start-all.sh

hadoop fs -mkdir /mahendra

cat >sample

hadoop fs -put '/home/cseds/sample' /mahendra

hadoop fs -rm -r /mahendra

$ hadoop fs -rm -r /mahendra/output2

**3-A**

hadoop dfsadmin -safemode leave

**1.Create a new folder wordcount and copy the WordCount.java file in it.**

Create the sub folder inputdata in wordcount folder, create a text file and enter your input data.(demo.txt)

Create another subfolder, classfiles to store the java class files.

**2.$export HADOOP\_CLASSPATH=$(hadoop classpath**)

Ensure the class path set correctly.

$echo $HADOOP\_CLASSPATH

If it is not printing the output then type $sudo nano ~/.bshrc

it asks password enter

then at last the file append this path export HADOOP\_CLASSPATH=$(hadoop\_classpath)

check $echo $HADOOP\_CLASSPATH

**3.Create a directory in HDFS**

$hadoop fs -mkdir /WC #give any name for your folder

Create a sub folder in side WC to store input data.

$hadoop fs -mkdir /WC/input

Check the folders in Hadoop

Goto browser and check folder in http://localhost:9870/

Upload the input file to Hadoop input folder.

$hadoop fs -put '/home/cseds/wordcount/inputfolder/demo.txt' /WC/input

Change the current directory to wordcount directory

$cd \

$cd wordcount

$javac -classpath ${HADOOP\_CLASSPATH} -d /home/wordcount/classfiles /home/wordcount/WordCount.java

$jar -cvf wcj.jar -C classfiles .

$hadoop jar /home/wordcount/wcj.jar WordCount /WC/input /WC/output

$hadoop dfs -cat /WC/output/\*

**3.B**

ssh localhost

start-all.sh

create exp3b folder

create inputfile folder give sample.txt

create classfiles folder

create java file and save

export hp=$(hadoop classpath)

to check hadoop class path

$echo {hp}

javac -classpath ${hp} -d /home/cseds/exp3b/classfiles /home/cseds/exp3b/WordSCount.java

jar -cvf /home/cseds/exp3b/wordScount.jar -C /home/cseds/exp3b/classfiles/ .

cd exp3b

hadoop fs -mkdir /wcc

hadoop fs -mkdir /wcc/input

hadoop fs -put '/home/cseds/exp3b/sample.txt' /wcc/input

hadoop jar '/home/cseds/exp3b/wordScount.jar' WordSCount /wcc/input /wcc/output mahendra

**4 TH EXP**

create SWE folder

StopWordElimination.java

demo.txt

classfiles folder(empty)

stopwords.txt

IN the StopWordElimination java file

try to change the path like "/home/cseds/swe/stopwords.txt" in the stopwords.txt

#Execute the following cmds

start-all.sh

export hp=$(hadoop classpath)

echo $hp

hadoop dfsadmin -safemode leave

javac -classpath ${hp} -d /home/cseds/swe/classfiles /home/cseds/swe/StopWordElimination.java

cd swe

jar -cvf sw.jar -C /home/cseds/swe/classfiles/ .

hadoop fs -mkdir /mahendra

hadoop fs -mkdir /mahendra/input

hadoop fs -put '/home/cseds/swe/demo.txt' /mahendra/input

hadoop jar '/home/cseds/swe/sw.jar' StopWordElimination /mahendra/input /mahendra/output

hadoop fs -cat /mahendra/output/\*

**5 TH EXP**

create tempp folder

TemperatureStats.java

wet.csv(if there is any space in the file name try to rename it properly)

classfiles folder(empty)

#Execute the following cmds

start-all.sh

export hp=$(hadoop classpath)

echo $hp #properly check

hadoop dfsadmin -safemode leave

javac -classpath ${hp} -d /home/cseds/tempp/classfiles /home/cseds/tempp/TemperatureStats.java

cd tempp

jar -cvf temp.jar -C /home/cseds/tempp/classfiles/ .

hadoop fs -mkdir /temperature

hadoop fs -mkdir /temperature/input

hadoop fs -put '/home/cseds/temperature/wet.csv' /temperature/input

hadoop jar '/home/cseds/tempp/temp.jar' TemperatureStats /temperature/input /temperature/output

hadoop fs -cat /temperature/output/\*

**6-th EXP**

javac -classpath ${hp} -d /home/cseds/exp6/classfiles /home/cseds/exp6/SalesAnalysis.java

jar -cvf home/cseds/exp6/sales.jar -C /home/cseds/exp6/classfiles/ .

cd exp6

hadoop fs -mkdir /exp6

hadoop fs -mkdir /exp6/input

hadoop fs -put '/home/cseds/exp6/data.txt' /exp6/input

hadoop jar '/home/cseds/exp6/sales.jar' SalesAnalysis /exp6/input /exp6/output

#NOT COMPLETE

**7-TH EXP:**

exp-7:

Download the pig latest version in the ubuntu

change the name of the folder as 'pig' after extracting

move that file into home from downloads

sudo nano .bashrc

source .bashrc

pig --version

pig -x local

so it shows like

grunt>

text file

101,AAA,99,100,cricket

102,BBB,45,55,football

103,CCC,90,80,chess

104,DDD,79,89,cricket

105,EEE,99,99,chess

#load the text file

grunt> A = LOAD 'sports.txt' using PigStorage(',') as (rollno:int, name:chararray, m1:int, m2:int, sport:chararray);

grunt> dump A

grunt> B = order A by m1 desc;

grunt> dump B

output:

(105,EEE,99,99,chess)

(101,AAA,99,100,cricket)

(103,CCC,90,80,chess)

(104,DDD,79,89,cricket)

(102,BBB,45,55,football)

grunt> D = group A by name;

grunt> dump D

grunt> D = group A by name;

grunt> dump D

(AAA,{(101,AAA,99,100,cricket)})

(BBB,{(102,BBB,45,55,football)})

(CCC,{(103,CCC,90,80,chess)})

(DDD,{(104,DDD,79,89,cricket)})

(EEE,{(105,EEE,99,99,chess)})

grunt> F= filter A by (m2>85);

grunt> dump F

(101,AAA,99,100,cricket)

(104,DDD,79,89,cricket)

(105,EEE,99,99,chess)

grunt> G = foreach A generate (rollno,name);

grunt> dump G

((101,AAA))

((102,BBB))

((103,CCC))

((104,DDD))

((105,EEE))