**ROLL NO 17**

**Doc.csv**

1,500,BANGLORE,RAVI

2,1000,PATNA,SANJAY

3,500,HYDERABAD,RANVIJAY

4,800,GUJARAT,AKASH

5,1200,ASSAM,DHIREN

**Cloudera Terminal**

**Command : -** hdfs dfs -mkdir /Newinput

**Command : -** hdfs dfs -put /home/cloudera/Desktop/Doc.csv Newinput

**Hive Terminal**

**Command : -** create database New;

**Command : -** show databases;

**Command : -** use new;

**Command : -** create table Sales(

sale\_id int,

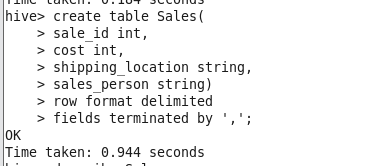
cost int,

shipping\_location string,

sales\_person string)

row format delimited

fields terminated by ',';



**Command : -** describe Sales;

**Command : -** LOAD DATA LOCAL INPATH '/home/cloudera/Desktop/Doc.csv' INTO TABLE Sales;

**Command : -** select \* from sales;

**ROLL NO 18**

**record.txt**

Aisha,Mumbai,8490709893

Suhana,Pune,9070989384

Rishi,Jharkand,8490989370

Suraj,Banglore,907049893

Vishnu,Delhi,84907093

**Cloudera terminal**

**Command: -** hdfs dfs -mkdir /BDAV

**Command: -** hdfs dfs -mkdir /BDAV/pigfiles

**Command: -** hdfs dfs -ls /BDAV

**Command: -** hdfs dfs -put /home/cloudera/Desktop/record.txt /BDAV/pigfiles

**Pig Shell**

**Command: -** A= LOAD '/home/cloudera/Desktop/record.txt' USING PigStorage(',') AS (names:chararray,city:chararray,phone:chararray);

**Command: -** DESCRIBE A;

**Command: -** B= LOAD '/home/cloudera/Desktop/record.txt' USING PigStorage(',') AS (names:chararray,city:chararray);

**Command: -** Dump B;

**Command: -** group\_by = GROUP A by city;

**Command: -** Dump group\_by

**ROLL NO 19**

**data.txt**

Water Jug

Water Jug

Water Jug

Hill Climbing

Hill Climbing

Hill Climbing

A star

A star

A star

AO star

AO star

AO star

AO star

AO star

Tree

Tree

Tree

Water Jug

**Cloudera Terminal**

**Command: -** hdfs dfs -mkdir /spark

**Command: -** hdfs dfs -put /home/cloudera/Desktop/data.txt /spark

**Command: -** hdfs dfs -ls /spark

**Spark Terminal**

**reduceByKey**

**Command: -** val a1 = sc.textFile("/spark/data.txt")

**Command: -** a1.collect()

**Command: -** a1.count()

**Command: -** val splitdata=a1.flatMap(line => line.split(" "));

**Command: -** splitdata.collect()

**Command: -** val mapdata = splitdata.map(word =>(word,1))

**Command: -** mapdata.collect()

**Command: -** val reducedata = mapdata.reduceByKey(\_+\_);

**Command: -** reducedata.collect()

**aggregateByKey**

**Command: -** val initialCount=0;

**Command: -** val addToCounts = (n: Int, v: Int) => n+1

**Command: -** val sumPartitionCounts = (p1:Int,p2:Int)=>p1+p2

**Command: -** val countByKey=mapdata.aggregateByKey(initialCount)(addToCounts, sumPartitionCounts)

**Command: -** countByKey.collect()

**sortByKey**

**ROLL NO 22**

**Sales.csv**

1,100,10000,DELHI

2,500,50000,HARYANA

3,250,30000,PUNJAB

4,230,40000,DIU AND DAMAN

5,350,70000,KOLKATA

**Cloudera Terminal**

**Command: -** hdfs dfs -put /home/cloudera/Desktop/Sales.csv /hive

**Command: -** hdfs dfs -ls /hive

**Hive Terminal**

**Command: -** create database Sales;

**Command: -** show databases;

**Command: -** use sales;

**Command: -** create table Sales(

sale\_id int,

units\_sold int,

profit int,

shipping\_location string)

row format delimited

fields terminated by ',';

**Command: -** describe Sales;

**Command: -** LOAD DATA LOCAL INPATH '/home/cloudera/Desktop/Sales.csv' INTO TABLE Sales

**Command: -** select \* from Sales;

**Command: -** select min(profit) from Sales;

**Command: -** select max(profit) from Sales;