

## Task 2: Load data into chosen cloud technology

### 1. Load and merge the data in HDFS:

- A. Firstly, upload the 5 downloaded CSV files in the local home directory and check whether the same are present in it using the ls command.

```
hduser@Dell:/usr/local/hadoop$ ls
```

- B. Then, move the files on Hadoop using the put command and check for their presence.

```
hduser@Dell:/usr/local/hadoop$ hadoop fs -put QueryResults.csv /
```

```
hduser@Dell:/usr/local/hadoop$ hadoop fs -put QueryResults_1.csv /
```

```
hduser@Dell:/usr/local/hadoop$ hadoop fs -put QueryResults_2.csv /
```

```
hduser@Dell:/usr/local/hadoop$ hadoop fs -put QueryResults_3.csv /
```

```
hduser@Dell:/usr/local/hadoop$ hadoop fs -put QueryResults_4.csv /
```

```
hduser@Dell:/usr/local/hadoop$ hadoop fs -ls /
```

- C. Lastly, merge all the 5 CSV files into a single CSV file using the cat command and again check whether that single CSV file is present in it.

```
hduser@Dell:/usr/local/hadoop$ cat QueryResults.csv QueryResults_1.csv  
QueryResults_2.csv QueryResults_3.csv QueryResults_4.csv > Final_QueryResults.csv
```

```
hduser@Dell:/usr/local/hadoop$ hadoop fs -ls /
```

```
hduser@Dell:/usr/local/hadoop$ ls
LICENSE-binary  NOTICE-binary  QueryResults.csv  QueryResults_2.csv  QueryResults_4.csv  bin  include  libexec  logs  share
LICENSE.txt     NOTICE.txt     QueryResults_1.csv  QueryResults_3.csv  README.txt          etc  lib      licenses-binary  sbin

hduser@Dell:/usr/local/hadoop$ hadoop fs -put QueryResults.csv /
2021-10-22 13:26:50,345 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hduser@Dell:/usr/local/hadoop$ hadoop fs -put QueryResults_1.csv /
2021-10-22 13:27:31,796 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hduser@Dell:/usr/local/hadoop$ hadoop fs -put QueryResults_2.csv /
2021-10-22 13:27:45,357 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hduser@Dell:/usr/local/hadoop$ hadoop fs -put QueryResults_3.csv /
2021-10-22 13:27:57,928 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hduser@Dell:/usr/local/hadoop$ hadoop fs -put QueryResults_4.csv /
2021-10-22 13:28:08,852 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hduser@Dell:/usr/local/hadoop$ hadoop fs -ls /
2021-10-22 13:28:35,376 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 8 items
-rw-r--r-- 1 hduser supergroup 54871720 2021-10-22 13:26 /QueryResults.csv
-rw-r--r-- 1 hduser supergroup 62237507 2021-10-22 13:27 /QueryResults_1.csv
-rw-r--r-- 1 hduser supergroup 62794654 2021-10-22 13:27 /QueryResults_2.csv
-rw-r--r-- 1 hduser supergroup 68622160 2021-10-22 13:27 /QueryResults_3.csv
-rw-r--r-- 1 hduser supergroup 14598 2021-10-22 13:28 /QueryResults_4.csv
drwxr-xr-x - hduser supergroup 0 2021-10-15 15:49 /bigdata
drwxr-xr-x - hduser supergroup 0 2021-10-20 16:34 /tmp
drwxr-xr-x - hduser supergroup 0 2021-10-20 16:41 /user

hduser@Dell:/usr/local/hadoop$ cat QueryResults.csv QueryResults_1.csv QueryResults_2.csv QueryResults_3.csv QueryResults_4.csv > Final_QueryResults.csv
hduser@Dell:/usr/local/hadoop$ hadoop fs -ls /
2021-10-22 14:30:10,950 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 9 items
-rw-r--r-- 1 hduser supergroup 248540639 2021-10-22 14:25 /Final_QueryResults.csv
-rw-r--r-- 1 hduser supergroup 54871720 2021-10-22 13:26 /QueryResults.csv
-rw-r--r-- 1 hduser supergroup 62237507 2021-10-22 13:27 /QueryResults_1.csv
-rw-r--r-- 1 hduser supergroup 62794654 2021-10-22 13:27 /QueryResults_2.csv
-rw-r--r-- 1 hduser supergroup 68622160 2021-10-22 13:27 /QueryResults_3.csv
-rw-r--r-- 1 hduser supergroup 14598 2021-10-22 13:28 /QueryResults_4.csv
drwxr-xr-x - hduser supergroup 0 2021-10-15 15:49 /bigdata
drwxr-xr-x - hduser supergroup 0 2021-10-22 14:04 /tmp
drwxr-xr-x - hduser supergroup 0 2021-10-20 16:41 /user
hduser@Dell:/usr/local/hadoop$
```

## 2. Clean the data in PIG:

- A. Firstly, load the data from HDFS to PIG, specifying each data type.

```
grunt>stackdata=LOAD'hdfs://localhost:9870/usr/local/hadoopFinal_QueryResults.csv' USING PigStorage(',') AS (id:int, posttypeid:int, acceptedanswerid:int, parentid:int, creationdate:chararray, deletiondate:chararray, score:int, viewcount:int, body:chararray, owneruserid:int, ownerdisplayname:chararray, lasteditoruserid:int, lasteditordisplayname:chararray, lasteditdate:chararray, lastactivitydate:chararray, title:chararray, tags:chararray, answercount:int, commentcount:int, favoritecount:int, closeddate:chararray, communityowneddate:chararray );
```

```
grunt> DESCRIBE stackdata;
```

- B. Now, we do not need every field present in the data. Therefore, we generate a new table with the required fields only and also remove the empty records present in the data.

```
grunt> pickCols = FOREACH stackdata GENERATE id, score, viewcount, owneruserid, title, tags, (REPLACE(body,'\r\n',' ')) AS body;
```

```
grunt> DESCRIBE pickCols;
```

- C. Further, we need to remove the duplicate records present in the data. So, removing the same using the DISTINCT function.

```
grunt> datadistinct = DISTINCT pickCols;
```

- D. Also, the additional records after the 200,000 records count is completed need to be removed. Therefore, removing the same using the LIMIT function and thereafter reverifying the count of the remaining records using the COUNT\_STAR function which return the value 200,000!

```
grunt> datalimit = LIMIT datadistinct 200000;
```

```
grunt> stackfull = GROUP datalimit ALL;
```

```
grunt> stackcount = FOREACH stackfull GENERATE COUNT_STAR(datalimit.id) AS cnt;
```

```
grunt> dump stackcount;
```

- E. Lastly, creating a new file folder to save this cleaned up data.

```
grunt> STORE datalimit INTO 'Pig_QueryResults' USING PigStorage(',');
```

```
grunt> stackdata = LOAD '/hadoopFinal/QueryResults.csv' USING PigStorage(',') AS (id:int, posttypeid:int, acceptedanswerid:int, parentid:int, creationdate:chararray, deletiondate:chararray, score:int, viewcount:int, body:chararray, owneruserid:int, ownerdisplayname:chararray, lasteditoruserid:int, lasteditordisplayname:chararray, lasteditdate:chararray, lastactivitydate:chararray, title:chararray, tags:chararray, answercount:int, commentcount:int, favoritecount:int, closeddate:chararray, communityowneddate:chararray);
grunt> DESCRIBE stackdata;
stackdata: (id: int,posttypeid: int,acceptedanswerid: int,parentid: int,creationdate: chararray,deletiondate: chararray,score: int,viewcount: int,body: chararray,owneruserid: int,ownerdisplayname: chararray,lasteditoruserid: int,lasteditordisplayname: chararray,lasteditdate: chararray,lastactivitydate: chararray,title: chararray,tags: chararray,answercount: int,commentcount: int,favoritecount: int,closeddate: chararray,communityowneddate: chararray)
grunt> pickCols = FOREACH stackdata GENERATE id, score, viewcount, owneruserid, title, tags, (REPLACE(body, '\n',' ')) AS body;
grunt> DESCRIBE pickCols;
pickCols: (id: int,score: int,viewcount: int,owneruserid: int,title: chararray,tags: chararray,body: chararray)
grunt> datadistinct = DISTINCT pickCols;
grunt> datalimit = LIMIT datadistinct 200000;
grunt> stackfull = GROUP datalimit ALL;
grunt> stackcount = FOREACH stackfull GENERATE COUNT_STAR(datalimit.id) AS cnt;
grunt> dump stackcount;
2021-10-23 23:46:19,831 [main] INFO org.apache.pig.tools.pigstats.ScriptState - Pig features used in the script: GROUP_BY,DISTINCT,LIMIT
2021-10-23 23:46:19,894 [main] INFO org.apache.pig.data.SchemaUpBackend - Key [pig.schemaup] was not set... will not generate code.
```