VISVESVARAYA TECHNOLOGICAL UNIVERSITY
"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

BIG DATA ANALYTICS
(20CS6PEBDA)

Submitted by

S R POOJA
(1BM19CS135)

in partial fulfillment for the award of the degree of
BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING

B.M.S. COLLEGE OF ENGINEERING
(Autonomous Institution under VTU)
BENGALURU-560019
May-2022 to July-2022

B. M. S. College of Engineering,
Bull Temple Road, Bangalore 560019
(Affiliated To Visvesvaraya Technological University, Belgaum)
Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the Lab work entitled "**BIG DATA ANALYTICS**" was carried out by **S R POOJA (1BM19CS135),** who is bonafide student of **B. M. S. College of Engineering.** It is in partial fulfillment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum during the year 2022. The Lab report has been approved as it satisfies the academic requirements in respect of the course **BIG DATA ANALYTICS (20CS6PEBDA)** work prescribed for the said degree.

Name of the Lab-In charge   ANTARA   ROY
                                                                              CHOUDHURY

Designation   Assistant Professor
Department of CSE   Department of CSE
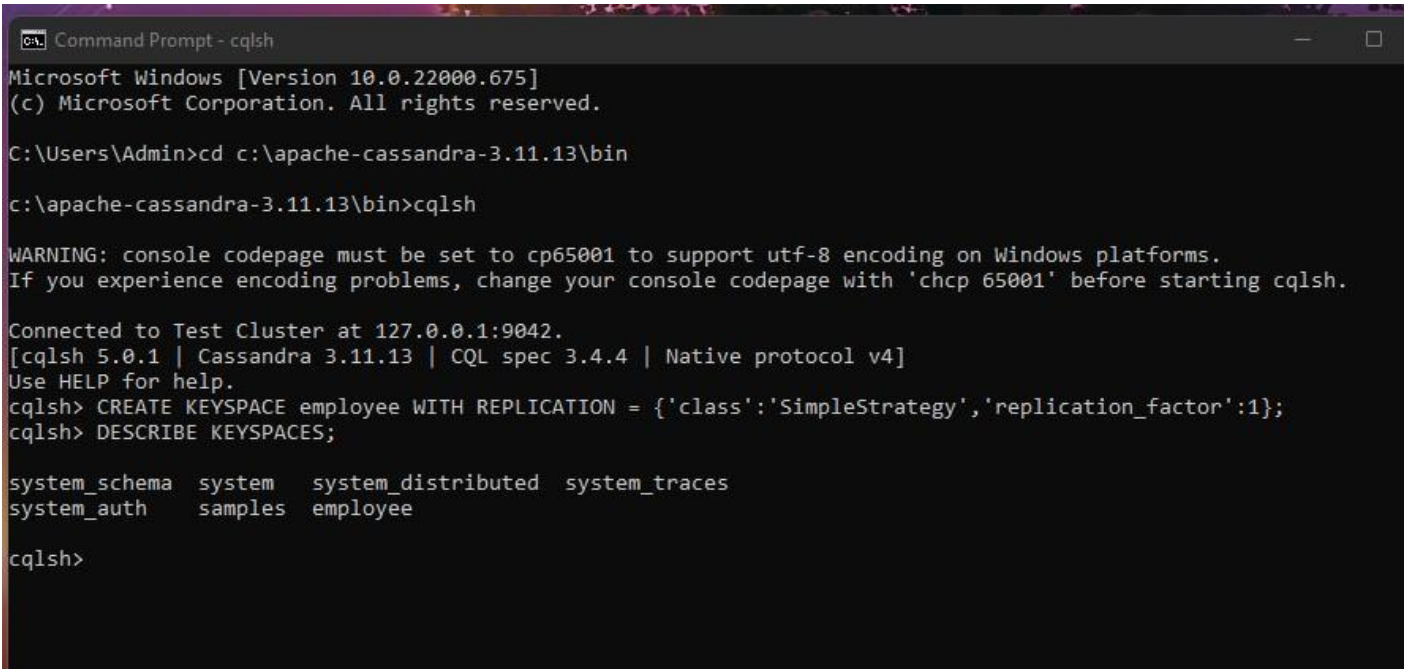BMSCE, Bengaluru   BMSCE, Bengaluru

`

Index Sheet

CO1	Apply the concept of NoSQL, Hadoop or Spark for a given task
CO2	Analyze the Big Data and obtain insight using data analytics mechanisms.
CO3	Design and implement Big data applications by applying NoSQL, Hadoop or Spark

Cassandra Lab Program 1: -

Perform the following DB operations using Cassandra.
1. Create a key space by name Employee

2. Create a column family by name Employee-Info with attributes Emp_Id Primary Key, Emp_Name, Designation, Date_of_Joining, Salary, Dept_Name

```
cqlsh:employee> CREATE TABLE EMPLOYEEINFO( EMPID INT, EMPNAME TEXT, DESIGNATION TEXT, DATEOFJOINING TIMESTAMP, SAL
ARY DOUBLE, DEPTNAME TEXT, PRIMARY KEY(EMPID,SALARY));
cqlsh:employee>
```

```
cqlsh:employee> SELECT * FROM EMPLOYEEINFO;

 empid | salary | dateofjoining | deptname | designation | empname
-------+--------+---------------+----------+-------------+--------

(0 rows)
cqlsh:employee>
```

3. Insert the values into the table in batch

```
cqlsh:employee> BEGIN BATCH
            ... INSERT INTO EMPLOYEEINFO (EMPID, EMPNAME, DESIGNATION, DATEOFJOINING, SALARY, DEPTNAME)
            ... VALUES(1,'LOKESH','ASSISTANT MANAGER', '2005-04-6', 50000, 'MARKETING')
            ... INSERT INTO EMPLOYEEINFO (EMPID, EMPNAME, DESIGNATION, DATEOFJOINING, SALARY, DEPTNAME)
            ... VALUES(2,'DHEERAJ','ASSISTANT MANAGER', '2013-11-10', 30000, 'LOGISTICS')
            ... INSERT INTO EMPLOYEEINFO (EMPID, EMPNAME, DESIGNATION, DATEOFJOINING, SALARY, DEPTNAME)
            ... VALUES(3,'CHIRAG','ASSISTANT MANAGER', '2011-07-1', 115000, 'SALES')
            ... INSERT INTO EMPLOYEEINFO (EMPID, EMPNAME, DESIGNATION, DATEOFJOINING, SALARY, DEPTNAME)
            ... VALUES(4,'DHANUSH','ASSISTANT MANAGER', '2010-04-26', 75000, 'MARKETING')
            ...  INSERT INTO EMPLOYEEINFO (EMPID, EMPNAME, DESIGNATION, DATEOFJOINING, SALARY, DEPTNAME)
            ... VALUES(5,'ESHA','ASSISTANT MANAGER', '2010-04-26', 85000, 'TECHNICAL')
            ... INSERT INTO EMPLOYEEINFO (EMPID, EMPNAME, DESIGNATION, DATEOFJOINING, SALARY, DEPTNAME)
            ... VALUES(6,'FARHAN','MANAGER', '2010-04-26', 95000, 'TECHNICAL')
            ... INSERT INTO EMPLOYEEINFO (EMPID, EMPNAME, DESIGNATION, DATEOFJOINING, SALARY, DEPTNAME)
            ... VALUES(7,'JIMMY','MANAGER', '2010-04-26', 95000, 'PR')
            ... INSERT INTO EMPLOYEEINFO (EMPID, EMPNAME, DESIGNATION, DATEOFJOINING, SALARY, DEPTNAME)
            ... VALUES(121,'HARRY','REGIONAL MANAGER', '2010-04-26', 99000, 'MANAGEMENT')
            ... APPLY BATCH;
```

```
cqlsh:employee>  SELECT * FROM EMPLOYEEINFO;

 empid | salary  | dateofjoining                   | deptname   | designation       | empname
-------+---------+---------------------------------+------------+-------------------+--------
     5 |   85000 | 2010-04-25 18:30:00.000000+0000 |  TECHNICAL | ASSISTANT MANAGER |    ESHA
     1 |   50000 | 2005-04-05 18:30:00.000000+0000 |  MARKETING | ASSISTANT MANAGER |  LOKESH
     2 |   30000 | 2013-11-09 18:30:00.000000+0000 |  LOGISTICS | ASSISTANT MANAGER | DHEERAJ
     4 |   75000 | 2010-04-25 18:30:00.000000+0000 |  MARKETING | ASSISTANT MANAGER | DHANUSH
   121 |   99000 | 2010-04-25 18:30:00.000000+0000 | MANAGEMENT |  REGIONAL MANAGER |   HARRY
     7 |   95000 | 2010-04-25 18:30:00.000000+0000 |         PR |           MANAGER |   JIMMY
     6 |   95000 | 2010-04-25 18:30:00.000000+0000 |  TECHNICAL |           MANAGER |  FARHAN
     3 | 1.15e+05 | 2011-06-30 18:30:00.000000+0000 |      SALES | ASSISTANT MANAGER |  CHIRAG

(8 rows)
cqlsh:employee>
```

4. Update Employee name and Department of Emp-Id 121

```
cqlsh:employee> UPDATE EMPLOYEEINFO SET EMPNAME='HARRY', DEPTNAME='MANAGEMENT' WHERE EMPID=121 AND SALARY=99000;
cqlsh:employee>  SELECT * FROM EMPLOYEEINFO;

 empid | salary   | dateofjoining                    | deptname   | designation       | empname
-------+----------+----------------------------------+------------+-------------------+---------
     5 |    85000 | 2010-04-25 18:30:00.000000+0000  | TECHNICAL  | ASSISTANT MANAGER |   ESHA
     1 |    50000 | 2005-04-05 18:30:00.000000+0000  | MARKETING  | ASSISTANT MANAGER |  LOKESH
     2 |    30000 | 2013-11-09 18:30:00.000000+0000  | LOGISTICS  | ASSISTANT MANAGER | DHEERAJ
     4 |    75000 | 2010-04-25 18:30:00.000000+0000  | MARKETING  | ASSISTANT MANAGER | DHANUSH
   121 |    99000 | 2010-04-25 18:30:00.000000+0000  | MANAGEMENT |   REGIONAL MANAGER |   HARRY
     7 |    95000 | 2010-04-25 18:30:00.000000+0000  |         PR |           MANAGER |   JIMMY
     6 |    95000 | 2010-04-25 18:30:00.000000+0000  | TECHNICAL  |           MANAGER |  FARHAN
     3 | 1.15e+05 | 2011-06-30 18:30:00.000000+0000  |      SALES | ASSISTANT MANAGER |  CHIRAG

(8 rows)
cqlsh:employee>
```

5. Sort the details of Employee records based on salary (Note:- cql>PAGING OFF)

```
cqlsh:employee> select * from EMPLOYEEINFO where empid IN(1,2,3,4,5,6,7) ORDER BY salary DESC allow filtering;

 empid | salary   | dateofjoining                    | deptname  | designation       | empname
-------+----------+----------------------------------+-----------+-------------------+---------
     3 | 1.15e+05 | 2011-06-30 18:30:00.000000+0000  |     SALES | ASSISTANT MANAGER |  CHIRAG
     6 |    95000 | 2010-04-25 18:30:00.000000+0000  | TECHNICAL |           MANAGER |  FARHAN
     7 |    95000 | 2010-04-25 18:30:00.000000+0000  |        PR |           MANAGER |   JIMMY
     5 |    85000 | 2010-04-25 18:30:00.000000+0000  | TECHNICAL | ASSISTANT MANAGER |    ESHA
     4 |    75000 | 2010-04-25 18:30:00.000000+0000  | MARKETING | ASSISTANT MANAGER | DHANUSH
     1 |    50000 | 2005-04-05 18:30:00.000000+0000  | MARKETING | ASSISTANT MANAGER |  LOKESH
     2 |    30000 | 2013-11-09 18:30:00.000000+0000  | LOGISTICS | ASSISTANT MANAGER | DHEERAJ

(7 rows)
cqlsh:employee>
```

6. Alter the schema of the table Employee_Info to add a column Projects which stores a set of Projects done by the corresponding Employee.

```
(7 rows)
cqlsh:employee> ALTER TABLE EMPLOYEEINFO ADD PROJECTS LIST<TEXT>;
cqlsh:employee> SELECT * FROM EMPLOYEEINFO;

 empid | salary   | dateofjoining                    | deptname   | designation       | empname | projects
-------+----------+----------------------------------+------------+-------------------+---------+---------
     5 |    85000 | 2010-04-25 18:30:00.000000+0000  | TECHNICAL  | ASSISTANT MANAGER |    ESHA |    null
     1 |    50000 | 2005-04-05 18:30:00.000000+0000  | MARKETING  | ASSISTANT MANAGER |  LOKESH |    null
     2 |    30000 | 2013-11-09 18:30:00.000000+0000  | LOGISTICS  | ASSISTANT MANAGER | DHEERAJ |    null
     4 |    75000 | 2010-04-25 18:30:00.000000+0000  | MARKETING  | ASSISTANT MANAGER | DHANUSH |    null
   121 |    99000 | 2010-04-25 18:30:00.000000+0000  | MANAGEMENT |   REGIONAL MANAGER |   HARRY |    null
     7 |    95000 | 2010-04-25 18:30:00.000000+0000  |         PR |           MANAGER |   JIMMY |    null
     6 |    95000 | 2010-04-25 18:30:00.000000+0000  | TECHNICAL  |           MANAGER |  FARHAN |    null
     3 | 1.15e+05 | 2011-06-30 18:30:00.000000+0000  |      SALES | ASSISTANT MANAGER |  CHIRAG |    null

(8 rows)
cqlsh:employee>
```

7. Update the altered table to add project names.

8. Create a TTL of 15 seconds to display the values of Employees.

//BEFORE 15 seconds

Cassandra Lab Program 2: -

Perform the following DB operations using Cassandra.

1.Create a key space by name Library

```
C:\ Command Prompt - CQLSH

cqlsh> create keyspace library with replication = {
   ... 'class':'SimpleStrategy', 'replication_factor':1
   ... };
cqlsh> describe keyspaces

system_schema  system    samples              employee
system_auth    library   system_distributed   system_traces

cqlsh> USE library;
cqlsh:library> _
```

2. Create a column family by name Library-Info with attributes Stud_Id Primary Key,

Counter_value of type Counter,

Stud_Name, Book-Name, Book-Id, Date_of_issue

```
cqlsh> USE library;
cqlsh:library> CREATE TABLE LIBRARY_INFO( STUDID INT PRIMARY KEY, STUDNAME TEXT, BOOKNAME TEXT, DATEOFISSUE TIMESTAMP,
 COUNTER_VALUE COUNTER);
InvalidRequest: Error from server: code=2200 [Invalid query] message="Cannot mix counter and non counter columns in th
e same table"
cqlsh:library> CREATE TABLE LIBRARY_INFO( STUDID INT, STUDNAME TEXT, BOOKNAME TEXT, BOOKID INT, DATEOFISSUE TIMESTAMP,
 COUNTER_VALUE COUNTER, PRIMARY KEY(STUDID, STUDNAME, BOOKNAME, BOOKID, DATEOFISSUE));
cqlsh:library> SELECT * FROM LIBRARYINFO;
InvalidRequest: Error from server: code=2200 [Invalid query] message="unconfigured table libraryinfo"
cqlsh:library> SELECT * FROM LIBRARY_INFO;

 studid | studname | bookname | bookid | dateofissue | counter_value
--------+----------+----------+--------+-------------+---------------

(0 rows)
cqlsh:library>
```

3. Insert the values into the table in batch

```
cqlsh:library> update library_info  set counter_value = counter_value + 1 where studid = 1 and studname = 'MAHESH' and
 bookname = 'Harry Potter' and bookid = 1 and dateofissue = '2022-01-02';
cqlsh:library> SELECT * FROM LIBRARY_INFO;

 studid | studname | bookname     | bookid | dateofissue                     | counter_value
--------+----------+--------------+--------+---------------------------------+--------------
      1 |   MAHESH | Harry Potter |      1 | 2022-01-01 18:30:00.000000+0000 |             1

(1 rows)
cqlsh:library>
```

```
cqlsh:library> update library_info  set counter_value = counter_value + 1 where studid = 2 and studname = 'Ramesh' and
 bookname = 'Wings of Fire' and bookid = 2 and dateofissue = '2022-01-02';
cqlsh:library> SELECT * FROM LIBRARY_INFO;

 studid | studname | bookname      | bookid | dateofissue                     | counter_value
--------+----------+---------------+--------+---------------------------------+--------------
      1 |   MAHESH | Harry Potter  |      1 | 2022-01-01 18:30:00.000000+0000 |             1
      2 |   Ramesh | Wings of Fire |      2 | 2022-01-01 18:30:00.000000+0000 |             1

(2 rows)
cqlsh:library>
```

4. Display the details of the table created and increase the value of the counter

```
cqlsh:library> update library_info  set counter_value = counter_value + 1 where studid = 112 and studname = 'Rajesh' a
nd bookname = 'BDA' and bookid = 3 and dateofissue = '2022-01-02';
cqlsh:library> SELECT * FROM LIBRARY_INFO;

 studid | studname | bookname      | bookid | dateofissue                     | counter_value
--------+----------+---------------+--------+---------------------------------+--------------
      1 |   MAHESH | Harry Potter  |      1 | 2022-01-01 18:30:00.000000+0000 |             1
      2 |   Ramesh | Wings of Fire |      2 | 2022-01-01 18:30:00.000000+0000 |             1
    112 |   Rajesh |           BDA |      3 | 2022-01-01 18:30:00.000000+0000 |             1

(3 rows)
cqlsh:library>
```

```
(3 rows)
cqlsh:library> update library_info  set counter_value = counter_value + 1 where studid = 112 and studname = 'Rajesh' a
nd bookname = 'BDA' and bookid = 3 and dateofissue = '2022-01-02';
cqlsh:library> SELECT * FROM LIBRARY_INFO;

 studid | studname | bookname      | bookid | dateofissue                     | counter_value
--------+----------+---------------+--------+---------------------------------+--------------
      1 |   MAHESH | Harry Potter  |      1 | 2022-01-01 18:30:00.000000+0000 |             1
      2 |   Ramesh | Wings of Fire |      2 | 2022-01-01 18:30:00.000000+0000 |             1
    112 |   Rajesh |           BDA |      3 | 2022-01-01 18:30:00.000000+0000 |             2

(3 rows)
cqlsh:library>
```

```
studid | studname | bookname      | bookid | dateofissue                       | counter_value
-------+----------+---------------+--------+-----------------------------------+--------------
   113 | Ranjith  |           rpa |      4 | 2022-01-01 18:30:00.000000+0000   |             1
     1 | MAHESH   | Harry Potter  |      1 | 2022-01-01 18:30:00.000000+0000   |             1
     2 | Ramesh   | Wings of Fire |      2 | 2022-01-01 18:30:00.000000+0000   |             1
   112 | Rajesh   |           BDA |      3 | 2022-01-01 18:30:00.000000+0000   |             3

(4 rows)
```

5. Write a query to show that a student with id 112 has taken a book "BDA" 3 times.

Command Prompt - CQLSH

```
cqlsh:library> select * from library_info where studid = 112;

studid | studname | bookname | bookid | dateofissue                       | counter_value
-------+----------+----------+--------+-----------------------------------+--------------
   112 | Rajesh   |      BDA |      3 | 2022-01-01 18:30:00.000000+0000   |             3

(1 rows)
cqlsh:library>
```

6. Export the created column to a csv file

```
cqlsh:library> copy library_info (studid, studname, bookname, bookid, dateofissue, counter_value) to 'C:\Users\Admin\O
neDrive\Desktop\BDA Lab\data.csv';
Using 7 child processes

Starting copy of library.library_info with columns [studid, studname, bookname, bookid, dateofissue, counter_value].
Processed: 4 rows; Rate:      2 rows/s; Avg. rate:      1 rows/s
4 rows exported to 1 files in 3.004 seconds.
cqlsh:library> _
```

	Clipboard		⌄		Font		⌄		Alignment

POSSIBLE DATA LOSS Some features might be lost if you save this workbook in the comma-delimited

| A1 | ▼ | ⋮ | ✕ | ✓ | *fx* | 113 | |

	A	B	C	D	E	F	G	H	I
1	113	Ranjith	rpa		4	2022-01-0:	1		
2	2	Ramesh	Wings of F		2	2022-01-0:	1		
3	112	Rajesh	BDA		3	2022-01-0:	3		
4	1	MAHESH	Harry Pott		1	2022-01-0:	1		
5									
6									
7									

7. Import a given csv dataset from local file system into Cassandra column family

```
cqlsh:library> copy library_info (studid, studname, bookname, bookid, dateofissue, counter_value) from 'C:\Users\Admin\OneDrive\Desktop\BDA Lab\data.csv';
Using 7 child processes

Starting copy of library.library_info with columns [studid, studname, bookname, bookid, dateofissue, counter_value].
Process ImportProcess-10:      2 rows/s; Avg. rate:      2 rows/s
TPraceback (most recent call last):
rocess ImportProcess-8:
P Process ImportProcess-11:
TTraceback (most recent call last):
rocess ImportProcess-9:
 P File "C:\Python27\lib\multiprocessing\process.py", line 267, in _bootstrap
 File "C:\Python27\lib\multiprocessing\process.py", line 267, in _bootstrap
Traceback (most recent call last):
Praceback (most recent call last):
 P File "C:\Python27\lib\multiprocessing\process.py", line 267, in _bootstrap
   self.run()
rocess ImportProcess-12:
 rocess ImportProcess-14:
rocess ImportProcess-13:
 T File "c:\apache-cassandra-3.11.13\bin\..\pylib\cqlshlib\copyutil.py", line 2339, in run
   self.run()
TTraceback (most recent call last):
 File "C:\Python27\lib\multiprocessing\process.py", line 267, in _bootstrap
raceback (most recent call last):
   self.run()
raceback (most recent call last):
     self.run()
 File "C:\Python27\lib\multiprocessing\process.py", line 267, in _bootstrap
 File "c:\apache-cassandra-3.11.13\bin\..\pylib\cqlshlib\copyutil.py", line 2339, in run
 File "C:\Python27\lib\multiprocessing\process.py", line 267, in _bootstrap
   self.close()
   self.run()
 File "c:\apache-cassandra-3.11.13\bin\..\pylib\cqlshlib\copyutil.py", line 2339, in run
   self.run()
 File "c:\apache-cassandra-3.11.13\bin\..\pylib\cqlshlib\copyutil.py", line 2343, in close
   File "c:\apache-cassandra-3.11.13\bin\..\pylib\cqlshlib\copyutil.py", line 2339, in run
   self.close()
 File "C:\Python27\lib\multiprocessing\process.py", line 267, in _bootstrap
 File "c:\apache-cassandra-3.11.13\bin\..\pylib\cqlshlib\copyutil.py", line 2339, in run
   File "c:\apache-cassandra-3.11.13\bin\..\pylib\cqlshlib\copyutil.py", line 2339, in run
   self._session.cluster.shutdown()
   self.run()
   self.close()
   File "c:\apache-cassandra-3.11.13\bin\..\lib\cassandra-driver-internal-only-3.11.0-bb96859b.zip\cassandra-driver-3.11.0-bb96859b\cassandra\cluster.py", line 1259, in shutdown
   File "c:\apache-cassandra-3.11.13\bin\..\pylib\cqlshlib\copyutil.py", line 2343, in close
 File "c:\apache-cassandra-3.11.13\bin\..\pylib\cqlshlib\copyutil.py", line 2339, in run
   self.close()
   self.close()
   self.close()
File "c:\apache-cassandra-3.11.13\bin\..\pylib\cqlshlib\copyutil.py", line 2343, in close
```

File "c:\apache-cassandra-3.11.13\bin\..\lib\cassandra-driver-internal-only-3.11.0-bb96859b.zip\cassandra-driver-3.11.0-bb96859b\cassandra\io\asyncorereactor.py", line 373, in close
File "c:\apache-cassandra-3.11.13\bin\..\lib\cassandra-driver-internal-only-3.11.0-bb96859b.zip\cassandra-driver-3.11.0-bb96859b\cassandra\io\asyncorereactor.py", line 335, in create_timer
    File "c:\apache-cassandra-3.11.13\bin\..\lib\cassandra-driver-internal-only-3.11.0-bb96859b.zip\cassandra-driver-3.11.0-bb96859b\cassandra\io\asyncorereactor.py", line 373, in close
  self._connection.close()
File "c:\apache-cassandra-3.11.13\bin\..\lib\cassandra-driver-internal-only-3.11.0-bb96859b.zip\cassandra-driver-3.11.0-bb96859b\cassandra\io\asyncorereactor.py", line 335, in create_timer
  self._connection.close()
   AsyncoreConnection.create_timer(0, partial(asyncore.dispatcher.close, self))
   File "c:\apache-cassandra-3.11.13\bin\..\lib\cassandra-driver-internal-only-3.11.0-bb96859b.zip\cassandra-driver-3.11.0-bb96859b\cassandra\io\asyncorereactor.py", line 373, in close
  cls._loop.add_timer(timer)
File "c:\apache-cassandra-3.11.13\bin\..\lib\cassandra-driver-internal-only-3.11.0-bb96859b.zip\cassandra-driver-3.11.0-bb96859b\cassandra\io\asyncorereactor.py", line 373, in close
  cls._loop.add_timer(timer)
  AsyncoreConnection.create_timer(0, partial(asyncore.dispatcher.close, self))
AA   AsyncoreConnection.create_timer(0, partial(asyncore.dispatcher.close, self))
 ttributeError: 'NoneType' object has no attribute 'add_timer'
  File "c:\apache-cassandra-3.11.13\bin\..\lib\cassandra-driver-internal-only-3.11.0-bb96859b.zip\cassandra-driver-3.11.0-bb96859b\cassandra\io\asyncorereactor.py", line 335, in create_timer
ttributeError: 'NoneType' object has no attribute 'add_timer'
  File "c:\apache-cassandra-3.11.13\bin\..\lib\cassandra-driver-internal-only-3.11.0-bb96859b.zip\cassandra-driver-3.11.0-bb96859b\cassandra\io\asyncorereactor.py", line 335, in create_timer
  File "c:\apache-cassandra-3.11.13\bin\..\lib\cassandra-driver-internal-only-3.11.0-bb96859b.zip\cassandra-driver-3.11.0-bb96859b\cassandra\io\asyncorereactor.py", line 335, in create_timer
    AsyncoreConnection.create_timer(0, partial(asyncore.dispatcher.close, self))
      cls._loop.add_timer(timer)
  cls._loop.add_timer(timer)
A File "c:\apache-cassandra-3.11.13\bin\..\lib\cassandra-driver-internal-only-3.11.0-bb96859b.zip\cassandra-driver-3.11.0-bb96859b\cassandra\io\asyncorereactor.py", line 335, in create_timer
ttributeError: 'NoneType' object has no attribute 'add_timer'
 A   cls._loop.add_timer(timer)
  cls._loop.add_timer(timer)
ttributeError: 'NoneType' object has no attribute 'add_timer'
AAttributeError: 'NoneType' object has no attribute 'add_timer'
ttributeError: 'NoneType' object has no attribute 'add_timer'
Processed: 4 rows; Rate: 1 rows/s; Avg. rate: 2 rows/s
4 rows imported from 1 files in 2.356 seconds (0 skipped).
cqlsh:library>

MongoDB Lab Program 1 (CRUD Demonstration): -

Execute the queries and upload a document with output.

I. CREATE DATABASE IN MONGODB. use myDB; db;

(Confirm the existence of your database) show dbs; (To list all

databases)

II.CRUD (CREATE, READ, UPDATE, DELETE) OPERATIONS

1. To create a collection by the name "Student". Let us take a look at the collection list prior to the creation of the new collection "Student".

db.createCollection("Student"); => sql equivalent CREATE TABLE STUDENT(…);

2. To drop a collection by the name "Student".

db.Student.drop();

3. Create a collection by the name "Students" and store the following data in it.

db.Student.insert({_id:1,StudName:"MichelleJacintha",Grade:"VII",Hobbies:"In t ernetS

urfing"});

4. Insert the document for "AryanDavid" in to the Students collection only if it does not
   already exist in the collection. However, if it is already present in the collection, then
   update the document with new values. (Update his Hobbies from "Skating" to
   "Chess".) Use "Update else insert" (if there is an existing document, it will attempt to
   update it, if there is no existing document then it will insert it).

db.Student.update({_id:3,StudName:"AryanDavid",Grade:"VII"},{$set:{Hobbies:&qu o t;Skatin

g"}},{upsert:true});

```
local    0.000GB
> db.createCollection("Student");
{ "ok" : 1 }
> db.Student.drop();
true
> db.createCollection("Student");
{ "ok" : 1 }
> db.Student.insert({_id:1, StudName:"MichelleJacintha", Grade:"VII", Hobbies:"InternetSurfing"});
WriteResult({ "nInserted" : 1 })
> db.Student.insert({_id:1, StudName:"MichelleJacintha", Grade:"VII", Hobbies:"InternetSurfing"});
WriteResult({
        "nInserted" : 0,
        "writeError" : {
                "code" : 11000,
                "errmsg" : "E11000 duplicate key error collection: myDB.Student index: _id_ dup key: { _id: 1.0 }"
        }
})
> db.Student.updateelseinsert({_id:3, StudName:"AryanDavid", Grade:"VII"},{$set:{Hobbies:"Skating"}},{upset:true});
uncaught exception: TypeError: db.Student.updateelseinsert is not a function :
@(shell):1:1
> db.Student.update({_id:3, StudName:"AryanDavid", Grade:"VII"},{$set:{Hobbies:"Skating"}},{upsert:true});
WriteResult({ "nMatched" : 0, "nUpserted" : 1, "nModified" : 0, "_id" : 3 })
>
```

```
C:\ Command Prompt - mongo                                          —  □  ×
> show collections
Student
> db.Student.find();
{ "_id" : 1, "StudName" : "MichelleJacintha", "Grade" : "VII", "Hobbies" : "InternetSurfing" }
{ "_id" : 3, "Grade" : "VII", "StudName" : "AryanDavid", "Hobbies" : "Skating" }
>
```

5. FIND METHOD

A. To search for documents from the "Students" collection based on certain search
criteria.

db.Student.find({StudName:"Aryan David"});

({cond..},{columns.. column:1, columnname:0})

```
> db.Student.find({StudName:"AryanDavid"});
{ "_id" : 3, "Grade" : "VII", "StudName" : "AryanDavid", "Hobbies" : "Skating" }
>
```

B. To display only the StudName and Grade from all the documents of the Students

collection. The identifier_id should be suppressed and NOT displayed.

db.Student.find({},{StudName:1,Grade:1,_id:0});

```
Command Prompt - mongo
> db.Student.find({},{StudName:1,Grade:1,_id:0});
{ "StudName" : "MichelleJacintha", "Grade" : "VII" }
{ "Grade" : "VII", "StudName" : "AryanDavid" }
>
```

C. To find those documents where the Grade is set to 'VII'

db.Student.find({Grade:{$eq:'VII'}}).pretty();
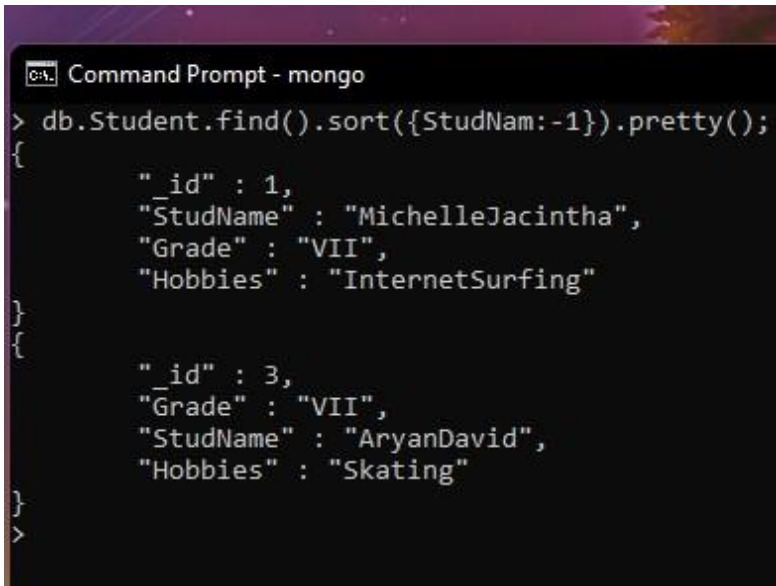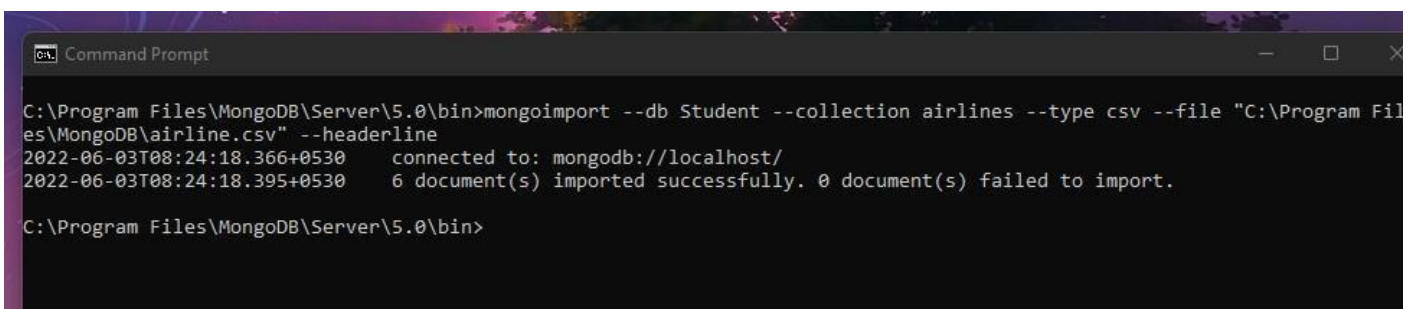
```
Command Prompt - mongo
> db.Student.find({Grade:{$eq:'VII'}}).pretty();
{
        "_id" : 1,
        "StudName" : "MichelleJacintha",
        "Grade" : "VII",
        "Hobbies" : "InternetSurfing"
}
{
        "_id" : 3,
        "Grade" : "VII",
        "StudName" : "AryanDavid",
        "Hobbies" : "Skating"
}
>
```

D. To find those documents from the Students collection where the Hobbies is set to

either 'Chess' or is set to 'Skating'.

db.Student.find({Hobbies :{ $in: ['Chess','Skating']}}).pretty ();

```
C:\ Command Prompt - mongo
> db.Student.find({Hobbies:{$in: ['Chess','Skating']}}).pretty();
{
        "_id" : 3,
        "Grade" : "VII",
        "StudName" : "AryanDavid",
        "Hobbies" : "Skating"
}
>
```

E. To find documents from the Students collection where the StudName begins with

"M". db.Student.find({StudName:/^M/}).pretty();

```
C:\ Command Prompt - mongo
> db.Student.find({StudName:/^M/}).pretty();
{
        "_id" : 1,
        "StudName" : "MichelleJacintha",
        "Grade" : "VII",
        "Hobbies" : "InternetSurfing"
}
>
```

F. To find documents from the Students collection where the StudNamehas an "e" in

any position. db.Student.find({StudName:/e/}).pretty();

```
C:\ Command Prompt - mongo
> db.Student.find({StudName:/e/}).pretty();
{
        "_id" : 1,
        "StudName" : "MichelleJacintha",
        "Grade" : "VII",
        "Hobbies" : "InternetSurfing"
}
>
```

G. To find the number of documents in the Students collection. db.Student.count();

```
C:\ Command Prompt - mongo
> db.Student.count();
2
>
```

H. To sort the documents from the Students collection in the descending order of StudName.

db.Student.find().sort({StudName:-1}).pretty();

```
Command Prompt - mongo
> db.Student.find().sort({StudNam:-1}).pretty();
{
        "_id" : 1,
        "StudName" : "MichelleJacintha",
        "Grade" : "VII",
        "Hobbies" : "InternetSurfing"
}
{
        "_id" : 3,
        "Grade" : "VII",
        "StudName" : "AryanDavid",
        "Hobbies" : "Skating"
}
>
```

III. Import data from a CSV file

Given a CSV file "sample.txt" in the D:drive, import the file into the MongoDB collection, "SampleJSON". The collection is in the database "test".

mongoimport --db Student --collection airlines --type csv –headerline --file /home/hduser/Desktop/airline.csv
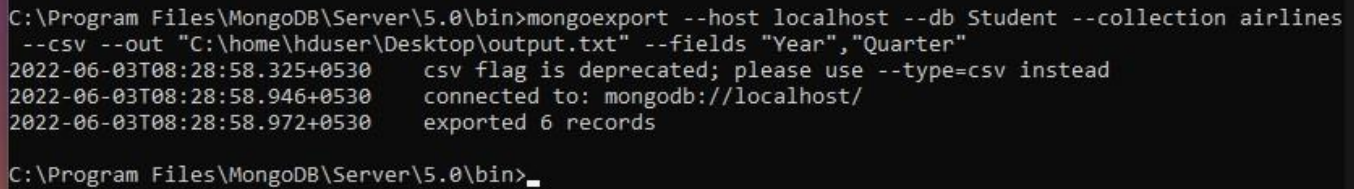
```
Command Prompt
C:\Program Files\MongoDB\Server\5.0\bin>mongoimport --db Student --collection airlines --type csv --file "C:\Program Fil
es\MongoDB\airline.csv" --headerline
2022-06-03T08:24:18.366+0530    connected to: mongodb://localhost/
2022-06-03T08:24:18.395+0530    6 document(s) imported successfully. 0 document(s) failed to import.

C:\Program Files\MongoDB\Server\5.0\bin>
```

IV. Export data to a CSV file

This command used at the command prompt exports MongoDB JSON documents from "Customers" collection in the "test" database into a CSV file "Output.txt" in the D:drive.

mongoexport --host localhost --db Student --collection airlines --csv --out
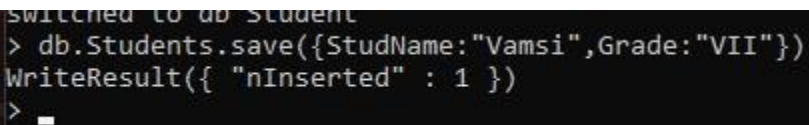/home/hduser/Desktop/output.txt –fields "Year","Quarter"

```
C:\Program Files\MongoDB\Server\5.0\bin>mongoexport --host localhost --db Student --collection airlines
 --csv --out "C:\home\hduser\Desktop\output.txt" --fields "Year","Quarter"
2022-06-03T08:28:58.325+0530    csv flag is deprecated; please use --type=csv instead
2022-06-03T08:28:58.946+0530    connected to: mongodb://localhost/
2022-06-03T08:28:58.972+0530    exported 6 records

C:\Program Files\MongoDB\Server\5.0\bin>_
```

V.   Save Method :

Save() method will insert a new document, if the document with the _id does not exist.

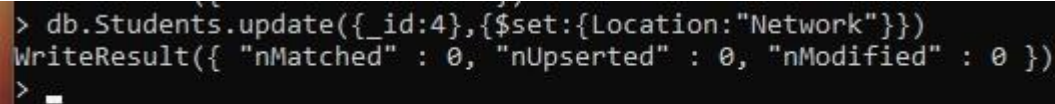If it exists it will replace the exisiting document.

db.Students.save({StudName:"Vamsi", Grade:"VI"})

```
switched to db Student
> db.Students.save({StudName:"Vamsi",Grade:"VII"})
WriteResult({ "nInserted" : 1 })
> _
```

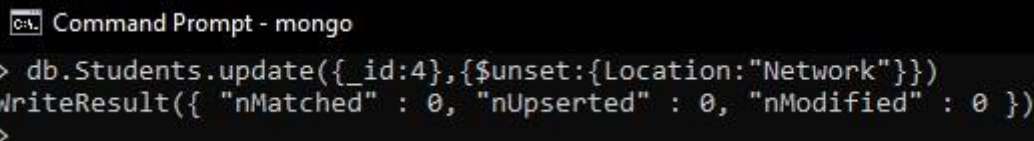VI.   Add a new field to existing Document:

db.Students.update({_id:4},{$set:{Location:"Network"}})

```
> db.Students.update({_id:4},{$set:{Location:"Network"}})
WriteResult({ "nMatched" : 0, "nUpserted" : 0, "nModified" : 0 })
> _
```

VII. Remove the field in an existing Document

db.Students.update({_id:4},{$unset:{Location:"Network"}})

```
Command Prompt - mongo
> db.Students.update({_id:4},{$unset:{Location:"Network"}})
WriteResult({ "nMatched" : 0, "nUpserted" : 0, "nModified" : 0 })
>
```

VIII.   Finding Document based on search criteria suppressing few fields

db.Student.find({_id:1},{StudName:1,Grade:1,_id:0}); To find those documents

where the Grade is not set to 'VII'

db.Student.find({Grade:{$ne:'VII'}}).pretty();

To find documents from the Students collection where the StudName ends with s.

db.Student.find({StudName:/s$/}).pretty();

```
> db.Student.find({_id:1},{StudName:1,Grade:1,_id:0});
>
```

```
 Command Prompt - mongo
> db.Student.find({Grade:{$ne:'VII'}}).pretty();
> db.Student.find({StudName:/s$/}).pretty();
> _
```

IX. to set a particular field value to NULL

```
> db.Students.update({_id:3},{$set:{Location:null}})
WriteResult({ "nMatched" : 0, "nUpserted" : 0, "nModified" : 0 })
>
```

X Count the number of documents in Student Collections

```
> db.Student.count()
0
>
```

XI. Count the number of documents in Student Collections with grade :VII

db.Students.count({Grade:"VII"}) retrieve first 3 documents

db.Students.find({Grade:"VII"}).limit(3).pretty(); Sort the document

in Ascending order db.Students.find().sort({StudName:1}).pretty();

Note: for desending order : db.Students.find().sort({StudName:-

1}).pretty(); to Skip the 1 st two documents from the Students

Collections db.Students.find().skip(2).pretty()

```
> db.Students.find().sort({StudName:1}).pretty();
{
        "_id" : ObjectId("629979944de3211e43081306"),
        "StudName" : "Vamsi",
        "Grade" : "VII"
}
>
```

XII. Create a collection by name "food" and add to each document add a "fruits" array

db.food.insert({ _id:1,

fruits:['grapes','mango','apple'] })

db.food.insert({ _id:2,

fruits:['grapes','mango','cherry'] })

db.food.insert({ _id:3, fruits:['banana','mango'] })

```
Command Prompt - mongo
> db.food.insert({_id:1,fruits:['grapes','mango','apple']})
WriteResult({ "nInserted" : 1 })
> db.food.insert({_id:2,fruits:['grapes','mango','cherry']})
WriteResult({ "nInserted" : 1 })
> db.food.insert({_id:3,fruits:['banana','mango']})
WriteResult({ "nInserted" : 1 })
>
```

To find those documents from the "food" collection which has the "fruits array"

constitute of "grapes", "mango" and "apple".

db.food.find ({fruits: ['grapes','mango','apple'] }). pretty().

```
> db.food.find({fruits:['grapes','mango','apple']}).pretty()
{ "_id" : 1, "fruits" : [ "grapes", "mango", "apple" ] }
>
```

To find in "fruits" array having "mango" in the first index position.

db.food.find ({'fruits.1':'grapes'})

```
> db.food.find({'fruits.1':'grapes'})
>
```

To find those documents from the "food" collection where the size of the array is two.

db.food.find ({"fruits": {$size:2}})

```
> db.food.find ( {"fruits": {$size:2}} )
{ "_id" : 3, "fruits" : [ "banana", "mango" ] }
>
```

To find the document with a particular id and display the first two elements from the

array "fruits"

db.food.find({_id:1},{"fruits":{$slice:2}})

```
> db.food.find({_id:1},{"fruits":{$slice:2}})
{ "_id" : 1, "fruits" : [ "grapes", "mango" ] }
>
```

To find all the documets from the food collection which have elements mango and

grapes in the array "fruits"

db.food.find({fruits:{$all:["mango","grapes"]}})

```
> db.food.find({fruits:{$all:["mango","grapes"]}})
{ "_id" : 1, "fruits" : [ "grapes", "mango", "apple" ] }
{ "_id" : 2, "fruits" : [ "grapes", "mango", "cherry" ] }
>
```

update on Array: using particular id replace the element present in the 1 st index

position of the fruits array with apple

db.food.update({_id:3},{$set:{'fruits.1':'apple'}}) insert

new key value pairs in the fruits array

db.food.update({_id:2},{$push:{price:{grapes:80,mango:200,cherry:100}}})

```
> db.food.update({_id:3},{$set:{'fruits.1':'apple'}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.food.update({_id:2},{$push:{price:{grapes:80,mango:200,cherry:100}}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
>
```

Note: perform query operations using - pop, addToSet, pullAll and pull

XII. Aggregate Function :

Create a collection Customers with fields custID, AcctBal, AcctType.

Now group on "custID" and compute the sum of "AccBal". db.Customers.aggregate

({$group : { _id : "$custID",TotAccBal : {$sum:"$AccBal"} } }); match on AcctType:"S"

then group on "CustID" and compute the sum of "AccBal". db.Customers.aggregate

({$match:{AcctType:"S"}},{$group : { _id : "$custID",TotAccBal :

{$sum:"$AccBal"} } }); match on AcctType:"S" then group on "CustID" and

compute the sum of "AccBal" and total balance greater than 1200.

db.Customers.aggregate ({$match:{AcctType:"S"}},{$group : { _id : "$custID",TotAccBal :

{$sum:"$AccBal"} } }, {$match:{TotAccBal:{$gt:1200}}});

```
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.Customers.aggregate ( {$group : { _id : "$custID",TotAccBal : {$sum:"$AccBal"} } } );
> db.Customers.aggregate ( {$match:{AcctType:"S"}},{$group : { _id : "$custID",TotAccBal :
... {$sum:"$AccBal"} } } );
uncaught exception: SyntaxError: illegal character :
@(shell):1:43
> db.Customers.aggregate ( {$match:{AcctType:"S"}},{$group : { _id :"$custID",TotAccBal :{$sum:"$AccBal
"} } } );
> db.Customers.aggregate ( {$match:{AcctType:"S"}},{$group : { _id : "$custID",TotAccBal :{$sum:"$AccBa
l"} } }, {$match:{TotAccBal:{$gt:1200}}});
>
```

MongoDB Lab Program 2 (CRUD Demonstration): -

1) Using MongoDB

i) Create a database for Students and Create a Student Collection (_id,Name, USN, Semester, Dept_Name, CGPA, Hobbies(Set)). ii) Insert required documents to the collection.

iii) First Filter on "Dept_Name:CSE" and then group it on "Semester" and

compute the Average CPGA for that semester and flter those documents where the "Avg_CPGA" is greater than 7.5.

iv) Command used to export MongoDB JSON documents from "Student" Collection into the "Students" database into a CSV fle "Output.txt".

```
> db.createCollection("Student");
{ "ok" : 1 }
```

```
> db.Student.insert({_id:1,name:"ananya",USN:"1BM19CS095",Sem:6,Dept_Name:"CSE",CGPA:"8.1",Hobbies:"Badminton"});
WriteResult({ "nInserted" : 1 })
> db.Student.insert({_id:2,name:"bharath",USN:"1BM19CS002",Sem:6,Dept_Name:"CSE",CGPA:"8.3",Hobbies:"Swimming"});
WriteResult({ "nInserted" : 1 })
> db.Student.insert({_id:3,name:"chandana",USN:"1BM19CS006",Sem:6,Dept_Name:"CSE",CGPA:"7.1",Hobbies:"Cycling"});
WriteResult({ "nInserted" : 1 })
> db.Student.insert({_id:4,name:"hrithik",USN:"1BM19CS010",Sem:6,Dept_Name:"CSE",CGPA:"8.6",Hobbies:"Reading"});
WriteResult({ "nInserted" : 1 })
> db.Student.insert({_id:5,name:"kanika",USN:"1BM19CS090",Sem:6,Dept_Name:"CSE",CGPA:"9.2",Hobbies:"Cycling"});
WriteResult({ "nInserted" : 1 })
```

```
> db.Student.update({_id:1},{$set:{CGPA:9.0}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.Student.update({_id:2},{$set:{CGPA:9.1}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.Student.update({_id:3},{$set:{CGPA:8.1}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.Student.update({_id:4},{$set:{CGPA:6.5}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.Student.update({_id:5},{$set:{CGPA:8.6}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.students.aggregate({$match:{Dept_Name:"CSE"}},{$group:{_id:"$Sem",AvgCGPA:{$avg:"$CGPA"}}},{$match:{AvgCGPA:{$gt:7.5}}});
> db.Student.aggregate({$match:{Dept_Name:"CSE"}},{$group:{_id:"$Sem",AvgCGPA:{$avg:"$CGPA"}}},{$match:{AvgCGPA:{$gt:7.5}}});
{ "_id" : 6, "AvgCGPA" : 8.26 }
```

```
bmsce@bmsce-Precision-T1700:~$ mongoexport --host localhost --db nayana_db --collection Student --csv --out /home/bmsce/Desktop/output.txt
--fields "_id","Name","USN","Sem","Dept_Name","CGPA","Hobbies"
2022-04-20T15:13:53.933+0530    csv flag is deprecated; please use --type=csv instead
2022-04-20T15:13:53.935+0530    connected to: localhost
2022-04-20T15:13:53.935+0530    exported 5 records
```

```
1 _id,Name,USN,Sem,Dept_Name,CGPA,Hobbies
2 1,,1BM19CS095,6,CSE,9,Badminton
3 2,,1BM19CS002,6,CSE,9.1,Swimming
4 3,,1BM19CS006,6,CSE,8.1,Cycling
5 4,,1BM19CS010,6,CSE,6.5,Reading
6 5,,1BM19CS090,6,CSE,8.6,Cycling
```

2) Create a mongodb collection Bank. Demonstrate the following by choosing felds of your choice.

1. Insert three documents

2. Use Arrays(Use Pull and Pop operation)

3. Use Index

4. Use Cursors

5. Updation

```
> db.createCollection("Bank");
{ "ok" : 1 }
> db.insert({CustID:1, Name:"Trivikram Hegde", Type:"Savings", Contact:["9945678231", "080-22364587"]});
uncaught exception: TypeError: db.insert is not a function :
@(shell):1:1
> db.Bank.insert({CustID:1, Name:"Trivikram Hegde", Type:"Savings", Contact:["9945678231", "080-22364587"]});
WriteResult({ "nInserted" : 1 })
> db.Bank.insert({CustID:2, Name:"Vishvesh Bhat", Type:"Savings", Contact:["6325985615", "080-23651452"]});
WriteResult({ "nInserted" : 1 })
> db.Bank.insert({CustID:3, Name:"Vaishak Bhat", Type:"Savings", Contact:["8971456321", "080-33529458"]});
WriteResult({ "nInserted" : 1 })
> db.Bank.insert({CustID:4, Name:"Pramod P Parande", Type:"Current", Contact:["9745236589", "080-56324587"]});
WriteResult({ "nInserted" : 1 })
> db.Bank.insert({CustID:4, Name:"Shreyas R S", Type:"Current", Contact:["9445678321","044-65611729", "080-25639856"]});
WriteResult({ "nInserted" : 1 })
> db.Bank.find({});
{ "_id" : ObjectId("625d77809329139694f188a2"), "CustID" : 1, "Name" : "Trivikram Hegde", "Type" : "Savings", "Contact" : [ "9945678231", "080
-22364587" ] }
{ "_id" : ObjectId("625d77bd9329139694f188a3"), "CustID" : 2, "Name" : "Vishvesh Bhat", "Type" : "Savings", "Contact" : [ "6325985615", "080-2
3651452" ] }
{ "_id" : ObjectId("625d77e69329139694f188a4"), "CustID" : 3, "Name" : "Vaishak Bhat", "Type" : "Savings", "Contact" : [ "8971456321", "080-33
529458" ] }
{ "_id" : ObjectId("625d78229329139694f188a5"), "CustID" : 4, "Name" : "Pramod P Parande", "Type" : "Current", "Contact" : [ "9745236589", "08
0-56324587" ] }
{ "_id" : ObjectId("625d78659329139694f188a6"), "CustID" : 4, "Name" : "Shreyas R S", "Type" : "Current", "Contact" : [ "9445678321", "044-656
11729", "080-25639856" ] }
> db.Bank.updateMany({CustID:1},{$pop:{Contact:1}} );
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
> db.Bank.find({});
{ "_id" : ObjectId("625d77809329139694f188a2"), "CustID" : 1, "Name" : "Trivikram Hegde", "Type" : "Savings", "Contact" : [ "9945678231" ] }
{ "_id" : ObjectId("625d77bd9329139694f188a3"), "CustID" : 2, "Name" : "Vishvesh Bhat", "Type" : "Savings", "Contact" : [ "6325985615", "080-2
3651452" ] }
{ "_id" : ObjectId("625d77e69329139694f188a4"), "CustID" : 3, "Name" : "Vaishak Bhat", "Type" : "Savings", "Contact" : [ "8971456321", "080-33
529458" ] }
{ "_id" : ObjectId("625d78229329139694f188a5"), "CustID" : 4, "Name" : "Pramod P Parande", "Type" : "Current", "Contact" : [ "9745236589", "08
0-56324587" ] }
{ "_id" : ObjectId("625d78659329139694f188a6"), "CustID" : 4, "Name" : "Shreyas R S", "Type" : "Current", "Contact" : [ "9445678321", "044-656
11729", "080-25639856" ] }
```

```
{ "_id" : ObjectId("625d78659329139694f188a6"), "CustID" : 4, "Name" : "Shreyas R S", "Type" : "Current", "Contact" : [ "9445678321", "044-656
11729", "080-25639856" ] }
> db.Bank.updateMany({},{$pull:{Contact:"080-25639856"}} );
{ "acknowledged" : true, "matchedCount" : 5, "modifiedCount" : 1 }
> db.Bank.find({});
{ "_id" : ObjectId("625d77809329139694f188a2"), "CustID" : 1, "Name" : "Trivikram Hegde", "Type" : "Savings", "Contact" : [ "9945678231" ] }
{ "_id" : ObjectId("625d77bd9329139694f188a3"), "CustID" : 2, "Name" : "Vishvesh Bhat", "Type" : "Savings", "Contact" : [ "6325985615", "080-2
3651452" ] }
{ "_id" : ObjectId("625d77e69329139694f188a4"), "CustID" : 3, "Name" : "Vaishak Bhat", "Type" : "Savings", "Contact" : [ "8971456321", "080-33
529458" ] }
{ "_id" : ObjectId("625d78229329139694f188a5"), "CustID" : 4, "Name" : "Pramod P Parande", "Type" : "Current", "Contact" : [ "9745236589", "08
0-56324587" ] }
{ "_id" : ObjectId("625d78659329139694f188a6"), "CustID" : 4, "Name" : "Shreyas R S", "Type" : "Current", "Contact" : [ "9445678321", "044-656
11729" ] }
> db.Bank.createIndex({Name:1, Type:1},{name:});
uncaught exception: SyntaxError: expected expression, got '}' :
@(shell):1:43
> db.Bank.createIndex({Name:1, Type:1},{name:"Find current account holders"});
{
        "createdCollectionAutomatically" : false,
        "numIndexesBefore" : 1,
        "numIndexesAfter" : 2,
        "ok" : 1
}
> db.Bank.find({});
{ "_id" : ObjectId("625d77809329139694f188a2"), "CustID" : 1, "Name" : "Trivikram Hegde", "Type" : "Savings", "Contact" : [ "9945678231" ] }
{ "_id" : ObjectId("625d77bd9329139694f188a3"), "CustID" : 2, "Name" : "Vishvesh Bhat", "Type" : "Savings", "Contact" : [ "6325985615", "080-2
3651452" ] }
{ "_id" : ObjectId("625d77e69329139694f188a4"), "CustID" : 3, "Name" : "Vaishak Bhat", "Type" : "Savings", "Contact" : [ "8971456321", "080-33
529458" ] }
{ "_id" : ObjectId("625d78229329139694f188a5"), "CustID" : 4, "Name" : "Pramod P Parande", "Type" : "Current", "Contact" : [ "9745236589", "08
0-56324587" ] }
{ "_id" : ObjectId("625d78659329139694f188a6"), "CustID" : 4, "Name" : "Shreyas R S", "Type" : "Current", "Contact" : [ "9445678321", "044-656
11729" ] }
> db.Bank.getIndexes()
[
        {
                "v" : 2,
```

```
@(shell):1:20
> db.Bank.update({_id:625d78659329139694f188a6}, {$set: {CustID:5}},{upsert:true});
uncaught exception: SyntaxError: identifier starts immediately after numeric literal :
@(shell):1:20
> db.Bank.update({_id:"625d78659329139694f188a6"}, {$set: {CustID:5}},{upsert:true});
WriteResult({
        "nMatched" : 0,
        "nUpserted" : 1,
        "nModified" : 0,
        "_id" : "625d78659329139694f188a6"
})
> db.Bank.find({});
{ "_id" : ObjectId("625d77809329139694f188a2"), "CustID" : 1, "Name" : "Trivikram Hegde", "Type" : "Savings", "Contact" : [ "9945678231" ] }
{ "_id" : ObjectId("625d77bd9329139694f188a3"), "CustID" : 2, "Name" : "Vishvesh Bhat", "Type" : "Savings", "Contact" : [ "6325985615", "080-2
3651452" ] }
{ "_id" : ObjectId("625d77e69329139694f188a4"), "CustID" : 3, "Name" : "Vaishak Bhat", "Type" : "Savings", "Contact" : [ "8971456321", "080-33
529458" ] }
{ "_id" : ObjectId("625d78229329139694f188a5"), "CustID" : 4, "Name" : "Pramod P Parande", "Type" : "Current", "Contact" : [ "9745236589", "08
0-56324587" ] }
{ "_id" : ObjectId("625d78659329139694f188a6"), "CustID" : 4, "Name" : "Shreyas R S", "Type" : "Current", "Contact" : [ "9445678321", "044-656
11729" ] }
{ "_id" : "625d78659329139694f188a6", "CustID" : 5 }
> db.Bank.update({_id:"625d78659329139694f188a6", CustID:5}, {$set: {Name:"Sumantha K S", Type:"Savings", Contact:["9856321478","011-65897458"
]}},{upsert:true});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.Bank.find({});
{ "_id" : ObjectId("625d77809329139694f188a2"), "CustID" : 1, "Name" : "Trivikram Hegde", "Type" : "Savings", "Contact" : [ "9945678231" ] }
{ "_id" : ObjectId("625d77bd9329139694f188a3"), "CustID" : 2, "Name" : "Vishvesh Bhat", "Type" : "Savings", "Contact" : [ "6325985615", "080-2
3651452" ] }
{ "_id" : ObjectId("625d77e69329139694f188a4"), "CustID" : 3, "Name" : "Vaishak Bhat", "Type" : "Savings", "Contact" : [ "8971456321", "080-33
529458" ] }
{ "_id" : ObjectId("625d78229329139694f188a5"), "CustID" : 4, "Name" : "Pramod P Parande", "Type" : "Current", "Contact" : [ "9745236589", "08
0-56324587" ] }
{ "_id" : ObjectId("625d78659329139694f188a6"), "CustID" : 4, "Name" : "Shreyas R S", "Type" : "Current", "Contact" : [ "9445678321", "044-656
11729" ] }
{ "_id" : "625d78659329139694f188a6", "CustID" : 5, "Contact" : [ "9856321478", "011-65897458" ], "Name" : "Sumantha K S", "Type" : "Savings"
}
>
```

1) Using MongoDB,

i) Create a database for Faculty and Create a Faculty Collection(Faculty_id, Name, Designation ,Department, Age, Salary, Specialization(Set)). ii) Insert required documents to the collection.

iii) First Filter on "Dept_Name:MECH" and then group it on "Designation"
and compute the Average Salary for that Designation and flter those

documents where the "Avg_Sal" is greater than 650000. iv) Demonstrate
usage of import and export commands

Write MongoDB queries for the following: 1) To display only the product name from
all the documents of the product collection.

2) To display only the Product ID, ExpiryDate as well as the quantity from the document of the
   product collection where the _id column is 1.

3) To fnd those documents where the price is not set to 15000.

4) To fnd those documents from the Product collection where the quantity is set to 9 and the
   product name is set to 'monitor'. 5) To fnd documents from the Product collection where the
   Product name ends in 'd'.

```
}
> db.createCollection("faculty");
{ "ok" : 1 }
> db.faculty.insert({_id:1,name:"Dr. Balaraman Ravindran",designation:"Professor",department:"CSE",age:45,salary:100000,specialization:['pytho
n','mysql','sklearn', 'tensorflow']});
WriteResult({ "nInserted" : 1 })
> db.faculty.insert({_id:2,name:"Dr. Mahadev Ghorki",designation:"Assistant Professor",department:"CSE",age:35,salary:80000,specialization:['p
ython','numpy','sklearn', 'tensorflow', 'java']});
WriteResult({ "nInserted" : 1 })
> db.faculty.insert({_id:3,name:"Dr. Praveen Borade",designation:"Associate Professor",department:"ME",age:40,salary:75000,specialization:['au
tocad', 'aerodynamics', 'thermal physics']});
WriteResult({ "nInserted" : 1 })
> db.faculty.insert({_id:4,name:"Dr. Madhav Nayak",designation:"Assistant Professor",department:"ME",age:37,salary:95000,specialization:['auto
cad', 'flight-dynamics', 'Finite Element Analysis']});
WriteResult({ "nInserted" : 1 })
> db.faculty.aggregate ( {$match:{department:"ME"}}, {$group : {_id : "$designation", AverageSal :{$avg:"$salary"} } }, {$match:{AverageSal:{
$gt:50000}}});
{ "_id" : "Associate Professor", "AverageSal" : 75000 }
{ "_id" : "Assistant Professor", "AverageSal" : 95000 }
> db.createCollection("product");
{ "ok" : 1 }
> db.product.insert({pid:1,pname:"keyboard",mdate:2001,price:1800,quantity:2});
WriteResult({ "nInserted" : 1 })
> db.product.insert({pid:2,pname:"mouse",mdate:2005,price:1500,quantity:5});
WriteResult({ "nInserted" : 1 })
> db.product.insert({pid:3,pname:"monitor",mdate:2015,price:10000,quantity:9});
WriteResult({ "nInserted" : 1 })
> db.product.insert({pid:4,pname:"motherboard",mdate:2021,price:15000,quantity:4});
WriteResult({ "nInserted" : 1 })
> db.product.find({},{pname:1,_id:0})
{ "pname" : "keyboard" }
{ "pname" : "mouse" }
{ "pname" : "monitor" }
{ "pname" : "motherboard" }
> db.product.find({pid:1},{pid:1,_id:0,mdate:1,quantity:1});
{ "pid" : 1, "mdate" : 2001, "quantity" : 2 }
> db.product.find({price:{$ne:15000}},{pname:1,_id:0});
{ "pname" : "keyboard" }
```

3) Create a mongodb collection Hospital. Demonstrate the following by choosing
felds of choice.

1
.   Insert three documents

2
.   Use Arrays(Use Pull and Pop operation)

3

.   Use Index

4
.   Use Cursors

    Updation

5
.

```
{ "pname" : "motherboard" }
> db.product.find({pid:1},{pid:1,_id:0,mdate:1,quantity:1});
{ "pid" : 1, "mdate" : 2001, "quantity" : 2 }
>  db.product.find({price:{$ne:15000}},{pname:1,_id:0});
{ "pname" : "keyboard" }
{ "pname" : "mouse" }
{ "pname" : "monitor" }
> db.product.find({$and:[{quantity:{$eq:9}},{pname:{$eq:"monitor"}}]},{pname:1,_id:0})
{ "pname" : "monitor" }
> db.product.find({pname:/d$/},{pname:1,quantity:1,_id:0})
{ "pname" : "keyboard", "quantity" : 2 }
{ "pname" : "motherboard", "quantity" : 4 }
> db.createCollection("hospital");
{ "ok" : 1 }
> db.hospital.insert({_id:1, Name: "Anshuman Agarwal", age:23, diseases:["fever", "diarrhoea", "wheezing", "gastritis"]});
WriteResult({ "nInserted" : 1 })
> db.hospital.insert({_id:2, Name: "Pinky Chaubey", age:35, diseases:["fever","nausea", "food infection", "indigestion", "kidney stones"]});
WriteResult({ "nInserted" : 1 })
> db.hospital.insert({_id:3, Name: "Amresh Chowpati", age:63, diseases:["hyperglycemia", "diabetes mellitus", "food poisoning", "cold"]});
WriteResult({ "nInserted" : 1 })
> db.hospital.updateMany({},{$pull:{diseases:"fever"}});
{ "acknowledged" : true, "matchedCount" : 3, "modifiedCount" : 2 }
> db.hospital.updateOne({_id:1},{$pop:{diseases:-1}});
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
> db.hospital.find({"diseases.2":"nausea"});
> db.hospital.find({"diseases.1":"nausea"});
> d.hospital.find({});
uncaught exception: ReferenceError: d is not defined :
@(shell):1:1
> db.hospital.find({});
{ "_id" : 1, "Name" : "Anshuman Agarwal", "age" : 23, "diseases" : [ "wheezing", "gastritis" ] }
{ "_id" : 2, "Name" : "Pinky Chaubey", "age" : 35, "diseases" : [ "nausea", "food infection", "indigestion", "kidney stones" ] }
{ "_id" : 3, "Name" : "Amresh Chowpati", "age" : 63, "diseases" : [ "hyperglycemia", "diabetes mellitus", "food poisoning", "cold" ] }
> db.hospital.find({"diseases.0":"nausea"});
{ "_id" : 2, "Name" : "Pinky Chaubey", "age" : 35, "diseases" : [ "nausea", "food infection", "indigestion", "kidney stones" ] }
> db.hospital.update({_id:3},{$set:{'diseases.1':'sarscov'}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
>
```

Hadoop Commands

hdusersbmsce-OptiPlus-3000:-$ sudo su hduser [sudo] password for hduser:

hdusersbmsce-OptiPlus-3000: $ start-all.sh

This script is Deprecated. Instead use start-dfs.sh and start-yarn.sh

22/06/06 14:43:45 WARN util.NativeCodeLoader: Unable to load native-hadoop

Library for your platform... using builtin-java classes where applicable Starting namenodes on [localhost] localhost: nanenade running as process 3396. Stop it first. localhost: datanode running as process 3564, Stop it first. starting secondary nanenodes [0.0.0.0)

0.0.0.0: secondarynamenode running as process 3773. Stop it first. O22/06/06 14:43:47 WARN uttt.NativeCodeLoader: Unable to load native-hadoop library for your starting yarn daemons resource process 3932. Stop it first.

Localhost: running as process 4255. stop it first.

6003 Jps

3932 ResourceManager

3773 SecondaryNameNode 4255 NodeManager

hdusersbmsce-OptiPlus-3060:-$ hdfs dfs -mkdir /khushil hdusersbmsce-OptiPlus-3060: $ hdfs dfs -ls /

22/06/06 14:45:30 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable Found 19 itens

drwxr-xr-x hduser supergroup 02022-06-06 11:44 /AAA drwxr-xr-x -hduser supergroup 2022-06-03 12:17 /Army drwxr-xr-x hduser supergroup 02022-06-06 11:40 /Avnit drwxr-xr-x -hduser supergroup 02022-05-31 10:44 /88 drwxr-xr-x -hduser supergroup 02022-06-01 15:03 /Cath drwxr-xr-x -hduser supergroup drwxr-xr-x hduser supergroup drwxr-xr-x -hduser supergroup drwxr-xr-x -hduser supergroup drwxr-xr-x -hduser supergroup drwxr-xr-x -hduser supergroup drwxr-xr-x -hduser supergroup drwxr-xr-x -hduser supergroup drwxr-xr-x -hduser supergroup drwxr-xr-x -hduser supergroup drwxr-xr-x -hduser supergroup

82022-06-04 10:06 /FFF

02022-06-06 14:40 /Kmrv

02022-06-06 14:44 /Khushil

02022-06-01 15:03 /Neha

02022-06-04 09:54 /WC.txt

0 2022-06-04 09:54 /welcone.txt

02022-06-06 11:36 /abc

62022-06-03 12:13 /akash

0 2022-06-03 15:12 /darshan

0 2022-06-04 09:31 /ghh 8 2022-06-06 11:45 /hello drwxr-xr-x -hduser supergroup 62022-06-04 09:35 /rahul drwxr-xr-x -hduser supergroup 02022-06-03 12:11 /shre drwxr-xr-x .hduser supergroup 02022-06-03 12:41 /shreshtha

hdusersbmsce-OptiPlus-3060:-$ hdfs dfs put /home/hduser/Desktop/6b.txt /Khushil/WC.txt

22/05/06 14:46:40 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using butltin-java classes where applicable hduserabesce-OptiPlex-3060:-$ hdfs dfs cat /Khushil/WC.txt

22/06/06 14:47:00 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable hello fron of

hdusersbmsce-OptiPlus-3040:-$ hdfs dfs-get /Khushil/WC.txt /home/hduser/Downloads/newic.txt

22/05/06 14:51:43 WARN util.NativeCodeLoader: Unable to load nattve-hadoop library for your platform... using builtin-java classes where applicable hdusersbmsce-OptiPlus-3066:-$ cd Downloads hdusersbmsce-OptiPlus-3060:-

```
/Downloads$ cat newwMC.Ext hello from 6E
  hdusersbmsce-OptiPlus-3060:-$ hdfs dfs -1s /Khushil/
  22/06/06 14:54:04 WARN util.NativeCodeLoader: Unable to load native-hadoop
  Library for your platform... using builtin java classes where applicable
  Found 2 itens
  -rw-r--r-- 1 hduser supergroup
  23 2822-06-06 14:46 /Khushil/MC.txt
  1 hduser supergroup
  23 2022-06-06 14:58 /Khushil/newwc.txt
  hdusersbmsce-OptiPlus-3060:-5 hdfs drs -getmerge /Khushil/wc.txt
  /Khushil/newwc.txt /bone/hduser/Desktop/newmerge.txt
  22/06/06 14:55:18 NARN util.NativeCodeLoader: Unable to load nattve-hadoop library for your platform... using
  butitin-Java classes where applicable hduserabesce-OptiPlex-3060:~$ cd Desktop hduser@besce-OptiPlex-3060:-
  /Desktops cat newmerge.txt hello from 68
  D B
  hello from 68
  D B
  hdusersbmsce-OptiPlus-3060:-/Desktops hadoop fs getfacl /Khushil/ 22/06/06 14:56:24 WARN util.NativeCodeLoader:
  Unable to load native hadoop library for your platform... using builtin java classes where applicable
  # file: /Khushil
  # owner: hduser # group: supergroup user::rwx group::r-x other::r-x
  hdusersbmsce-OptiPlus-3060:-/Desktop5 hdfs dfs copyToLocal /Khushil/HC.txt
  /home/hduser/Desktop
  22/05/06 14:58:09 WARN util.NativeCodeLoader: Unable to load native-hadoop Library for your platform... using
  butltin-java classes where applicable hdusersbmsce-OptiPlus-3000:-/Desktop5 cat MC.txt hello fron 68
  hdusersbmsce-OptiPlus-3060:-/Desktops hdfs dfs -cat /Khushil/MC.txt 22/06/06 14:58:59 WARN
  util.NativeCodeLoader: Unable to load native-hadoop Library for your platform... ustng bulltin-Java classes where
  applicable hello from GB B
  hdusersbmsce-OptiPlus-3060:-/Desktop5 hadoop fs - /Khushil /FFF 22/06/06 14:59:46 WARN util.NativeCodeLoader:
  Unable to load native-hadoop Library for your platform... using builtin-java classes where applicable hduseransce-
  OptiPlex-3060:-/Desktops hadoop fs-Ls /FFF 22/05/06 15:00:00 WARN util.NativeCodeLoader: Unable to load native-
  hadoop library for your platform... using butltin-java classes where applicable Found 2 itens drwxr- xr-x -hduser
  supergroup TWEE 1 hduser supergroup 02022-05-06 14:50
  /FFF/Khushil 17 2022-05-04 10:06 /FFF/MC.txt
  hdusersbmsce-OptiPlus-3060:-/Desktops hadoop fs cp /FFF/ /LLL
  22/06/06 15:09:34 WARN util.NativeCodeLoader: Unable to load native hadoop library for your platform... using
  butltin-java classes where applicable hdusersbmsce-OptiPlus-3060:-/Desktops hadoop fs -Ls /LLL
  22/06/06 15:10:07 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using
  builtin-java classes where applicable Found 2 1tens
  drwxr-xr-x -hduser supergroup hdusersbmsce-OptiPlus-3000:-/Desktops
  02022-06-06 15:09 /LLL/KHUSHIL
  17 2022-00-00 15:09 /LLL/MC.txt
```

Hadoop Programs

1) Word Count

WCMapper Java Class file.

```java
// Importing libraries
import java.io.IOException;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.MapReduceBase;
import org.apache.hadoop.mapred.Mapper;
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reporter;

public class WCMapper extends MapReduceBase implements
Mapper<LongWritable,
                    Text, Text, IntWritable> {

// Map function
    public void map(LongWritable key, Text value, OutputCollector<Text,
            IntWritable> output, Reporter rep) throws IOException

{

String line = value.toString();

// Splitting the line on spaces for (String word : line.split(" "))
{ if (word.length() > 0)
{ output.collect(new Text(word), new IntWritable(1));
        } }      } }
```

Reducer Code

```java
// Importing libraries
import java.io.IOException;
import java.util.Iterator;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.MapReduceBase;
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reducer;
import org.apache.hadoop.mapred.Reporter;

public class WCReducer extends MapReduceBase implements Reducer<Text,
IntWritable, Text, IntWritable> {
```

```java
// Reduce function
public void reduce(Text key, Iterator<IntWritable> value,
OutputCollector<Text, IntWritable> output,
                        Reporter rep) throws IOException
{ int count = 0;

// Counting the frequency of each words while (value.hasNext())
{
IntWritable i = value.next(); count += i.get();
}


   output.collect(key, new IntWritable(count)); }
}
```

Driver Code:

```java
//      Importing      libraries      import      java.io.IOException;      import
org.apache.hadoop.conf.Configured;   import   org.apache.hadoop.fs.Path;   import
org.apache.hadoop.io.IntWritable;    import      org.apache.hadoop.io.Text;    import
org.apache.hadoop.mapred.FileInputFormat;                               import
org.apache.hadoop.mapred.FileOutputFormat;                              import
org.apache.hadoop.mapred.JobClient;  import  org.apache.hadoop.mapred.JobConf;
import org.apache.hadoop.util.Tool; import org.apache.hadoop.util.ToolRunner;

        public class WCDriver extends Configured implements Tool {

public int run(String args[]) throws IOException
{ if (args.length < 2)
{
System.out.println("Please give valid inputs"); return -1;
}


JobConf conf = new JobConf(WCDriver.class);
FileInputFormat.setInputPaths(conf,                new                Path(args[0]));
FileOutputFormat.setOutputPath(conf, new Path(args[1]));
conf.setMapperClass(WCMapper.class); conf.setReducerClass(WCReducer.class);
conf.setMapOutputKeyClass(Text.class);
```

```
conf.setMapOutputValueClass(IntWritable.class);
conf.setOutputKeyClass(Text.class); conf.setOutputValueClass(IntWritable.class);
JobClient.runJob(conf); return 0;
}

// Main Method
public static void main(String args[]) throws Exception
{ int exitCode = ToolRunner.run(new WCDriver(), args);
System.out.println(exitCode);
}
}
```

Output :

```
drwxr-xr-x   - hduser supergroup          0 2022-06-01 09:46 /user1
hduser@bmsce-Precision-T1700:~$ hdfs dfs -mkdir /input_khushil
hduser@bmsce-Precision-T1700:~$ hdfs dfs -ls /
Found 19 items
drwxr-xr-x   - hduser supergroup          0 2022-06-06 12:35 /CSE
drwxr-xr-x   - hduser supergroup          0 2022-06-06 12:23 /FFF
drwxr-xr-x   - hduser supergroup          0 2022-06-06 12:36 /LLL
drwxr-xr-x   - hduser supergroup          0 2022-06-20 12:06 /amit_bda
drwxr-xr-x   - hduser supergroup          0 2022-06-03 14:52 /bharath
drwxr-xr-x   - hduser supergroup          0 2022-06-03 14:43 /bharath035
drwxr-xr-x   - hduser supergroup          0 2022-05-31 10:21 /example
drwxr-xr-x   - hduser supergroup          0 2022-06-01 15:13 /foldernew
drwxr-xr-x   - hduser supergroup          0 2022-06-06 15:04 /hemang061
drwxr-xr-x   - hduser supergroup          0 2022-06-20 15:13 /input_khushil
drwxr-xr-x   - hduser supergroup          0 2022-06-03 12:27 /irfan
drwxr-xr-x   - hduser supergroup          0 2022-06-01 15:09 /muskan
drwxr-xr-x   - hduser supergroup          0 2022-06-06 15:04 /new_folder
drwxr-xr-x   - hduser supergroup          0 2022-05-31 10:26 /one
drwxr-xr-x   - hduser supergroup          0 2022-06-20 12:17 /output
drwxr-xr-x   - hduser supergroup          0 2022-06-03 12:08 /saurab
drwxrwxr-x   - hduser supergroup          0 2019-08-01 16:19 /tmp
drwxr-xr-x   - hduser supergroup          0 2019-08-01 16:03 /user
drwxr-xr-x   - hduser supergroup          0 2022-06-01 09:46 /user1
hduser@bmsce-Precision-T1700:~$ hdfs dfs -put /home/hduser/sample.txt /input_khushil
hduser@bmsce-Precision-T1700:~$ hdfs dfs -ls /input_khushil
Found 1 items
-rw-r--r--   1 hduser supergroup         52 2022-06-20 15:15 /input_khushil/sample.txt
hduser@bmsce-Precision-T1700:~$ hadoop jar /home/hduser/khushil/WordCount.jar WCDriver
/input_khushil /input_khushil/output_khushil
22/06/20 15:16:44 INFO Configuration.deprecation: session.id is deprecated. Instead, use
dfs.metrics.session-id
22/06/20 15:16:44 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker,
sessionId=
22/06/20 15:16:44 INFO jvm.JvmMetrics: Cannot initialize JVM Metrics with
processName=JobTracker, sessionId= - already initialized
22/06/20 15:16:44 WARN mapreduce.JobSubmitter: Hadoop command-line option parsing not
performed. Implement the Tool interface and execute your application with ToolRunner to remedy
this.
22/06/20 15:16:44 INFO mapred.FileInputFormat: Total input paths to process : 1
22/06/20 15:16:44 INFO mapreduce.JobSubmitter: number of splits:1
22/06/20 15:16:44 INFO mapreduce.JobSubmitter: Submitting tokens for job:
job_local230197290_0001
22/06/20 15:16:44 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
22/06/20 15:16:44 INFO mapred.LocalJobRunner: OutputCommitter set in config null
22/06/20 15:16:44 INFO mapreduce.Job: Running job: job_local230197290_0001
22/06/20 15:16:44 INFO mapred.LocalJobRunner: OutputCommitter is
org.apache.hadoop.mapred.FileOutputCommitter
22/06/20 15:16:44 INFO mapred.LocalJobRunner: Waiting for map tasks
22/06/20 15:16:44 INFO mapred.LocalJobRunner: Starting task:
attempt_local230197290_0001_m_000000_0
22/06/20 15:16:44 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
22/06/20 15:16:44 INFO mapred.MapTask: Processing split:
hdfs://localhost:54310/input_khushil/sample.txt:0+52
22/06/20 15:16:44 INFO mapred.MapTask: numReduceTasks: 1
22/06/20 15:16:44 INFO mapred.MapTask: (EQUATOR) 0 kvi 26214396(104857584)
22/06/20 15:16:44 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100
22/06/20 15:16:44 INFO mapred.MapTask: soft limit at 83886080
22/06/20 15:16:44 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600
22/06/20 15:16:44 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
```

```
                   ac cime crupscu (mə)-r
               CPU time spent (ms)=0
               Physical memory (bytes) snapshot=0
               Virtual memory (bytes) snapshot=0
               Total committed heap usage (bytes)=471859200
       Shuffle Errors
               BAD_ID=0
               CONNECTION=0
               IO_ERROR=0
               WRONG_LENGTH=0
               WRONG_MAP=0
               WRONG_REDUCE=0
       File Input Format Counters
               Bytes Read=52
       File Output Format Counters
               Bytes Written=63
0
hduser@bmsce-Precision-T1700:~$ hdfs dfs -ls /input_khushil
Found 2 items
drwxr-xr-x   - hduser supergroup          0 2022-06-20 15:16 /input_khushil/output_khushil
-rw-r--r--   1 hduser supergroup         52 2022-06-20 15:15 /input_khushil/sample.txt
hduser@bmsce-Precision-T1700:~$ hdfs dfs -ls /input_khushil/output_khushil
Found 2 items
-rw-r--r--   1 hduser supergroup          0 2022-06-20 15:16
/input_khushil/output_khushil/_SUCCESS


-rw-r--r--   1 hduser supergroup         63 2022-06-20 15:16
/input_khushil/output_khushil/part-00000
hduser@bmsce-Precision-T1700:~$ hdfs dfs -cat /input_khushil/output_khushil/part-0000
cat: `/input_khushil/output_khushil/part-0000': No such file or directory
hduser@bmsce-Precision-T1700:~$ hdfs dfs -cat /input_khushil/output_khushil/part-00000
am      1
awesome     1
hadoop 2
hi      1
i       1
im      1
is      1
khushil     1
learing     1
```

2) Top N

Driver-TopN.class **package** samples.topn;

```java
import java.io.IOException; import java.util.StringTokenizer;
import org.apache.hadoop.conf.Configuration; import
org.apache.hadoop.fs.Path; import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text; import
org.apache.hadoop.mapreduce.Job; import
org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import
org.apache.hadoop.mapreduce.lib.output.FileOutputFormat; import
org.apache.hadoop.util.GenericOptionsParser;

public class TopN { public static void main(String[] args) throws
Exception {
Configuration conf = new Configuration();
    String[]    otherArgs   =   (new   GenericOptionsParser(conf,
args)).getRemainingArgs(); if (otherArgs.length != 2) {
System.err.println("Usage: TopN <in> <out>"); System.exit(2);
}
Job job = Job.getInstance(conf); job.setJobName("Top N");
job.setJarByClass(TopN.class); job.setMapperClass(TopNMapper.class)
job.setReducerClass(TopNReducer.class);
job.setOutputKeyClass(Text.class);
job.setOutputValueClass(IntWritable.class);
FileInputFormat.addInputPath(job, new Path(otherArgs[0]));
FileOutputFormat.setOutputPath(job, new
Path(otherArgs[1]));
  System.exit(job.waitForCompletion(true) ? 0 : 1); }


        public static class TopNMapper extends Mapper<Object, Text,
Text, IntWritable> { private static final IntWritable one = new
IntWritable(1);
 private Text word = new Text();
 private   String   tokens   =   "[_|$#<>\\^=\\[\\]\\*/\\\\,;,.\\-
:()?!\"']";
 public void map(Object key, Text value, Mapper<Object,
Text, Text, IntWritable>.Context context) throws IOException,
InterruptedException {
      String cleanLine =
value.toString().toLowerCase().replaceAll(this.tokens, " ");
StringTokenizer itr = new StringTokenizer(cleanLine); while
```

```java
(itr.hasMoreTokens()) { this.word.set(itr.nextToken().trim());
 context.write(this.word, one);
 }
 }
 }
 }
```

TopNCombiner.class **package** samples.topn;

```java
import java.io.IOException;
import org.apache.hadoop.io.IntWritable; import
org.apache.hadoop.io.Text; import org.apache.hadoop.mapreduce.Reducer

public class TopNCombiner extends Reducer<Text, IntWritable,
Text, IntWritable> { public void reduce(Text key,
Iterable<IntWritable> values, Reducer<Text, IntWritable, Text,
IntWritable>.Context context) throws IOException,
InterruptedException { int sum = 0;
for (IntWritable val : values) sum += val.get();
  context.write(key, new IntWritable(sum)); }
}
```

TopNMapper.class **package** samples.topn;

```java
import java.io.IOException; import java.util.StringTokenizer; import
org.apache.hadoop.io.IntWritable; import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

public class TopNMapper extends Mapper<Object, Text, Text,
IntWritable> { private static final IntWritable one = new
IntWritable(1);
 private Text word = new Text();
 private String tokens = "[_|$#<>\\^=\\[\\]\\*/\\\\,;,.\\- :()?!\"']"
 public vo```\\id map(Object key, Text value, Mapper<Object, Text,
Text, IntWritable>.Context context) throws IOException,
InterruptedException {
    String cleanLine =
value.toString().toLowerCase().replaceAll(this.tokens, " ");
StringTokenizer itr = new StringTokenizer(cleanLine); while
(itr.hasMoreTokens()) { this.word.set(itr.nextToken().trim());
```

```java
                context.write(this.word, one);
        }
    }
}
```

TopNReducer.class **package** samples.topn;

```java
import java.io.IOException; import java.util.HashMap; import
java.util.Map;
import org.apache.hadoop.io.IntWritable; import
org.apache.hadoop.io.Text; import org.apache.hadoop.mapreduce.Reducer
import utils.MiscUtils;
 public class TopNReducer extends Reducer<Text, IntWritable,

Text, IntWritable> { private Map<Text, IntWritable> countMap = new
HashMap<>();
 public    void    reduce(Text    key,    Iterable<IntWritable>    values,
Reducer<Text, IntWritable, Text, IntWritable>.Context context) throws
IOException, InterruptedException { int sum = 0;
for (IntWritable val : values) sum += val.get();
  this.countMap.put(new Text(key), new IntWritable(sum)); }
 protected void cleanup(Reducer<Text, IntWritable, Text,
IntWritable>.Context context) throws IOException,
InterruptedException {
Map<Text, IntWritable> sortedMap =
MiscUtils.sortByValues(this.countMap); int counter = 0;
for (Text key : sortedMap.keySet()) { if (counter++ == 20) break;
  context.write(key, sortedMap.get(key)); }
}
}
```

```
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hdfs dfs -mkdir /khushil_topn
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hdfs dfs -put ./input.txt /khushil_topn/
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hdfs dfs -ls /khushil_topn/
Found 1 items
-rw-r--r--   1 hduser supergroup        103 2022-06-27 15:43 /khushil_topn/input.txt
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hadoop jar topn.jar TopNDriver
/khushil_topn/input.txt /khushil_topn/output
Exception in thread "main" java.lang.ClassNotFoundException: TopNDriver
 at java.net.URLClassLoader.findClass(URLClassLoader.java:382)
 at java.lang.ClassLoader.loadClass(ClassLoader.java:418)
 at java.lang.ClassLoader.loadClass(ClassLoader.java:351)
 at java.lang.Class.forName0(Native Method)
 at java.lang.Class.forName(Class.java:348)
 at org.apache.hadoop.util.RunJar.run(RunJar.java:214)
 at org.apache.hadoop.util.RunJar.main(RunJar.java:136)
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hadoop jar topn.jar topn.TopNDriver
/khushil_topn/input.txt /khushil_topn/output
22/06/27 15:45:22 INFO Configuration.deprecation: session.id is deprecated. Instead, use
dfs.metrics.session-id
22/06/27 15:45:22 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker,
sessionId=
22/06/27 15:45:22 INFO input.FileInputFormat: Total input paths to process : 1
22/06/27 15:45:22 INFO mapreduce.JobSubmitter: number of splits:1
22/06/27 15:45:22 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local691635730_0001
22/06/27 15:45:22 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
22/06/27 15:45:22 INFO mapreduce.Job: Running job: job_local691635730_0001
22/06/27 15:45:22 INFO mapred.LocalJobRunner: OutputCommitter set in config null
22/06/27 15:45:22 INFO mapred.LocalJobRunner: OutputCommitter is
org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
22/06/27 15:45:22 INFO mapred.LocalJobRunner: Waiting for map tasks
22/06/27 15:45:22 INFO mapred.LocalJobRunner: Starting task: attempt_local691635730_0001_m_000000_0
22/06/27 15:45:22 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
22/06/27 15:45:22 INFO mapred.MapTask: Processing split:
hdfs://localhost:54310/khushil_topn/input.txt:0+103
22/06/27 15:45:22 INFO mapred.MapTask: (EQUATOR) 0 kvi 26214396(104857584)
22/06/27 15:45:22 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100
22/06/27 15:45:22 INFO mapred.MapTask: soft limit at 83886080
22/06/27 15:45:22 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600
22/06/27 15:45:22 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
22/06/27 15:45:22 INFO mapred.MapTask: Map output collector class =
org.apache.hadoop.mapred.MapTask$MapOutputBuffer
22/06/27 15:45:22 INFO mapred.LocalJobRunner:
22/06/27 15:45:22 INFO mapred.MapTask: Starting flush of map output
22/06/27 15:45:22 INFO mapred.MapTask: Spilling map output
22/06/27 15:45:22 INFO mapred.MapTask: bufstart = 0; bufend = 187; bufvoid = 104857600
22/06/27 15:45:22 INFO mapred.MapTask: kvstart = 26214396(104857584); kvend = 26214316(104857264);
length = 81/6553600
22/06/27 15:45:22 INFO mapred.MapTask: Finished spill 0
22/06/27 15:45:22 INFO mapred.Task: Task:attempt_local691635730_0001_m_000000_0 is done. And is in
the process of committing
22/06/27 15:45:22 INFO mapred.LocalJobRunner: map
22/06/27 15:45:22 INFO mapred.Task: Task 'attempt_local691635730_0001_m_000000_0' done.
22/06/27 15:45:22 INFO mapred.LocalJobRunner: Finishing task: attempt_local691635730_0001_m_000000_0
22/06/27 15:45:22 INFO mapred.LocalJobRunner: map task executor complete.
22/06/27 15:45:22 INFO mapred.LocalJobRunner: Waiting for reduce tasks
22/06/27 15:45:22 INFO mapred.LocalJobRunner: Starting task: attempt_local691635730_0001_r_000000_0
22/06/27 15:45:22 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
```

Output:

```
 Map input records=6
 Map output records=21
 Map output bytes=187
 Map output materialized bytes=235
 Input split bytes=110
 Combine input records=0
 Combine output records=0
 Reduce input groups=15
 Reduce shuffle bytes=235
 Reduce input records=21
 Reduce output records=15
 Spilled Records=42
 Shuffled Maps =1
 Failed Shuffles=0
 Merged Map outputs=1
 GC time elapsed (ms)=42
 CPU time spent (ms)=0
 Physical memory (bytes) snapshot=0
 Virtual memory (bytes) snapshot=0
 Total committed heap usage (bytes)=578289664
 Shuffle Errors
 BAD_ID=0
 CONNECTION=0
 IO_ERROR=0
 WRONG_LENGTH=0
 WRONG_MAP=0
 WRONG_REDUCE=0
 File Input Format Counters
 Bytes Read=103
 File Output Format Counters
 Bytes Written=105
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hdfs dfs -ls /khushil_topn/output/
Found 2 items
-rw-r--r--   1 hduser supergroup          0 2022-06-27 15:45 /khushil_topn/output/_SUCCESS
-rw-r--r--   1 hduser supergroup        105 2022-06-27 15:45 /khushil_topn/output/part-r-00000
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hdfs dfs -cat /khushil_topn/output/part-r-00000
hadoop  4
i3
am      2
hi      1
im      1
is      1
there   1
bye     1
learing 1
awesome 1
love    1
khushil 1
cool    1
and     1
using   1
hduser@bmsce-Precision-T1700:~/Desktop/temperature$
```

3) Average Temperature

AverageDriver **package** temp;

```java
import org.apache.hadoop.fs.Path; import
org.apache.hadoop.io.IntWritable; import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

public class AverageDriver { public static void main(String[] args)
throws Exception { if (args.length != 2) {
      System.err.println("Please   Enter   the   input   and   output
parameters");
System.exit(-1);
}
Job job = new Job();
job.setJarByClass(AverageDriver.class); job.setJobName("Max
temperature");
FileInputFormat.addInputPath(job, new Path(args[0]));
FileOutputFormat.setOutputPath(job, new Path(args[1]));
job.setMapperClass(AverageMapper.class);
job.setReducerClass(AverageReducer.class);
job.setOutputKeyClass(Text.class);
job.setOutputValueClass(IntWritable.class);
  System.exit(job.waitForCompletion(true) ? 0 : 1); }
}
```

AverageMapper
```java
package temp;

import java.io.IOException;
import         org.apache.hadoop.io.IntWritable;              import
org.apache.hadoop.io.LongWritable; import org.apache.hadoop.io.Text
import org.apache.hadoop.mapreduce.Mapper;

public class AverageMapper extends Mapper<LongWritable, Text,
Text, IntWritable> { public static final int MISSING = 9999;
 public void map(LongWritable key, Text value,
Mapper<LongWritable,  Text,  Text,  IntWritable>.Context  context)
throws IOException, InterruptedException { int temperature;
String line = value.toString(); String year = line.substring(15,
19); if (line.charAt(87) == '+') {
```

```
            temperature = Integer.parseInt(line.substring(88, 92));
} else { temperature = Integer.parseInt(line.substring(87, 92)); }
String quality = line.substring(92, 93);
if (temperature != 9999 && quality.matches("[01459]"))
      context.write(new Text(year), new IntWritable(temperature));
}
}
```

AverageReducer

```java
package temp;

import java.io.IOException;
import org.apache.hadoop.io.IntWritable; import
org.apache.hadoop.io.Text; import org.apache.hadoop.mapreduce.Reducer

public class AverageReducer extends Reducer<Text, IntWritable,
Text, IntWritable> { public void reduce(Text key,
Iterable<IntWritable> values, Reducer<Text, IntWritable, Text,
IntWritable>.Context context) throws IOException,
InterruptedException { int max_temp = 0; int count = 0;
for (IntWritable value : values) {
max_temp += value.get(); count++;
}
                context.write(key, new IntWritable(max_temp / count));
}
}
```

```
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ start-all.sh
This script is Deprecated. Instead use start-dfs.sh and start-yarn.sh
Starting namenodes on [localhost]
hduser@localhost's password:
localhost: starting namenode, logging to /usr/local/hadoop/logs/hadoop-hduser-namenode-bmsce-
Precision-T1700.out
hduser@localhost's password:
localhost: starting datanode, logging to /usr/local/hadoop/logs/hadoop-hduser-datanode-bmsce-
Precision-T1700.out
Starting secondary namenodes [0.0.0.0]
hduser@0.0.0.0's password:
0.0.0.0: starting secondarynamenode, logging to /usr/local/hadoop/logs/hadoop-hduser-
secondarynamenode-bmsce-Precision-T1700.out
starting yarn daemons
starting resourcemanager, logging to /usr/local/hadoop/logs/yarn-hduser-resourcemanager-bmsce-
Precision-T1700.out
hduser@localhost's password:
localhost: starting nodemanager, logging to /usr/local/hadoop/logs/yarn-hduser-nodemanager-bmsce-
Precision-T1700.out
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ jps
6832 NodeManager
6498 ResourceManager
6339 SecondaryNameNode
4887 org.eclipse.equinox.launcher_1.5.600.v20191014-2022.jar
6954 Jps
6123 DataNode
5951 NameNode
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hdfs dfs -le /
-le: Unknown command
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hdfs dfs -ls /
Found 31 items
drwxr-xr-x   - hduser supergroup          0 2022-06-06 12:35 /CSE
drwxr-xr-x   - hduser supergroup          0 2022-06-06 12:23 /FFF
drwxr-xr-x   - hduser supergroup          0 2022-06-06 12:36 /LLL
drwxr-xr-x   - hduser supergroup          0 2022-06-20 12:06 /amit_bda
drwxr-xr-x   - hduser supergroup          0 2022-06-27 11:42 /amit_lab
drwxr-xr-x   - hduser supergroup          0 2022-06-03 14:52 /bharath
drwxr-xr-x   - hduser supergroup          0 2022-06-03 14:43 /bharath035
drwxr-xr-x   - hduser supergroup          0 2022-06-24 14:54 /chi
drwxr-xr-x   - hduser supergroup          0 2022-05-31 10:21 /example
drwxr-xr-x   - hduser supergroup          0 2022-06-01 15:13 /foldernew
drwxr-xr-x   - hduser supergroup          0 2022-06-06 15:04 /hemang061
drwxr-xr-x   - hduser supergroup          0 2022-06-20 15:16 /input_khushil
drwxr-xr-x   - hduser supergroup          0 2022-06-03 12:27 /irfan
drwxr-xr-x   - hduser supergroup          0 2022-06-22 10:44 /lwde
drwxr-xr-x   - hduser supergroup          0 2022-06-27 13:03 /mapreducejoin_amit
drwxr-xr-x   - hduser supergroup          0 2022-06-22 15:32 /muskan
drwxr-xr-x   - hduser supergroup          0 2022-06-22 15:06 /muskan_op
drwxr-xr-x   - hduser supergroup          0 2022-06-22 15:35 /muskan_output
drwxr-xr-x   - hduser supergroup          0 2022-06-06 15:04 /new_folder
drwxr-xr-x   - hduser supergroup          0 2022-05-31 10:26 /one
drwxr-xr-x   - hduser supergroup          0 2022-06-24 15:30 /out55
drwxr-xr-x   - hduser supergroup          0 2022-06-20 12:17 /output
drwxr-xr-x   - hduser supergroup          0 2022-06-27 13:04 /output_TOPn
drwxr-xr-x   - hduser supergroup          0 2022-06-27 12:14 /output_Topn
drwxr-xr-x   - hduser supergroup          0 2022-06-24 12:42 /r1
drwxr-xr-x   - hduser supergroup          0 2022-06-24 12:24 /rgs
```

Output:

```
drwxr-xr-x   - hduser supergroup          0 2022-06-03 12:08 /saurab
drwxrwxr-x   - hduser supergroup          0 2019-08-01 16:19 /tmp
drwxr-xr-x   - hduser supergroup          0 2019-08-01 16:03 /user
drwxr-xr-x   - hduser supergroup          0 2022-06-01 09:46 /user1
-rw-r--r--   1 hduser supergroup       2436 2022-06-24 12:17 /wc.jar
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hdfs dfs -mkdir /khushil_temperature
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hdfs dfs -put ./1901 /khushil_temperature
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hdfs dfs -put ./1902 /khushil_temperature
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hdfs dfs -ls /khushil_temperature
Found 2 items
-rw-r--r--   1 hduser supergroup     888190 2022-06-27 14:47 /khushil_temperature/1901
-rw-r--r--   1 hduser supergroup     888978 2022-06-27 14:47 /khushil_temperature/1902
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hadoop jar ./avgtemp.jar AverageDriver
/khushil_temperature/1901 /khushil_temperature/output/
Exception in thread "main" java.lang.ClassNotFoundException: AverageDriver
 at java.net.URLClassLoader.findClass(URLClassLoader.java:382)
 at java.lang.ClassLoader.loadClass(ClassLoader.java:418)
 at java.lang.ClassLoader.loadClass(ClassLoader.java:351)
 at java.lang.Class.forName0(Native Method)
 at java.lang.Class.forName(Class.java:348)
 at org.apache.hadoop.util.RunJar.run(RunJar.java:214)
 at org.apache.hadoop.util.RunJar.main(RunJar.java:136)
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hadoop jar ./avgtemp.jar
temperature.AverageDriver /khushil_temperature/1901 /khushil_temperature/output/
22/06/27 14:53:27 INFO Configuration.deprecation: session.id is deprecated. Instead, use
dfs.metrics.session-id
22/06/27 14:53:27 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker,
sessionId=
22/06/27 14:53:27 WARN mapreduce.JobSubmitter: Hadoop command-line option parsing not performed.
Implement the Tool interface and execute your application with ToolRunner to remedy this.
22/06/27 14:53:27 INFO input.FileInputFormat: Total input paths to process : 1
22/06/27 14:53:27 INFO mapreduce.JobSubmitter: number of splits:1
22/06/27 14:53:28 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local254968295_0001
22/06/27 14:53:28 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
22/06/27 14:53:28 INFO mapreduce.Job: Running job: job_local254968295_0001
22/06/27 14:53:28 INFO mapred.LocalJobRunner: OutputCommitter set in config null
22/06/27 14:53:28 INFO mapred.LocalJobRunner: OutputCommitter is
org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
22/06/27 14:53:28 INFO mapred.LocalJobRunner: Waiting for map tasks
22/06/27 14:53:28 INFO mapred.LocalJobRunner: Starting task: attempt_local254968295_0001_m_000000_0
22/06/27 14:53:28 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
22/06/27 14:53:28 INFO mapred.MapTask: Processing split:
hdfs://localhost:54310/khushil_temperature/1901:0+888190
22/06/27 14:53:28 INFO mapred.MapTask: (EQUATOR) 0 kvi 26214396(104857584)
22/06/27 14:53:28 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100
22/06/27 14:53:28 INFO mapred.MapTask: soft limit at 83886080
22/06/27 14:53:28 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600
22/06/27 14:53:28 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
22/06/27 14:53:28 INFO mapred.MapTask: Map output collector class =
org.apache.hadoop.mapred.MapTask$MapOutputBuffer
22/06/27 14:53:28 INFO mapred.LocalJobRunner:
22/06/27 14:53:28 INFO mapred.MapTask: Starting flush of map output
22/06/27 14:53:28 INFO mapred.MapTask: Spilling map output
22/06/27 14:53:28 INFO mapred.MapTask: bufstart = 0; bufend = 59076; bufvoid = 104857600
22/06/27 14:53:28 INFO mapred.MapTask: kvstart = 26214396(104857584); kvend = 26188144(104752576);
length = 26253/6553600
22/06/27 14:53:28 INFO mapred.MapTask: Finished spill 0
```

```
FILE: Number of bytes written=723014
FILE: Number of read operations=0
FILE: Number of large read operations=0
FILE: Number of write operations=0
HDFS: Number of bytes read=1776380
HDFS: Number of bytes written=8
HDFS: Number of read operations=13
HDFS: Number of large read operations=0
HDFS: Number of write operations=4
Map-Reduce Framework
Map input records=6565
Map output records=6564
Map output bytes=59076
Map output materialized bytes=72210
Input split bytes=112
Combine input records=0
Combine output records=0
Reduce input groups=1
Reduce shuffle bytes=72210
Reduce input records=6564
Reduce output records=1
Spilled Records=13128
Shuffled Maps =1
Failed Shuffles=0
Merged Map outputs=1
GC time elapsed (ms)=55
CPU time spent (ms)=0
Physical memory (bytes) snapshot=0
Virtual memory (bytes) snapshot=0
Total committed heap usage (bytes)=999292928
Shuffle Errors
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0
File Input Format Counters
Bytes Read=888190
File Output Format Counters
Bytes Written=8
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hdfs dfs -ls /khushil_temperature/output/
Found 2 items
-rw-r--r--   1 hduser supergroup          0 2022-06-27 14:53 /khushil_temperature/output/_SUCCESS
-rw-r--r--   1 hduser supergroup          8 2022-06-27 14:53 /khushil_temperature/output/part-r-
00000
hduser@bmsce-Precision-T1700:~/Desktop/temperature$ hdfs dfs -cat /khushil_temperature/output/part-
r-00000
1901    46
hduser@bmsce-Precision-T1700:~/Desktop/temperature$
```

4) Join

```java
// JoinDriver.java  import org.apache.hadoop.conf.Configured;  import  org.apache.hadoop.fs.Path;  import
org.apache.hadoop.io.Text; import org.apache.hadoop.mapred.*;
import org.apache.hadoop.mapred.lib.MultipleInputs; import org.apache.hadoop.util.*;

public class JoinDriver extends Configured implements Tool {

public static class KeyPartitioner implements Partitioner<TextPair, Text> {
@Override
public void configure(JobConf job) {
}

@Override
public int getPartition(TextPair key, Text value, int numPartitions) {
return (key.getFirst().hashCode() & Integer.MAX_VALUE) % numPartitions;
}
}

@Override public int run(String[] args) throws Exception {

if (args.length != 3) {
System.out.println("Usage: <Department Emp Strength input>

<Department Name input> <output>"); return -1;
}

JobConf conf = new JobConf(getConf(), getClass());

conf.setJobName("Join 'Department Emp Strength input' with 'Department Name input'");

Path AInputPath = new Path(args[0]);
Path BInputPath = new Path(args[1]);
Path outputPath = new Path(args[2]);

MultipleInputs.addInputPath(conf, AInputPath, TextInputFormat.class,
Posts.class);
MultipleInputs.addInputPath(conf, BInputPath, TextInputFormat.class, User.class);
FileOutputFormat.setOutputPath(conf, outputPath); conf.setPartitionerClass(KeyPartitioner.class);
conf.setOutputValueGroupingComparator(TextPair.FirstComparator.class);
conf.setMapOutputKeyClass(TextPair.class); conf.setReducerClass(JoinReducer.class);
conf.setOutputKeyClass(Text.class);
JobClient.runJob(conf);

return 0;
}

public static void main(String[] args) throws Exception {
```

```java
int exitCode = ToolRunner.run(new JoinDriver(), args); System.exit(exitCode);
}
}


// JoinReducer.java import java.io.IOException; import java.util.Iterator; import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.*;
public class JoinReducer extends MapReduceBase implements Reducer<TextPair, Text, Text, Text> {
@Override
public void reduce (TextPair key, Iterator<Text> values, OutputCollector<Text, Text> output, Reporter reporter
throws IOException
{

Text nodeId = new Text(values.next()); while (values.hasNext()) {

Text node = values.next();
Text outValue = new Text(nodeId.toString() + "\t\t" + node.toString()); output.collect(key.getFirst(), outValue);
}
}
}

// User.java
import java.io.IOException;
import java.util.Iterator;
import org.apache.hadoop.conf.Configuration; import org.apache.hadoop.fs.FSDataInputStream; import
org.apache.hadoop.fs.FSDataOutputStream; import org.apache.hadoop.fs.FileSystem; import
org.apache.hadoop.fs.Path; import org.apache.hadoop.io.LongWritable; import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.*; import org.apache.hadoop.io.IntWritable;

public class User extends MapReduceBase implements Mapper<LongWritable, Text, TextPair, Text> {

@Override
public void map(LongWritable key, Text value, OutputCollector<TextPair, Text> output, Reporter reporter)

throws IOException

{

String valueString = value.toString();

String[] SingleNodeData = valueString.split("\t"); output.collect(new TextPair(SingleNodeData[0], "1"), new

Text(SingleNodeData[1]));
}
}

// Posts.java
import java.io.IOException;
```

```java
import org.apache.hadoop.io.*;
import org.apache.hadoop.mapred.*;

public class Posts extends MapReduceBase implements Mapper<LongWritable, Text, TextPair, Text> {

@Override
public void map(LongWritable key, Text value, OutputCollector<TextPair, Text> output, Reporter reporter)
throws IOException
{
String valueString = value.toString(); String[] SingleNodeData = valueString.split("\t");
output.collect(new TextPair(SingleNodeData[3], "0"), new

Text(SingleNodeData[9]));
}
}

// TextPair.java
import java.io.*;

import org.apache.hadoop.io.*; public class TextPair implements WritableComparable<TextPair> {

private Text first;
private Text second;

public TextPair() {
set(new Text(), new Text());
}

public TextPair(String first, String second) {
set(new Text(first), new Text(second));
}

public TextPair(Text first, Text second) {
set(first, second);
}

public void set(Text first, Text second) { this.first = first; this.second = second;
}

public Text getFirst() {
return first;
}

public Text getSecond() {
return second;
}
```

```java
@Override
    public void write(DataOutput out) throws IOException {
    first.write(out);
    second.write(out);
    }


    @Override
    public void readFields(DataInput in) throws IOException { first.readFields(in); second.readFields(in);
    }

    @Override public int hashCode() { return first.hashCode() * 163 + second.hashCode(); }

    @Override
    public boolean equals(Object o) { if (o instanceof TextPair) { TextPair tp = (TextPair) o;
    return first.equals(tp.first) && second.equals(tp.second);
    }
    return false;
    }


    @Override
    public String toString() { return first + "\t" + second;
    }

    @Override
    public int compareTo(TextPair tp) { int cmp = first.compareTo(tp.first); if (cmp != 0) { return cmp;
    }
    return second.compareTo(tp.second);
    }
// ^^ TextPair

// vv TextPairComparator public static class Comparator extends WritableComparator { private static final

    Text.Comparator TEXT_COMPARATOR = new Text.Comparator();

    public Comparator() { super(TextPair.class);
    }


    @Override
    public int compare(byte[] b1, int s1, int l1, byte[] b2, int s2, int l2) {
    try { int firstL1 = WritableUtils.decodeVIntSize(b1[s1]) + readVInt(b1, s1); int firstL2 =
    WritableUtils.decodeVIntSize(b2[s2]) + readVInt(b2, s2); int cmp = TEXT_COMPARATOR.compare(b1,
    s1, firstL1, b2, s2, firstL2); if (cmp != 0) { return cmp;
    }
                    return TEXT_COMPARATOR.compare(b1, s1 + firstL1, l1 - firstL1,

    b2, s2 + firstL2, l2 - firstL2);
    } catch (IOException e) { throw new IllegalArgumentException(e); }
```

```java
}
}

static {
    WritableComparator.define(TextPair.class, new Comparator()); }

public static class FirstComparator extends WritableComparator { private static final Text.Comparator

TEXT_COMPARATOR = new Text.Comparator();

public FirstComparator() { super(TextPair.class);
}

@Override
public int compare(byte[] b1, int s1, int l1, byte[] b2, int s2, int l2) {
 try { int firstL1 = WritableUtils.decodeVIntSize(b1[s1]) + readVInt(b1, s1); int firstL2 =
WritableUtils.decodeVIntSize(b2[s2]) + readVInt(b2, s2); return TEXT_COMPARATOR.compare(b1, s1,
firstL1, b2, s2, firstL2);
} catch (IOException e) { throw new IllegalArgumentException(e); }
}
@Override
                                        public int compare(WritableComparable a, WritableComparable b) {
if (a instanceof TextPair && b instanceof TextPair) {
return ((TextPair) a).first.compareTo(((TextPair) b).first);
}
return super.compare(a, b);
}
}
}
```

Output:

```
hduser@bmsce-Precision-T1700:~/khushil/join/MapReduceJoin$ hdfs dfs -ls /khushil_join
ls: `/khushil_join': No such file or directory
hduser@bmsce-Precision-T1700:~/khushil/join/MapReduceJoin$ hdfs dfs -mkdir /khushil_join
hduser@bmsce-Precision-T1700:~/khushil/join/MapReduceJoin$ hdfs dfs -ls /khushil_join
hduser@bmsce-Precision-T1700:~/khushil/join/MapReduceJoin$ hdfs dfs -put ./DeptName.txt
/khushil_join/
hduser@bmsce-Precision-T1700:~/khushil/join/MapReduceJoin$ hdfs dfs -put ./DeptStrength.txt
/khushil_join/
hduser@bmsce-Precision-T1700:~/khushil/join/MapReduceJoin$ hadoop jar MapReduceJoin.jar
/khushil_join/DeptName.txt /khushil_join/DeptStrength.txt /khushil_join/output/
22/06/27 15:12:24 INFO Configuration.deprecation: session.id is deprecated. Instead, use
dfs.metrics.session-id
22/06/27 15:12:24 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker,
sessionId=
22/06/27 15:12:24 INFO jvm.JvmMetrics: Cannot initialize JVM Metrics with processName=JobTracker,
sessionId= - already initialized
22/06/27 15:12:24 INFO mapred.FileInputFormat: Total input paths to process : 1
22/06/27 15:12:24 INFO mapred.FileInputFormat: Total input paths to process : 1
22/06/27 15:12:24 INFO mapreduce.JobSubmitter: number of splits:2
22/06/27 15:12:24 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local1238804660_0001
22/06/27 15:12:24 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
22/06/27 15:12:24 INFO mapred.LocalJobRunner: OutputCommitter set in config null
22/06/27 15:12:24 INFO mapreduce.Job: Running job: job_local1238804660_0001
22/06/27 15:12:24 INFO mapred.LocalJobRunner: OutputCommitter is
org.apache.hadoop.mapred.FileOutputCommitter
22/06/27 15:12:24 INFO mapred.LocalJobRunner: Waiting for map tasks
22/06/27 15:12:24 INFO mapred.LocalJobRunner: Starting task: attempt_local1238804660_0001_m_000000_0
22/06/27 15:12:24 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
22/06/27 15:12:24 INFO mapred.MapTask: Processing split:
hdfs://localhost:54310/khushil_join/DeptName.txt:0+59
22/06/27 15:12:24 INFO mapred.MapTask: numReduceTasks: 1
22/06/27 15:12:24 INFO mapred.MapTask: (EQUATOR) 0 kvi 26214396(104857584)
22/06/27 15:12:24 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100
22/06/27 15:12:24 INFO mapred.MapTask: soft limit at 83886080
22/06/27 15:12:24 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600
22/06/27 15:12:24 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
22/06/27 15:12:24 INFO mapred.MapTask: Map output collector class =
org.apache.hadoop.mapred.MapTask$MapOutputBuffer
22/06/27 15:12:24 INFO mapred.LocalJobRunner:
22/06/27 15:12:24 INFO mapred.MapTask: Starting flush of map output
22/06/27 15:12:24 INFO mapred.MapTask: Spilling map output
22/06/27 15:12:24 INFO mapred.MapTask: bufstart = 0; bufend = 63; bufvoid = 104857600
22/06/27 15:12:24 INFO mapred.MapTask: kvstart = 26214396(104857584); kvend = 26214384(104857536);
length = 13/6553600
22/06/27 15:12:24 INFO mapred.MapTask: Finished spill 0
22/06/27 15:12:24 INFO mapred.Task: Task:attempt_local1238804660_0001_m_000000_0 is done. And is in
the process of committing
22/06/27 15:12:24 INFO mapred.LocalJobRunner: hdfs://localhost:54310/khushil_join/DeptName.txt:0+59
22/06/27 15:12:24 INFO mapred.Task: Task 'attempt_local1238804660_0001_m_000000_0' done.
22/06/27 15:12:24 INFO mapred.LocalJobRunner: Finishing task:
attempt_local1238804660_0001_m_000000_0
22/06/27 15:12:24 INFO mapred.LocalJobRunner: Starting task: attempt_local1238804660_0001_m_000001_0
22/06/27 15:12:24 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
22/06/27 15:12:24 INFO mapred.MapTask: Processing split:
hdfs://localhost:54310/khushil_join/DeptStrength.txt:0+50
22/06/27 15:12:24 INFO mapred.MapTask: numReduceTasks: 1
22/06/27 15:12:24 INFO mapred.MapTask: (EQUATOR) 0 kvi 26214396(104857584)
22/06/27 15:12:24 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100
```

```
FILE: Number of bytes read=26370
FILE: Number of bytes written=782871
FILE: Number of read operations=0
FILE: Number of large read operations=0
FILE: Number of write operations=0
HDFS: Number of bytes read=277
HDFS: Number of bytes written=85
HDFS: Number of read operations=28
HDFS: Number of large read operations=0
HDFS: Number of write operations=5
Map-Reduce Framework
Map input records=8
Map output records=8
Map output bytes=117
Map output materialized bytes=145
Input split bytes=443
Combine input records=0
Combine output records=0
Reduce input groups=4
Reduce shuffle bytes=145
Reduce input records=8
Reduce output records=4
Spilled Records=16
Shuffled Maps =2
Failed Shuffles=0
Merged Map outputs=2
GC time elapsed (ms)=2
CPU time spent (ms)=0
Physical memory (bytes) snapshot=0
Virtual memory (bytes) snapshot=0
Total committed heap usage (bytes)=913833984
Shuffle Errors
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0
File Input Format Counters
Bytes Read=0
File Output Format Counters
Bytes Written=85
hduser@bmsce-Precision-T1700:~/khushil/join/MapReduceJoin$ hdfs dfs -cat /khushil_join/output2/part-
00000
A11      50            Finance
B12      100           HR
C13      250           Manufacturing
Dept_ID Total_Employee          Dept_Name
hduser@bmsce-Precision-T1700:~/khushil/join/MapReduceJoin$
```

Scala Programming:
Lab9:

```scala
val data=sc.textFile("sparkdata.txt")
data.collect;
val splitdata = data.flatMap(line => line.split(" "));
splitdata.collect;
val mapdata = splitdata.map(word => (word,1));
mapdata.collect;
val reducedata = mapdata.reduceByKey(_+_);
reducedata.collect;
```

```
scala> val data = sc.textFile("input.txt")
data: org.apache.spark.rdd.RDD[String] = input.txt MapPartitionsRDD[3] at textFile at <console>:23

scala> data.collect()
res3: Array[String] = Array(hi there im khushil, im here to run spark and hadoop, lets see which is better)

scala> val splitdata = data.flatMap(line => line.split(" "));
splitdata: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[4] at flatMap at <console>:23

scala> splitdata.collect();
res4: Array[String] = Array(hi, there, im, khushil, im, here, to, run, spark, and, hadoop, lets, see, which, is, better)

scala> val mapdata = splitdata.map(word=>(word,1));
mapdata: org.apache.spark.rdd.RDD[(String, Int)] = MapPartitionsRDD[5] at map at <console>:23

scala> val reducedata = mapdata.reduceByKey(_+_);
reducedata: org.apache.spark.rdd.RDD[(String, Int)] = ShuffledRDD[6] at reduceByKey at <console>:23

scala> reducedata.collect();
res5: Array[(String, Int)] = Array((im,2), (is,1), (here,1), (there,1), (better,1), (khushil,1), (lets,1), (spark,1), (run,1), (hadoop,1), (hi,1), (to,1), (see,1), (which,1), (and,1))

scala> reducedata.saveAsTextFile("output.txt");

scala>
```

Lab 10:

```scala
val textFile = sc.textFile("/home/bhoom/Desktop/wc.txt")
val counts = textFile.flatMap(line => line.split(" ")).map(word => (word,
1)).reduceByKey(_ + _)
import scala.collection.immutable.ListMap
val sorted=ListMap(counts.collect.sortWith(_._2 > _._2):_*)// sort in
descending order based on values
println(sorted)
for((k,v)<-sorted)
{ if(v>4)
{ print(k+",") print(v) println()
}}
```

```
scala> val filerdd = sc.textFile("input.txt");
filerdd: org.apache.spark.rdd.RDD[String] = input.txt MapPartitionsRDD[13] at textFile at <console>:24

scala> val counts = filerdd.flatMap(line=>line.split(" ")).map(word=>(word,1)).reduceByKey(_+_);
counts: org.apache.spark.rdd.RDD[(String, Int)] = ShuffledRDD[16] at reduceByKey at <console>:24

scala> import scala.collection.immutable.ListMap
import scala.collection.immutable.ListMap

scala> val sorted = ListMap(counts.collect.sortWith(_._2 > _._2): _*);
sorted: scala.collection.immutable.ListMap[String,Int] = ListMap(im -> 2, is -> 1, here -> 1, there -> 1
, better -> 1, khushil -> 1, lets -> 1, spark -> 1, run -> 1, hadoop -> 1, hi -> 1, to -> 1, see -> 1, w
hich -> 1, and -> 1)

scala> println(sorted);
ListMap(im -> 2, is -> 1, here -> 1, there -> 1, better -> 1, khushil -> 1, lets -> 1, spark -> 1, run -
> 1, hadoop -> 1, hi -> 1, to -> 1, see -> 1, which -> 1, and -> 1)

scala> for((k,v)<-sorted)
     | {
     | if(v>4)
     | {
     | print(k+",")
     | print(v)
     | println()
     | }
     | }

scala> for((k,v)<-sorted)
     | {
     | println(k+",")
     | println(v)
     | println()
     | }
im,
2

is,
1

here,
1

there,
1

better,
1

khushil,
1

lets,
1

spark,
1
```