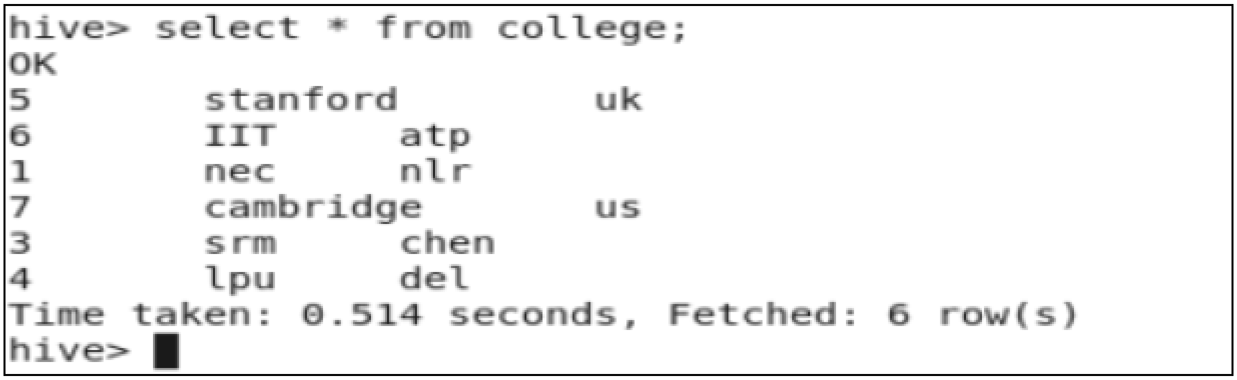
Deleting a row from the table :

delete from college where clg_id = 2;



```
hive> delete from college where clg_id=2;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different e
tion engine (i.e. spark, tez) or using Hive 1.X releases.
```

Data from Table now:



```
hive> select * from college;
OK
5       stanford        uk
6       IIT     atp
1       nec     nlr
7       cambridge       us
3       srm     chen
4       lpu     del
Time taken: 0.514 seconds, Fetched: 6 row(s)
hive> 
```

Row with clg_id 2 is deleted from the table