Session 9: ADVANCED HIVE

Assignment 1

Task 1:

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

Solution:

```
hive> select SUM(totalmedals)AS medals,country 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 versions. Consider using a diff
```

Output:

```
Total MapReduce CPU Time Spent: 4 seconds 640 r
OK
medals  country
1       Argentina
163     Australia
3       Austria
2       Belarus
8       Brazil
5       Canada
35      China
2       Costa Rica
1       Croatia
1       Denmark
39      France
32      Germany
11      Great Britain
9       Hungary
16      Italy
43      Japan
1       Lithuania
46      Netherlands
2       Norway
3       Poland
6       Romania
20      Russia
1       Serbia
2       Slovakia
1       Slovenia
11      South Africa
4       South Korea
3       Spain
9       Sweden
1       Trinidad and Tobago
3       Tunisia
7       Ukraine
267     United States
7       Zimbabwe
Time taken: 29.527 seconds, Fetched: 34 row(s)
```

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

Solution:

```
Time taken: 29.527 seconds, Fetched: 34 row(s)
hive> select SUM(totalmedals)AS medals,year 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. Consid
```

Output:

```
OK
medals  year
1       2000
1       2004
3       2008
6       2012
Time taken: 29.188
```

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

Solution:

```
hive> select SUM(totalmedals)AS medals,country from olympics group by country;
```

Output:

```
OK
medals  country
2       Afghanistan
8       Algeria
141     Argentina
10      Armenia
609     Australia
91      Austria
25      Azerbaijan
24      Bahamas
1       Bahrain
1       Barbados
97      Belarus
18      Belgium
1       Botswana
221     Brazil
41      Bulgaria
20      Cameroon
370     Canada
22      Chile
530     China
20      Chinese Taipei
13      Colombia
2       Costa Rica
```

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

Solution:

```
hive> select SUM(goldmedals)AS gold_medals,country from olympics group by country;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versi
```

Output:

```
OK
gold_medals     country
0       Afghanistan
2       Algeria
49      Argentina
0       Armenia
163     Australia
36      Austria
6       Azerbaijan
11      Bahamas
0       Bahrain
0       Barbados
17      Belarus
2       Belgium
0       Botswana
46      Brazil
8       Bulgaria
20      Cameroon
168     Canada
3       Chile
234     China
2       Chinese Taipei
2       Colombia
0       Costa Rica
35      Croatia
57      Cuba
0       Cyprus
14      Czech Republic
46      Denmark
3       Dominican Republic
0       Ecuador
1       Egypt
0       Eritrea
6       Estonia
13      Ethiopia
11      Finland
108     France
```

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:

```java
package task1_Assignment5;

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

public class stringsep_UDF extends UDF {

    String result = new String();
    public String evaluate(String sep, String[] text) {

        int l = text.length;
        for(int i=0;i<l;i++) {
            result = text[i]+ sep;
        }

        return result;
    }

}
```

Output:

We packaged the code in a jar and added to hive and then run the function on a table of data as shown below:

```
hive> select * from temp1;
OK
temp1._c0       temp1.cnt       temp1.yeardata
23      7       1990
22      9       1991
16      2       1993
23      2       1994
Time taken: 0.161 seconds, Fetched: 4 row(s)
```

```
hive> create temporary function concat_ws as "task1_Assignment5.stringsep_UDF";
```

```
hive> select concat_ws("/",temp1.yeardata) from temp1;
OK
_c0
1990/
1991/
1993/
1994/
Time taken: 0.153 seconds, Fetched: 4 row(s)
```

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.

Output:

```
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 = 4;
```

CREATE Statement:

```
hive> CREATE TABLE college(clg_id int,clg_name string,clg_loc string) clustered by (clg_id) into 5 buckets stored as ORC TBLP
ROPERTIES('transactional'='true');
OK
Time taken: 1.155 seconds
hive> show tables;
OK
tab_name
college
olympics
temp1
temperature_data
temperature_data_temp
temperature_data_vw
Time taken: 0.104 seconds, Fetched: 6 row(s)
```

INSERT Statement:

```
hive> INSERT into table college values(1,'nec','nellore'),(2,'vit','vellore'),(3,'srm','chennai'),(4,'mgr','chennai'),(5,'bit
s','pilani'),(6,'stanford','uk'),(7,'bitmesra','ranchi');
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execu
```

```
hive> select * from college;
OK
college.clg_id  college.clg_name         college.clg_loc
5       bits    pilani
6       stanford        uk
1       nec     nellore
7       bitmesra        ranchi
2       vit     vellore
3       srm     chennai
4       mgr     chennai
Time taken: 0.23 seconds, Fetched: 7 row(s)
hive>
```

UPDATE Statement:

```
hive> UPDATE college set clg_name='IIT' where clg_id=7;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the fu
hive> select * from college;
OK
college.clg_id  college.clg_name         college.clg_loc
5       bits    pilani
6       stanford        uk
1       nec     nellore
7       IIT     ranchi
2       vit     vellore
3       srm     chennai
4       mgr     chennai
Time taken: 0.352 seconds, Fetched: 7 row(s)
```

DELETE Statement:

```
hive> DELETE from college where clg_id=7;
hive> select * from college;
OK
college.clg_id  college.clg_name         college.clg_loc
5       bits    pilani
6       stanford        uk
1       nec     nellore
2       vit     vellore
3       srm     chennai
4       mgr     chennai
Time taken: 0.247 seconds, Fetched: 6 row(s)
hive>
```