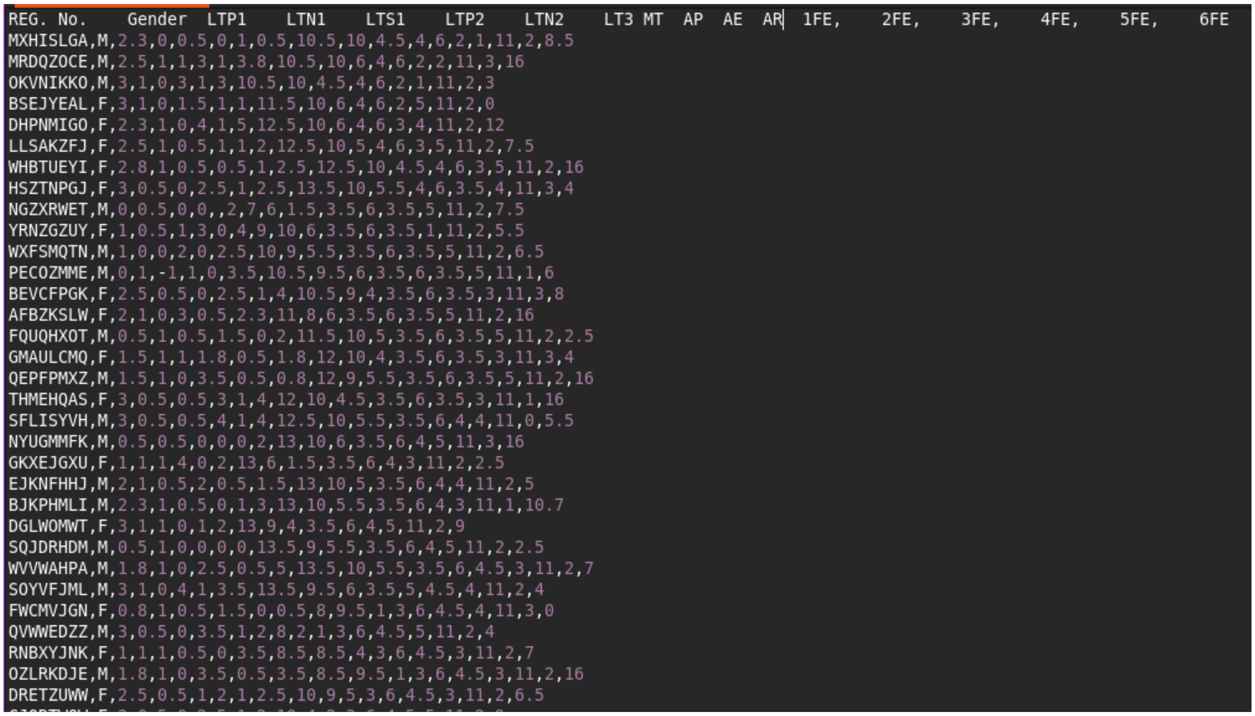
Lab Test WQD7007

Part 1

1. Question 1

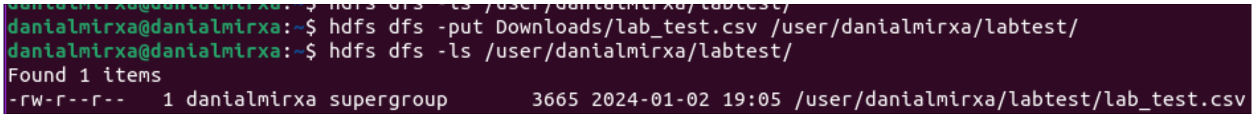
- The dataset Set 4 and Set 4 is combined using text editor, with Set 4 on top



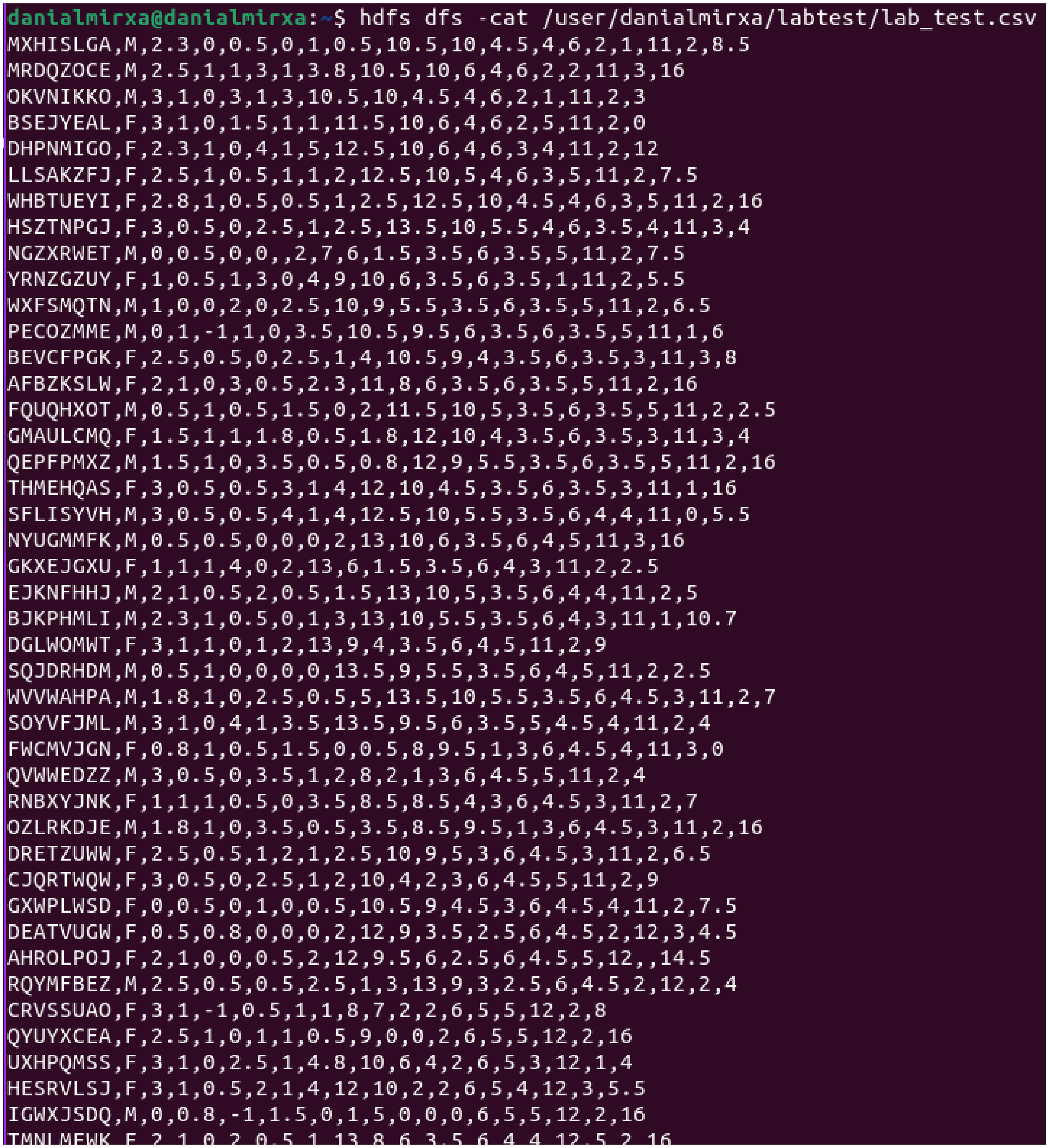
```
REG. No.    Gender  LTP1    LTN1    LTS1    LTP2    LTN2    LT3 MT  AP  AE  AR  1FE,    2FE,    3FE,    4FE,    5FE,    6FE
MXHISLGA,M,2.3,0,0.5,0,1,0.5,10.5,10,4.5,4,6,2,1,11,2,8.5
MRDQZOCE,M,2.5,1,1,3,1,3.8,10.5,10,6,4,6,2,2,11,3,16
OKVNIKKO,M,3,1,0,3,1,3,10.5,10,4.5,4,6,2,1,11,2,3
BSEJYEAL,F,3,1,0,1.5,1,1,11.5,10,6,4,6,2,5,11,2,0
DHPNMIGO,F,2.3,1,0,4,1,5,12.5,10,6,4,6,3,4,11,2,12
LLSAKZFJ,F,2.5,1,0.5,1,1,2,12.5,10,5,4,6,3,5,11,2,7.5
WHBTUEYI,F,2.8,1,0.5,0.5,1,2.5,12.5,10,4.5,4,6,3,5,11,2,16
HSZTNPGJ,F,3,0.5,0,2.5,1,2.5,13.5,10,5.5,4,6,3.5,4,11,3,4
NGZXRWET,M,0,0.5,0,0,,2,7,6,1.5,3.5,6,3.5,5,11,2,7.5
YRNZGZUY,F,1,0.5,1,3,0,4,9,10,6,3.5,6,3.5,1,11,2,5.5
WXFSMQTN,M,1,0,0,2,0,2.5,10,9,5.5,3.5,6,3.5,5,11,2,6.5
PECOZMME,M,0,1,-1,1,0,3.5,10.5,9.5,6,3.5,6,3.5,5,11,1,6
BEVCFPGK,F,2.5,0.5,0,2.5,1,4,10.5,9,4,3.5,6,3.5,3,11,3,8
AFBZKSLW,F,2,1,0,3,0.5,2.3,11,8,6,3.5,6,3.5,5,11,2,16
FQUQHXOT,M,0.5,1,0.5,1.5,0,2,11.5,10,5,3.5,6,3.5,5,11,2,2.5
GMAULCMQ,F,1.5,1,1,1.8,0.5,1.8,12,10,4,3.5,6,3.5,3,11,3,4
QEPFPMXZ,M,1.5,1,0,3.5,0.5,0.8,12,9,5.5,3.5,6,3.5,5,11,2,16
THMEHQAS,F,3,0.5,0.5,3,1,4,12,10,4.5,3.5,6,3.5,3,11,1,16
SFLISYVH,M,3,0.5,0.5,4,1,4,12.5,10,5.5,3.5,6,4,4,11,0,5.5
NYUGMMFK,M,0.5,0.5,0,0,0,2,13,10,6,3.5,6,4,5,11,3,16
GKXEJGXU,F,1,1,1,4,0,2,13,6,1.5,3.5,6,4,3,11,2,2.5
EJKNFHHJ,M,2,1,0.5,2,0.5,1.5,13,10,5,3.5,6,4,4,11,2,5
BJKPHMLI,M,2.3,1,0.5,0,1,3,13,10,5.5,3.5,6,4,3,11,1,10.7
DGLWOMWT,F,3,1,1,0,1,2,13,9,4,3.5,6,4,5,11,2,9
SQJDRHDM,M,0.5,1,0,0,0,0,13.5,9,5.5,3.5,6,4,5,11,2,2.5
WVVWAHPA,M,1.8,1,0,2.5,0.5,5,13.5,10,5.5,3.5,6,4.5,3,11,2,7
SOYVFJML,M,3,1,0,4,1,3.5,13.5,9.5,6,3.5,5,4.5,4,11,2,4
FWCMVJGN,F,0.8,1,0.5,1.5,0,0.5,8,9.5,1,3,6,4.5,4,11,3,0
QVWWEDZZ,M,3,0.5,0,3.5,1,2,8,2,1,3,6,4.5,5,11,2,4
RNBXYJNK,F,1,1,1,0.5,0,3.5,8.5,8.5,4,3,6,4.5,3,11,2,7
OZLRKDJE,M,1.8,1,0,3.5,0.5,3.5,8.5,9.5,1,3,6,4.5,3,11,2,16
DRETZUWW,F,2.5,0.5,1,2,1,2.5,10,9,5,3,6,4.5,3,11,2,6.5
```

- The dataset is name as lab_test.csv and uploaded to hdfs at

  /user/danialmirxa/labtest/lab_test.csv



```
danialmirxa@danialmirxa:~$ hdfs dfs -ls /user/danialmirxa/labtest/
danialmirxa@danialmirxa:~$ hdfs dfs -put Downloads/lab_test.csv /user/danialmirxa/labtest/
danialmirxa@danialmirxa:~$ hdfs dfs -ls /user/danialmirxa/labtest/
Found 1 items
-rw-r--r--   1 danialmirxa supergroup       3665 2024-01-02 19:05 /user/danialmirxa/labtest/lab_test.csv
```



```
danialmirxa@danialmirxa:~$ hdfs dfs -cat /user/danialmirxa/labtest/lab_test.csv
MXHISLGA,M,2.3,0,0.5,0,1,0.5,10.5,10,4.5,4,6,2,1,11,2,8.5
MRDQZOCE,M,2.5,1,1,3,1,3.8,10.5,10,6,4,6,2,2,11,3,16
OKVNIKKO,M,3,1,0,3,1,3,10.5,10,4.5,4,6,2,1,11,2,3
BSEJYEAL,F,3,1,0,1.5,1,1,11.5,10,6,4,6,2,5,11,2,0
DHPNMIGO,F,2.3,1,0,4,1,5,12.5,10,6,4,6,3,4,11,2,12
LLSAKZFJ,F,2.5,1,0.5,1,1,2,12.5,10,5,4,6,3,5,11,2,7.5
WHBTUEYI,F,2.8,1,0.5,0.5,1,2.5,12.5,10,4.5,4,6,3,5,11,2,16
HSZTNPGJ,F,3,0.5,0,2.5,1,2.5,13.5,10,5.5,4,6,3.5,4,11,3,4
NGZXRWET,M,0,0.5,0,0,,2,7,6,1.5,3.5,6,3.5,5,11,2,7.5
YRNZGZUY,F,1,0.5,1,3,0,4,9,10,6,3.5,6,3.5,1,11,2,5.5
WXFSMQTN,M,1,0,0,2,0,2.5,10,9,5.5,3.5,6,3.5,5,11,2,6.5
PECOZMME,M,0,1,-1,1,0,3.5,10.5,9.5,6,3.5,6,3.5,5,11,1,6
BEVCFPGK,F,2.5,0.5,0,2.5,1,4,10.5,9,4,3.5,6,3.5,3,11,3,8
AFBZKSLW,F,2,1,0,3,0.5,2.3,11,8,6,3.5,6,3.5,5,11,2,16
FQUQHXOT,M,0.5,1,0.5,1.5,0,2,11.5,10,5,3.5,6,3.5,5,11,2,2.5
GMAULCMQ,F,1.5,1,1,1.8,0.5,1.8,12,10,4,3.5,6,3.5,3,11,3,4
QEPFPMXZ,M,1.5,1,0,3.5,0.5,0.8,12,9,5.5,3.5,6,3.5,5,11,2,16
THMEHQAS,F,3,0.5,0.5,3,1,4,12,10,4.5,3.5,6,3.5,3,11,1,16
SFLISYVH,M,3,0.5,0.5,4,1,4,12.5,10,5.5,3.5,6,4,4,11,0,5.5
NYUGMMFK,M,0.5,0.5,0,0,0,2,13,10,6,3.5,6,4,5,11,3,16
GKXEJGXU,F,1,1,1,4,0,2,13,6,1.5,3.5,6,4,3,11,2,2.5
EJKNFHHJ,M,2,1,0.5,2,0.5,1.5,13,10,5,3.5,6,4,4,11,2,5
BJKPHMLI,M,2.3,1,0.5,0,1,3,13,10,5.5,3.5,6,4,3,11,1,10.7
DGLWOMWT,F,3,1,1,0,1,2,13,9,4,3.5,6,4,5,11,2,9
SQJDRHDM,M,0.5,1,0,0,0,0,13.5,9,5.5,3.5,6,4,5,11,2,2.5
WVVWAHPA,M,1.8,1,0,2.5,0.5,5,13.5,10,5.5,3.5,6,4.5,3,11,2,7
SOYVFJML,M,3,1,0,4,1,3.5,13.5,9.5,6,3.5,5,4.5,4,11,2,4
FWCMVJGN,F,0.8,1,0.5,1.5,0,0.5,8,9.5,1,3,6,4.5,4,11,3,0
QVWWEDZZ,M,3,0.5,0,3.5,1,2,8,2,1,3,6,4.5,5,11,2,4
RNBXYJNK,F,1,1,1,0.5,0,3.5,8.5,8.5,4,3,6,4.5,3,11,2,7
OZLRKDJE,M,1.8,1,0,3.5,0.5,3.5,8.5,9.5,1,3,6,4.5,3,11,2,16
DRETZUWW,F,2.5,0.5,1,2,1,2.5,10,9,5,3,6,4.5,3,11,2,6.5
CJQRTWQW,F,3,0.5,0,2.5,1,2,10,4,2,3,6,4.5,5,11,2,9
GXWPLWSD,F,0,0.5,0,1,0,0.5,10.5,9,4.5,3,6,4.5,4,11,2,7.5
DEATVUGW,F,0.5,0.8,0,0,0,2,12,9,3.5,2.5,6,4.5,2,12,3,4.5
AHROLPOJ,F,2,1,0,0,0.5,2,12,9.5,6,2.5,6,4.5,5,12,,14.5
RQYMFBEZ,M,2.5,0.5,0.5,2.5,1,3,13,9,3,2.5,6,4.5,2,12,2,4
CRVSSUAO,F,3,1,-1,0.5,1,1,8,7,2,2,6,5,5,12,2,8
QYUYXCEA,F,2.5,1,0,1,1,0.5,9,0,0,2,6,5,5,12,2,16
UXHPQMSS,F,3,1,0,2.5,1,4.8,10,6,4,2,6,5,3,12,1,4
HESRVLSJ,F,3,1,0.5,2,1,4,12,10,2,2,6,5,4,12,3,5.5
IGWXJSDQ,M,0,0.8,-1,1.5,0,1,5,0,0,0,6,5,5,12,2,16
TMNLMEWK,F,2,1,0,2,0.5,1,13,8,6,3.5,6,4,4,12,5,2,16
```

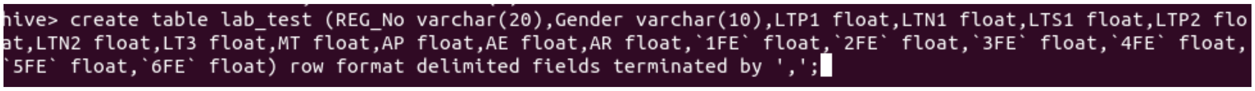
- The headers of the dataset is:

REG. No., Gender, LTP1, LTN1, LTS1, LTP2, LTN2, LT3, MT, AP, AE, AR, 1FE, 2FE, 3FE, 4FE, 5FE, 6FE

- **FIX: The first column header is REG. No., since column name cannot be separate with space in SQL, The name will be changed to REG_NO in SQL. Below is the query to create the table lab_test for this dataset.**

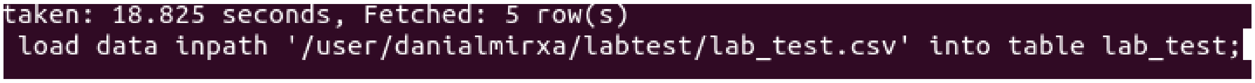
  Columns 1FE, 2FE, 3FE, 4FE, 5FE, 6FE **are started with number so they must be contained with “`”.**

Create table lab_test (REG_No varchar(20), Gender varchar(10), LTP1 float, LTN1 float, LTS1 float, LTP2 float, LTN2 float, LT3 float, MT float, AP float, AE float, AR float, `1FE` float, `2FE` float, `3FE` float, `4FE` float, `5FE` float, `6FE` float) row format delimited fields terminated by ‘,’;

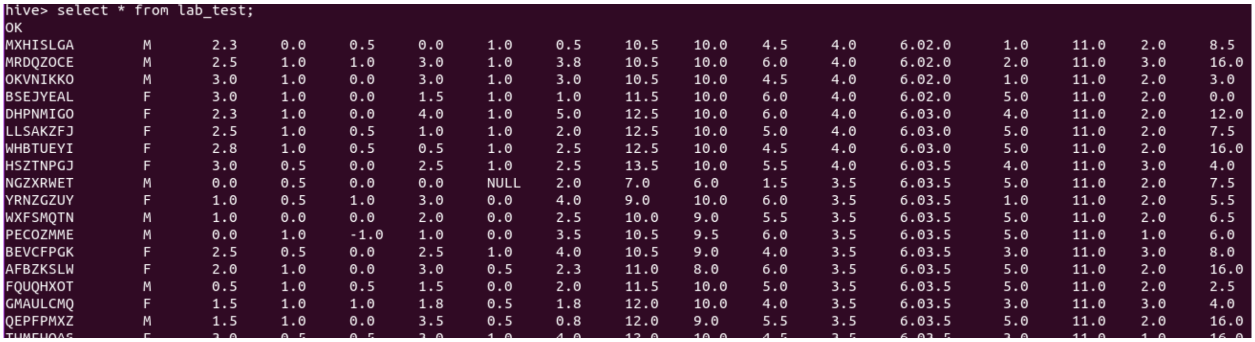


```
hive> create table lab_test (REG_No varchar(20),Gender varchar(10),LTP1 float,LTN1 float,LTS1 float,LTP2 flo
at,LTN2 float,LT3 float,MT float,AP float,AE float,AR float,`1FE` float,`2FE` float,`3FE` float,`4FE` float,
`5FE` float,`6FE` float) row format delimited fields terminated by ',';
```

- The dataset then loaded to the table



```
taken: 18.825 seconds, Fetched: 5 row(s)
 load data inpath '/user/danialmirxa/labtest/lab_test.csv' into table lab_test;
```



```
hive> select * from lab_test;
OK
MXHISLGA        M       2.3     0.0     0.5     0.0     1.0     0.5     10.5    10.0    4.5     4.0     6.02.0      1.0     11.0    2.0     8.5
MRDQZOCE        M       2.5     1.0     1.0     3.0     1.0     3.8     10.5    10.0    6.0     4.0     6.02.0      2.0     11.0    3.0     16.0
OKVNIKKO        M       3.0     1.0     0.0     3.0     1.0     3.0     10.5    10.0    4.5     4.0     6.02.0      1.0     11.0    2.0     3.0
BSEJYEAL        F       3.0     1.0     0.0     1.5     1.0     1.0     11.5    10.0    6.0     4.0     6.02.0      5.0     11.0    2.0     0.0
DHPNMIGO        F       2.3     1.0     0.0     4.0     1.0     5.0     12.5    10.0    6.0     4.0     6.03.0      4.0     11.0    2.0     12.0
LLSAKZFJ        F       2.5     1.0     0.5     1.0     1.0     2.0     12.5    10.0    5.0     4.0     6.03.0      5.0     11.0    2.0     7.5
WHBTUEYI        F       2.8     1.0     0.5     0.5     1.0     2.5     12.5    10.0    4.5     4.0     6.03.0      5.0     11.0    2.0     16.0
HSZTNPGJ        F       3.0     0.5     0.0     2.5     1.0     2.5     13.5    10.0    5.5     4.0     6.03.5      4.0     11.0    3.0     4.0
NGZXRWET        M       0.0     0.5     0.0     0.0     NULL    2.0     7.0     6.0     1.5     3.5     6.03.5      5.0     11.0    2.0     7.5
YRNZGZUY        F       1.0     0.5     1.0     3.0     0.0     4.0     9.0     10.0    6.0     3.5     6.03.5      1.0     11.0    2.0     5.5
WXFSMQTN        M       1.0     0.0     0.0     2.0     0.0     2.5     10.0    9.0     5.5     3.5     6.03.5      5.0     11.0    2.0     6.5
PECOZMME        M       0.0     1.0     -1.0    1.0     0.0     3.5     10.5    9.5     6.0     3.5     6.03.5      5.0     11.0    1.0     6.0
BEVCFPGK        F       2.5     0.5     0.0     2.5     1.0     4.0     10.5    9.0     4.0     3.5     6.03.5      3.0     11.0    3.0     8.0
AFBZKSLW        F       2.0     1.0     0.0     3.0     0.5     2.3     11.0    8.0     6.0     3.5     6.03.5      5.0     11.0    2.0     16.0
FQUQHXOT        M       0.5     1.0     0.5     1.5     0.0     2.0     11.5    10.0    5.0     3.5     6.03.5      5.0     11.0    2.0     2.5
GMAULCMQ        F       1.5     1.0     1.0     1.8     0.5     1.8     12.0    10.0    4.0     3.5     6.03.5      3.0     11.0    3.0     4.0
QEPFPMXZ        M       1.5     1.0     0.0     3.5     0.5     0.8     12.0    9.0     5.5     3.5     6.03.5      5.0     11.0    2.0     16.0
```

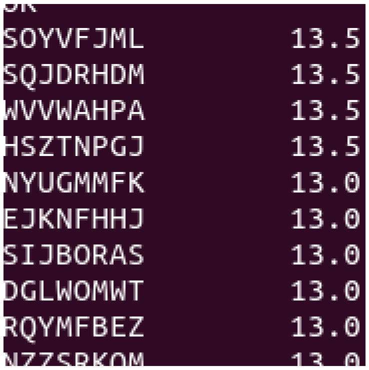
2. Question 2

a. 10 students that have the highest MT score

The query below select the REG_NO of student and MT score, then order it by MT score in descending order and limit by only the top 10 results

select REG_NO, MT from lab_test order by MT desc limit 10;

```
 select REG_NO, MT from lab_test order by MT desc limit 10;
```

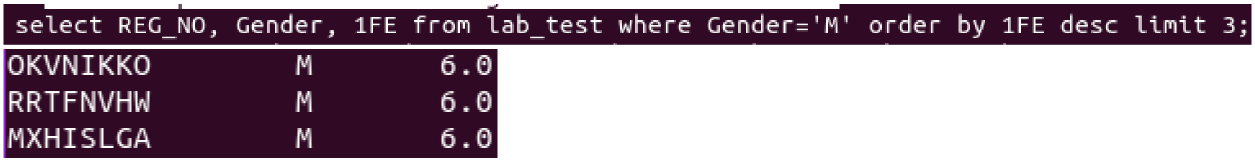


```
SOYVFJML        13.5
SQJDRHDM        13.5
WVVWAHPA        13.5
HSZTNPGJ        13.5
NYUGMMFK        13.0
EJKNFHHJ        13.0
SIJBORAS        13.0
DGLWOMWT        13.0
RQYMFBEZ        13.0
NZZSRKOM        13.0
```

b. 3 male and 3 female students that have the lowest 1FE score.

Queries below select REG_NO of students, the Gender and the 1FE score, the first query filter by Gender=’M’ for male and order by 1FE score in descending order and limit to top 3 results.

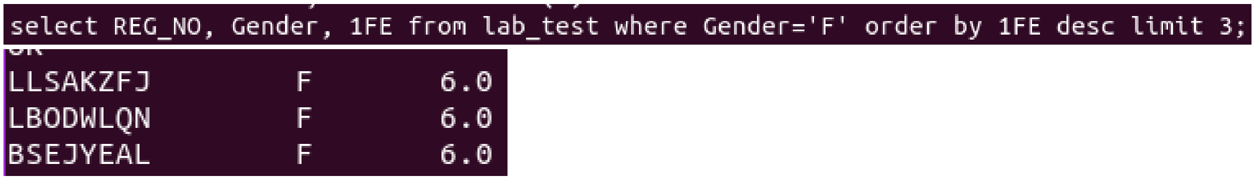
select REG_NO, Gender, 1FE from lab_test where Gender=’M’ order by 1FE desc limit 3;



```
select REG_NO, Gender, 1FE from lab_test where Gender='M' order by 1FE desc limit 3;
OKVNIKKO        M       6.0
RRTFNVHW        M       6.0
MXHISLGA        M       6.0
```

The second query is similar to the first one except filtered by Gender = ‘F’

select REG_NO, Gender, 1FE from lab_test where Gender=’F’ order by 1FE desc limit 3;

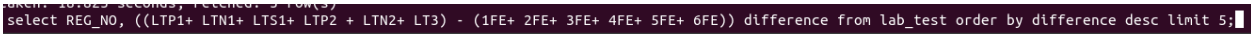


```
select REG_NO, Gender, 1FE from lab_test where Gender='F' order by 1FE desc limit 3;
LLSAKZFJ        F       6.0
LBODWLQN        F       6.0
BSEJYEAL        F       6.0
```

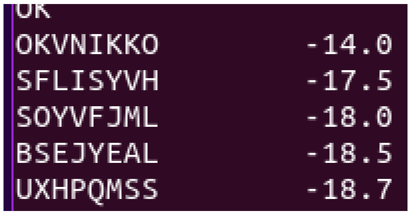
c. 5 students that shows the biggest difference from LT (LTP1, LTN1, LTS1, LTP2, LTN2, LT3) score to FE (1FE, 2FE, 3FE, 4FE, 5FE, 6FE) score.

Query below select REG_NO and adds LTP1, LTN1, LTS1, LTP2, LTN2, LT3 as first result and 1FE, 2FE, 3FE, 4FE, 5FE, 6FE as second result. Then it subtracts the first result with the second result and the result is named as ‘difference’. Then the table is ordered by ‘difference’ in descending order and limited to show top 5 results.

Select REG_NO ((LTP1+ LTN1+ LTS1+ LTP2+ LTN2+ LT3) - (1FE+ 2FE+ 3FE+ 4FE,+ 5FE+ 6FE)) difference order by difference desc limit 5;



```
select REG_NO, ((LTP1+ LTN1+ LTS1+ LTP2 + LTN2+ LT3) - (1FE+ 2FE+ 3FE+ 4FE+ 5FE+ 6FE)) difference from lab_test order by difference desc limit 5;
```



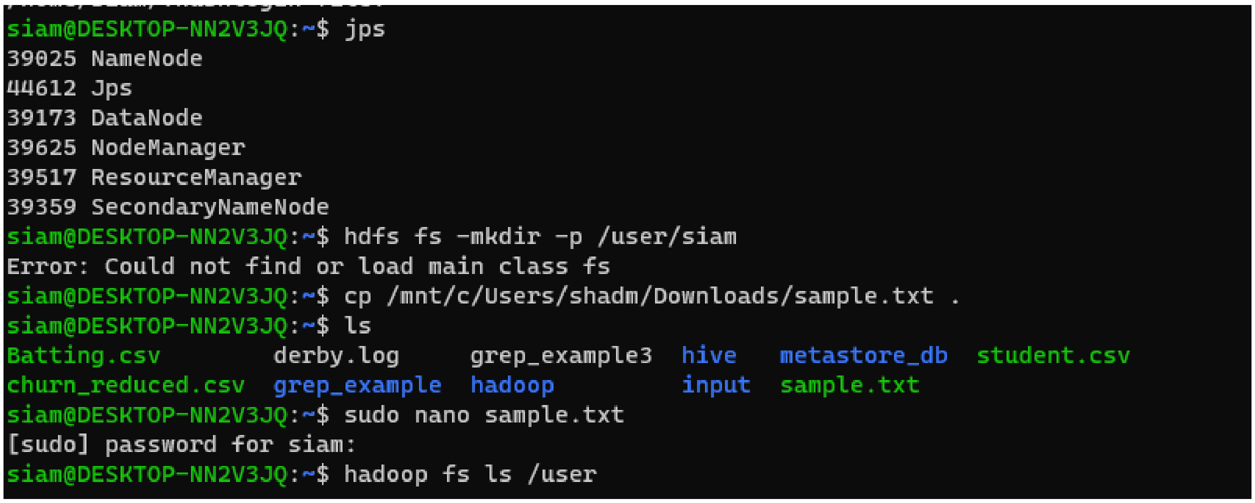
```
OK
OKVNIKKO        -14.0
SFLISYVH        -17.5
SOYVFJML        -18.0
BSEJYEAL        -18.5
UXHPQMSS        -18.7
```

Part 2

1. Import two sets of text (in .txt) from the Spectrum page to HDFS:

We have first downloaded the files set_04.txt and set_07.txt and combined them in one txt file called sample.txt. Then we moved the sample.txt file in the ubuntu machine running as wsl 2.

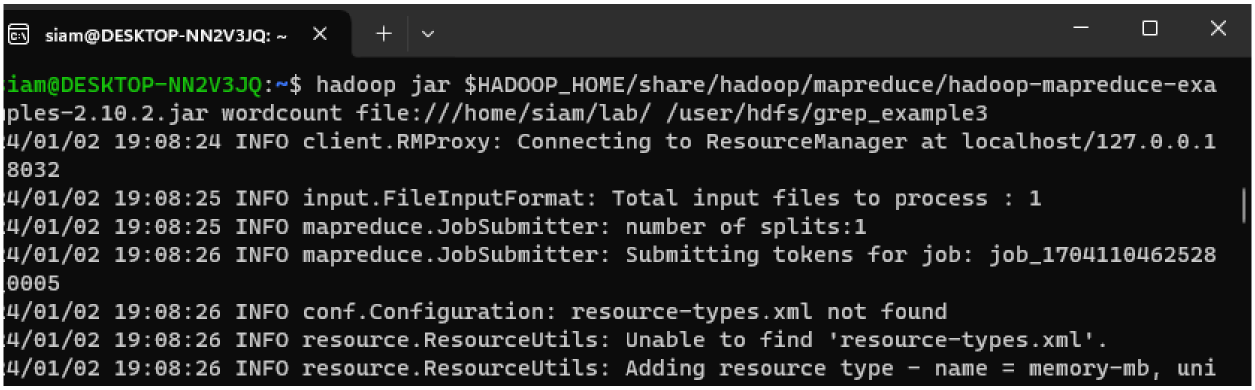
We also started hadoop with command **`start-all.sh`** and checked if all the nodes are running with command **`jps`**.



```
siam@DESKTOP-NN2V3JQ:~$ jps
39025 NameNode
44612 Jps
39173 DataNode
39625 NodeManager
39517 ResourceManager
39359 SecondaryNameNode
siam@DESKTOP-NN2V3JQ:~$ hdfs fs -mkdir -p /user/siam
Error: Could not find or load main class fs
siam@DESKTOP-NN2V3JQ:~$ cp /mnt/c/Users/shadm/Downloads/sample.txt .
siam@DESKTOP-NN2V3JQ:~$ ls
Batting.csv        derby.log     grep_example3  hive    metastore_db  student.csv
churn_reduced.csv  grep_example  hadoop         input   sample.txt
siam@DESKTOP-NN2V3JQ:~$ sudo nano sample.txt
[sudo] password for siam:
siam@DESKTOP-NN2V3JQ:~$ hadoop fs ls /user
```

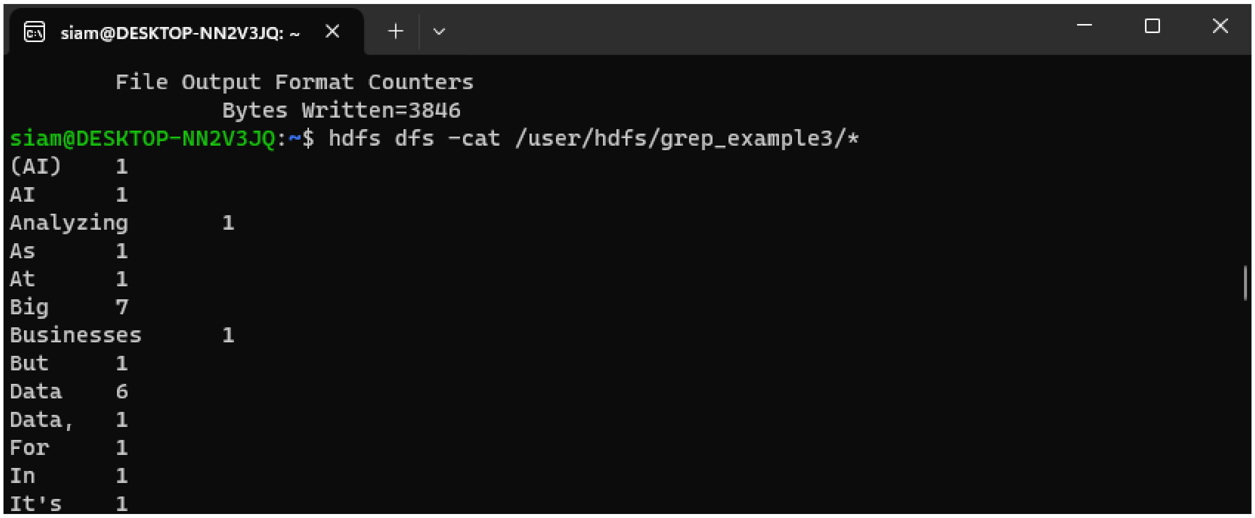
2. Run a word count program using Hadoop MapReduce concept to count the word occurrence of the imported texts as in step 1. Save the results in HDFS.

To run the **wordcount** function (of hadoop mapreduce). We used the following command and saved the word counts in the directory inside HDFS named **`/user/hdfs/grep_example3`**.



```
iam@DESKTOP-NN2V3JQ:~$ hadoop jar $HADOOP_HOME/share/hadoop/mapreduce/hadoop-mapreduce-exa
ples-2.10.2.jar wordcount file:///home/siam/lab/ /user/hdfs/grep_example3
4/01/02 19:08:24 INFO client.RMProxy: Connecting to ResourceManager at localhost/127.0.0.1
8032
4/01/02 19:08:25 INFO input.FileInputFormat: Total input files to process : 1
4/01/02 19:08:25 INFO mapreduce.JobSubmitter: number of splits:1
4/01/02 19:08:26 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1704110462528
0005
4/01/02 19:08:26 INFO conf.Configuration: resource-types.xml not found
4/01/02 19:08:26 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
4/01/02 19:08:26 INFO resource.ResourceUtils: Adding resource type - name = memory-mb, uni
```

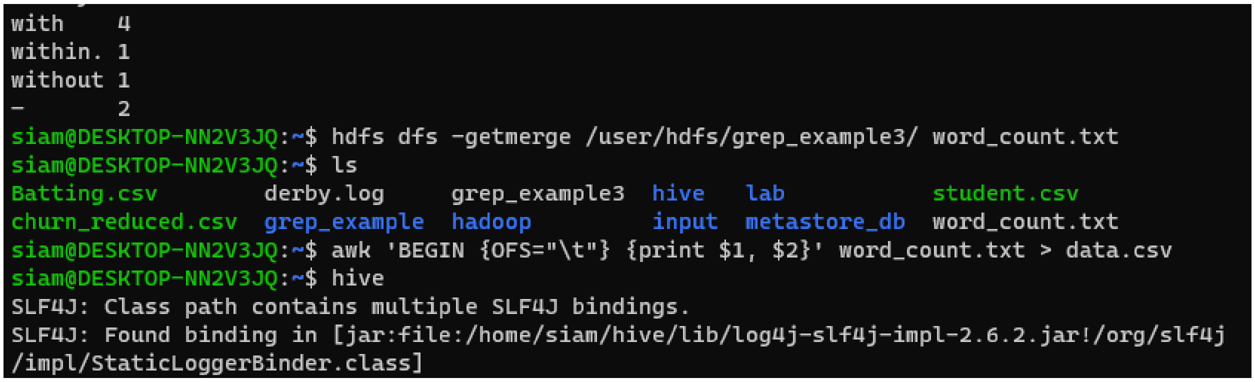
The picture below shows the result of the **wordcount** function.



```
        File Output Format Counters
                Bytes Written=3846
siam@DESKTOP-NN2V3JQ:~$ hdfs dfs -cat /user/hdfs/grep_example3/*
(AI)    1
AI      1
Analyzing       1
As      1
At      1
Big     7
Businesses      1
But     1
Data    6
Data,   1
For     1
In      1
It's    1
```

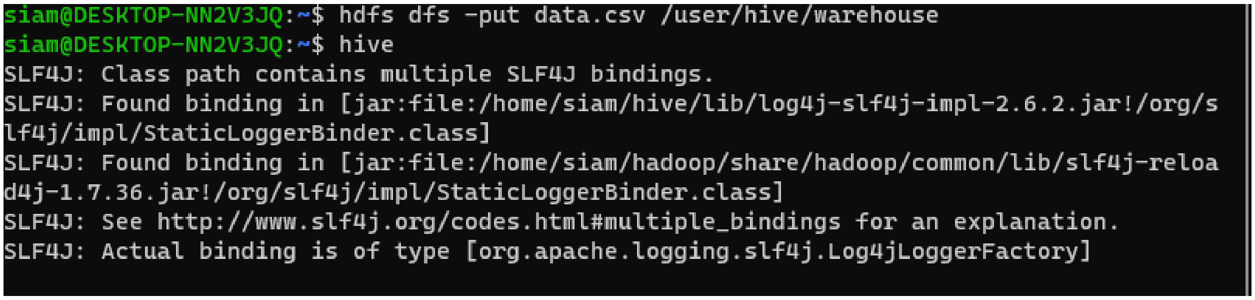
3. Importing the results of step 2 in Hive:

To import the result we first merged the output in the wordcount.txt file and created a tab separated csv file called data.csv.



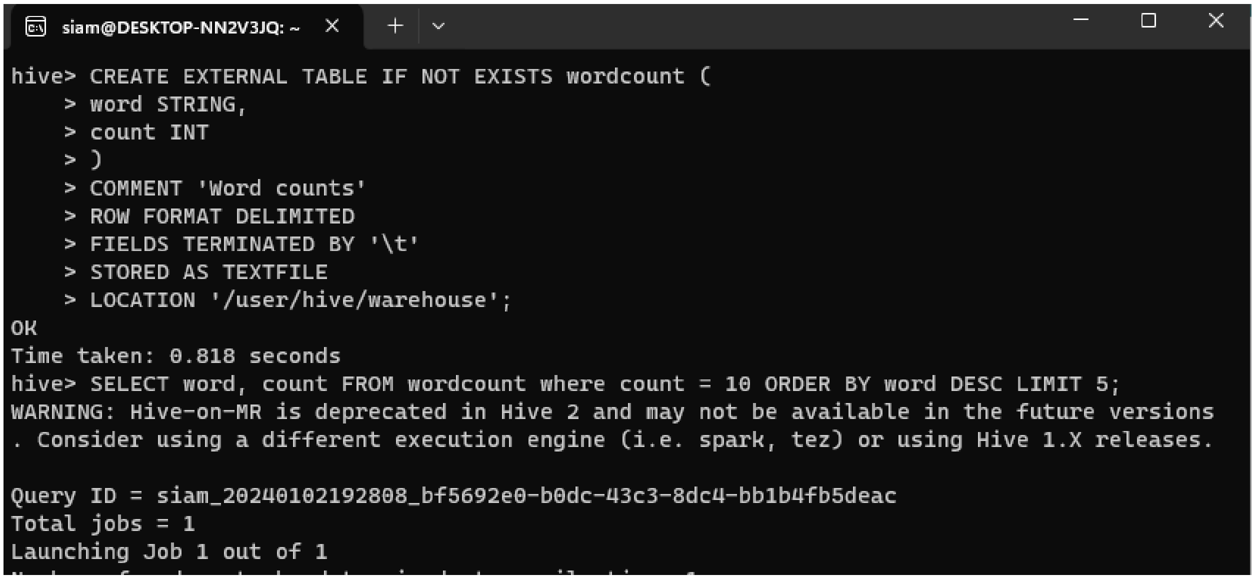
```
with    4
within. 1
without 1
-       2
siam@DESKTOP-NN2V3JQ:~$ hdfs dfs -getmerge /user/hdfs/grep_example3/ word_count.txt
siam@DESKTOP-NN2V3JQ:~$ ls
Batting.csv         derby.log     grep_example3  hive    lab            student.csv
churn_reduced.csv   grep_example  hadoop         input   metastore_db   word_count.txt
siam@DESKTOP-NN2V3JQ:~$ awk 'BEGIN {OFS="\t"} {print $1, $2}' word_count.txt > data.csv
siam@DESKTOP-NN2V3JQ:~$ hive
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/siam/hive/lib/log4j-slf4j-impl-2.6.2.jar!/org/slf4j
/impl/StaticLoggerBinder.class]
```

Before creating the table in **hive**, we imported the csv file in **HDFS**. And then we started hive.



```
siam@DESKTOP-NN2V3JQ:~$ hdfs dfs -put data.csv /user/hive/warehouse
siam@DESKTOP-NN2V3JQ:~$ hive
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/siam/hive/lib/log4j-slf4j-impl-2.6.2.jar!/org/s
lf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/siam/hadoop/share/hadoop/common/lib/slf4j-reloa
d4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
```

Next we created the table with the following command:

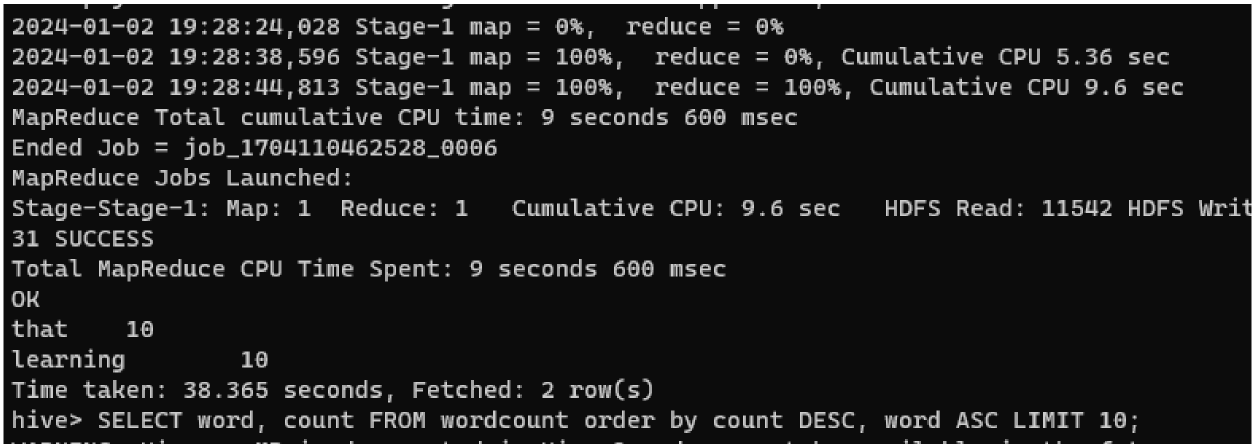


```
hive> CREATE EXTERNAL TABLE IF NOT EXISTS wordcount (
    > word STRING,
    > count INT
    > )
    > COMMENT 'Word counts'
    > ROW FORMAT DELIMITED
    > FIELDS TERMINATED BY '\t'
    > STORED AS TEXTFILE
    > LOCATION '/user/hive/warehouse';
OK
Time taken: 0.818 seconds
hive> SELECT word, count FROM wordcount where count = 10 ORDER BY word DESC LIMIT 5;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions
. Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.

Query ID = siam_20240102192808_bf5692e0-b0dc-43c3-8dc4-bb1b4fb5deac
Total jobs = 1
Launching Job 1 out of 1
```

Running the query:

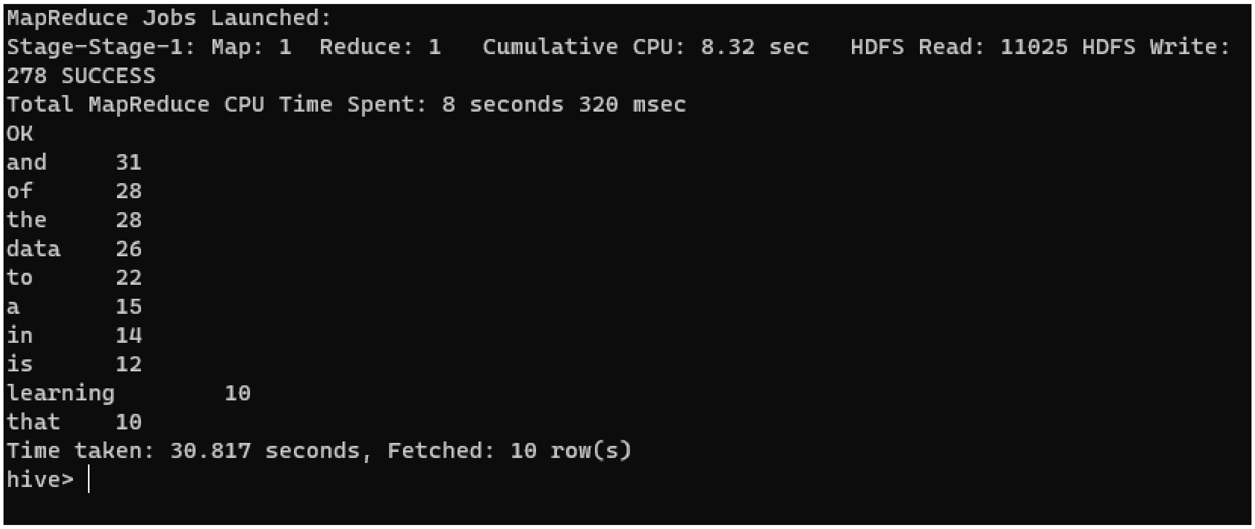
We run the following command to print the results:

a. **SELECT word, count FROM wordcount where count = 10 ORDER BY word DESC LIMIT 5;**



```
2024-01-02 19:28:24,028 Stage-1 map = 0%,  reduce = 0%
2024-01-02 19:28:38,596 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 5.36 sec
2024-01-02 19:28:44,813 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 9.6 sec
MapReduce Total cumulative CPU time: 9 seconds 600 msec
Ended Job = job_1704110462528_0006
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 9.6 sec   HDFS Read: 11542 HDFS Writ
31 SUCCESS
Total MapReduce CPU Time Spent: 9 seconds 600 msec
OK
that    10
learning        10
Time taken: 38.365 seconds, Fetched: 2 row(s)
hive> SELECT word, count FROM wordcount order by count DESC, word ASC LIMIT 10;
```

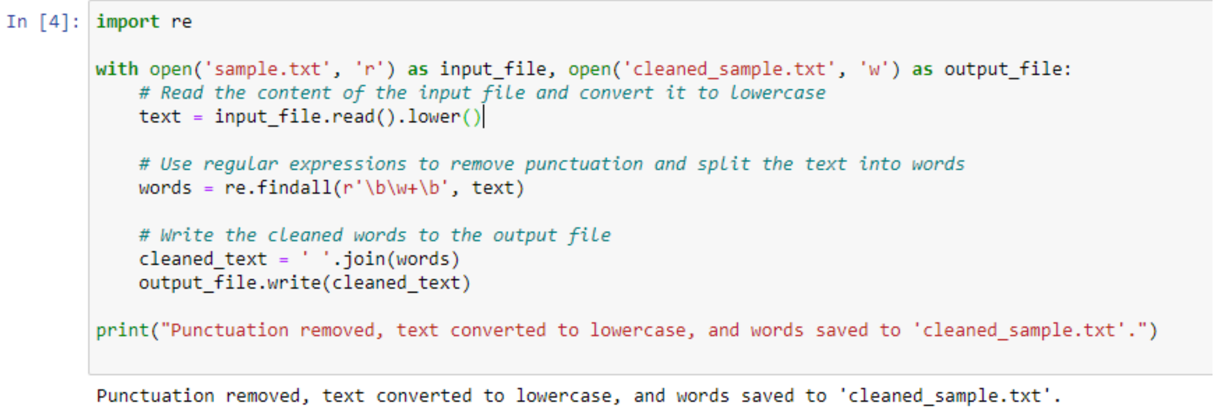
b. SELECT word, count FROM wordcount order by count DESC, word ASC LIMIT 10;



```
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 8.32 sec   HDFS Read: 11025 HDFS Write:
278 SUCCESS
Total MapReduce CPU Time Spent: 8 seconds 320 msec
OK
and     31
of      28
the     28
data    26
to      22
a       15
in      14
is      12
learning        10
that    10
Time taken: 30.817 seconds, Fetched: 10 row(s)
hive>
```

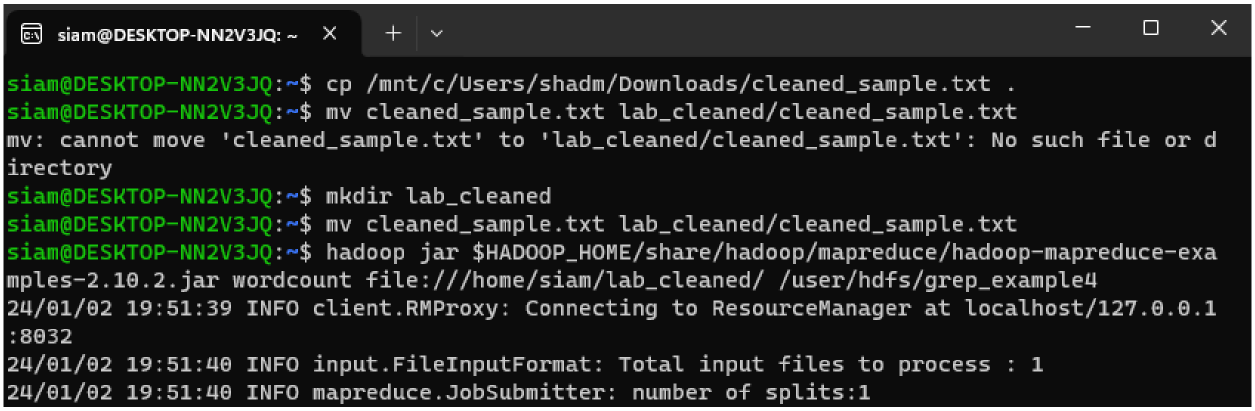
4. Applying preprocessing steps and cleaning the imported text in question 1:

To preprocess the text file we used python. We removed all the punctuations and converted all the words to small case letters and saved them to cleaned_sample.txt. Below is the script.



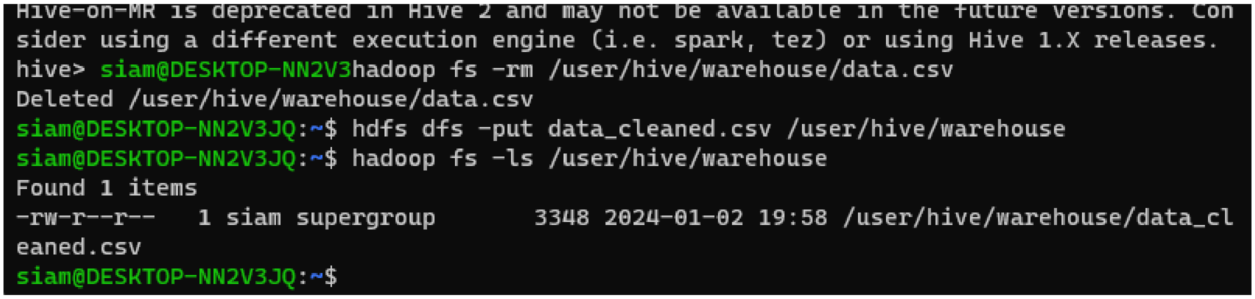
```
In [4]: import re

        with open('sample.txt', 'r') as input_file, open('cleaned_sample.txt', 'w') as output_file:
            # Read the content of the input file and convert it to lowercase
            text = input_file.read().lower()

            # Use regular expressions to remove punctuation and split the text into words
            words = re.findall(r'\b\w+\b', text)

            # Write the cleaned words to the output file
            cleaned_text = ' '.join(words)
            output_file.write(cleaned_text)

        print("Punctuation removed, text converted to lowercase, and words saved to 'cleaned_sample.txt'.")

        Punctuation removed, text converted to lowercase, and words saved to 'cleaned_sample.txt'.
```

After that we copied the cleaned file in the ubuntu vm and run **hdfs** wordcount function and pasted the result in **grep_example4** folder.



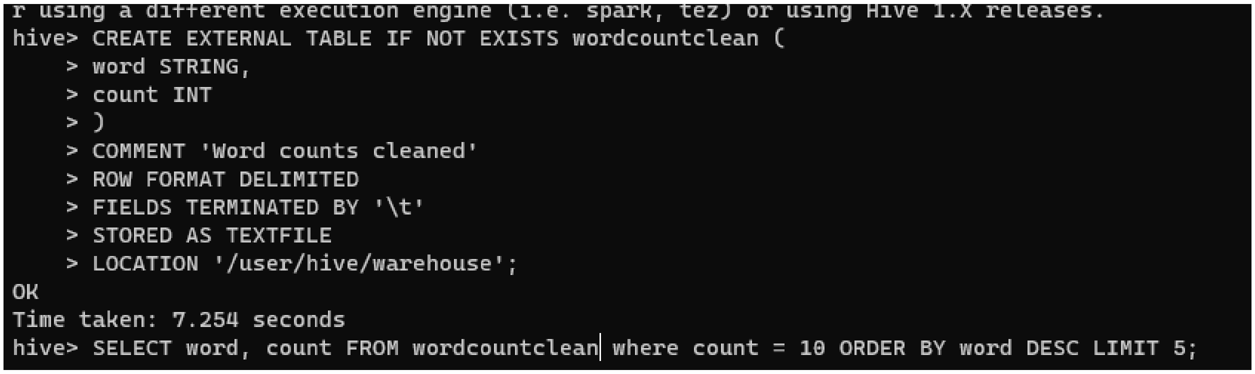
```
siam@DESKTOP-NN2V3JQ:~$ cp /mnt/c/Users/shadm/Downloads/cleaned_sample.txt .
siam@DESKTOP-NN2V3JQ:~$ mv cleaned_sample.txt lab_cleaned/cleaned_sample.txt
mv: cannot move 'cleaned_sample.txt' to 'lab_cleaned/cleaned_sample.txt': No such file or d
irectory
siam@DESKTOP-NN2V3JQ:~$ mkdir lab_cleaned
siam@DESKTOP-NN2V3JQ:~$ mv cleaned_sample.txt lab_cleaned/cleaned_sample.txt
siam@DESKTOP-NN2V3JQ:~$ hadoop jar $HADOOP_HOME/share/hadoop/mapreduce/hadoop-mapreduce-exa
mples-2.10.2.jar wordcount file:///home/siam/lab_cleaned/ /user/hdfs/grep_example4
24/01/02 19:51:39 INFO client.RMProxy: Connecting to ResourceManager at localhost/127.0.0.1
:8032
24/01/02 19:51:40 INFO input.FileInputFormat: Total input files to process : 1
24/01/02 19:51:40 INFO mapreduce.JobSubmitter: number of splits:1
```

Then we removed the old csv file and added the new csv file in **HDFS** for loading in hive.



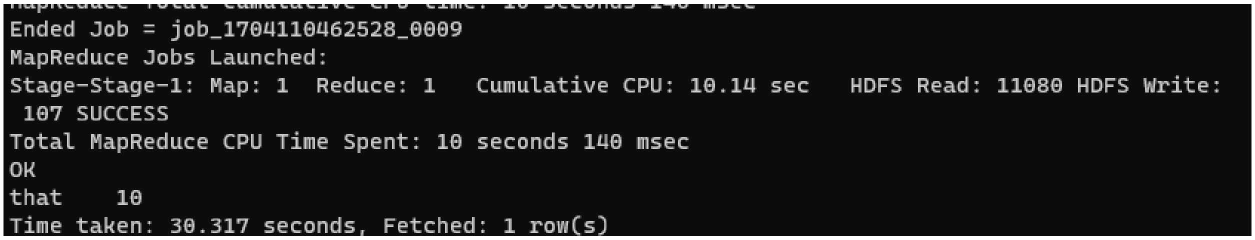
```
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Con
sider using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
hive> siam@DESKTOP-NN2V3hadoop fs -rm /user/hive/warehouse/data.csv
Deleted /user/hive/warehouse/data.csv
siam@DESKTOP-NN2V3JQ:~$ hdfs dfs -put data_cleaned.csv /user/hive/warehouse
siam@DESKTOP-NN2V3JQ:~$ hadoop fs -ls /user/hive/warehouse
Found 1 items
-rw-r--r--   1 siam supergroup       3348 2024-01-02 19:58 /user/hive/warehouse/data_cl
eaned.csv
siam@DESKTOP-NN2V3JQ:~$
```

Next we create the table with name `**wordcountclean**` in hive.



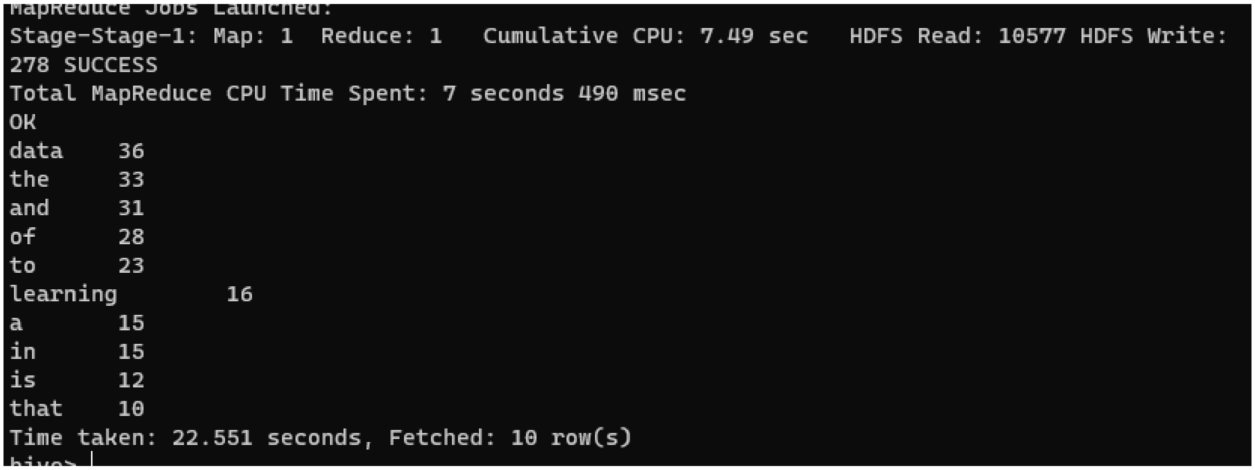
```
r using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
hive> CREATE EXTERNAL TABLE IF NOT EXISTS wordcountclean (
    > word STRING,
    > count INT
    > )
    > COMMENT 'Word counts cleaned'
    > ROW FORMAT DELIMITED
    > FIELDS TERMINATED BY '\t'
    > STORED AS TEXTFILE
    > LOCATION '/user/hive/warehouse';
OK
Time taken: 7.254 seconds
hive> SELECT word, count FROM wordcountclean where count = 10 ORDER BY word DESC LIMIT 5;
```

After running the query we got the following output :



```
MapReduce Total cumulative CPU time: 10 seconds 140 msec
Ended Job = job_1704110462528_0009
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 10.14 sec   HDFS Read: 11080 HDFS Write:
 107 SUCCESS
Total MapReduce CPU Time Spent: 10 seconds 140 msec
OK
that    10
Time taken: 30.317 seconds, Fetched: 1 row(s)
```

And for the next query we got the following result:



```
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 7.49 sec   HDFS Read: 10577 HDFS Write:
278 SUCCESS
Total MapReduce CPU Time Spent: 7 seconds 490 msec
OK
data    36
the     33
and     31
of      28
to      23
learning        16
a       15
in      15
is      12
that    10
Time taken: 22.551 seconds, Fetched: 10 row(s)
hive>
```

So, we can see in both the cases we have received a bit different result after cleaning the data.

Comparison:

Uncleaned Data:

Query 1:

- Word "that" appeared 10 times.
- Word “learning” appeared 10 times.

Query 2:

The top 10 most frequent words included "and," "of," "the," "data," "to," "a," "in," "is," "learning," and "that."

Cleaned Data:

Query 1:

- Word "that" appeared 10 times.

Query 2:

- The top 10 most frequent words included "data," "the," "and," "of," "to," "learning," "a," "in," "is," and "that."

Comment: The overall frequency of words changed in Query 2 for the cleaned data. This change suggests that removing punctuation and converting all words to lowercase affected the word count results.

In summary, cleaning and preprocessing the text data by removing punctuation and converting to lowercase can affect the word frequency results and may lead to faster query execution times, as shown in the provided results.