VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“JnanaSangama”, Belgaum -590014, Karnataka.

LAB REPORT
on

BIG DATA ANALYTICS (20CS6PEBDA)

Submitted by

ANITEJ PRASAD (1BM19CS194)

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

<u>CERTIFICATE</u>

This is to certify that the Lab work entitled **“BIG DATA ANALYTICS”** carried out by **ANITEJ PRASAD (1BM19CS194),** 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 a **BIG DATA ANALYTICS - (20CS6PEBDA)** work prescribed for the said degree.

Prof. Shyamala G
Assistant Professor
Department of CSE
BMSCE, Bengaluru

Dr. Jyothi S Nayak
Professor and Head of
Department of CSE
BMSCE, Bengaluru

`

Index Sheet

Sl. No.	Experiment Title	Page No.
1	MongoDB-Students	4
2	Cassandra-Employees	14
3	Using Cassandra create a Library database	21
4	Wordcount Program-hadoop	24
5	Wordcount Program--Mapreducer	29
6	Hadoop TopN Program	43
7	Average temperature	50
8	MeanMax Temperature	60
9	Join Operation using Mapreduce	70
10	Wordcount using Scala	90
11	Wordcount greater than 4 using Scala	91

Course Outcome

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

LAB1:

Using MongoDB create a database for students.

```
bmsce@bmsce-Precision-T1700: ~$ mongo
MongoDB shell version v3.6.8
connecting to: mongodb://127.0.0.1:27017
Implicit session: session {“id”: UUID("d66acdb3-8482-417d-8b75-d65dae4b53ee")}
MongoDB server version: 3.6.8
Server has startup warnings:
2022-04-11T18:49:15.627+0530 I STORAGE [initandlisten]
2022-04-11T18:49:15.627+0530 I STORAGE [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine
2022-04-11T18:49:15.627+0530 I STORAGE [initandlisten] ** See http://dochub.mongodb.org/core/prodnotes-filesystem
2022-04-11T18:49:18.771+0530 I CONTROL [initandlisten]
2022-04-11T18:49:18.771+0530 I CONTROL [initandlisten] ** WARNING: Access control is not enabled for the database.
2022-04-11T18:49:18.771+0530 I CONTROL [initandlisten] ** Read and write access to data and configuration is unrestricted.
2022-04-11T18:49:18.771+0530 I CONTROL [initandlisten]
```

```
> use Student

switched to db Student

> db.createCollection("student");

{ "ok" : 1 }

>
db.Student.insert({_id:1,StudName:"Megha",Grade:"vii",Hobbies:"Inter
netSurfing"});

WriteResult({ "nInserted" : 1 })

>
db.Student.update({_id:3,StudName:"Ayan",Grade:"vii"},{$set:{Hobbie
s:"skating"}},{upsert:true});

WriteResult({ "nMatched" : 0, "nUpserted" : 1, "nModified" : 0, "_id" :
3 })

> db.Student.find({StudName:"Ayan"});

{ "_id" : 3, "Grade" : "vii", "StudName" : "Ayan", "Hobbies" : "skating"
}

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

{ "StudName" : "Megha", "Grade" : "vii" }

{ "Grade" : "vii", "StudName" : "Ayan" }


> db.Student.find({Grade:{$eq:'vii'}}).pretty();

{

        "_id" : 1,

        "StudName" : "Megha",

        "Grade" : "vii",
        "Hobbies" : "InternetSurfing"
}
{ "_id" : 3, "Grade" : "vii", "StudName" : "Ayan", "Hobbies" : "skating"
}
> db.Student.find({Grade:{$eq:'vii'}});
{ "_id" : 1, "StudName" : "Megha", "Grade" : "vii", "Hobbies" :
"InternetSurfing" }
{ "_id" : 3, "Grade" : "vii", "StudName" : "Ayan", "Hobbies" : "skating"
}
> db.Student.find({Grade:{$eq:'vii'}}).pretty();
{
        "_id" : 1,
        "StudName" : "Megha",
        "Grade" : "vii",
        "Hobbies" : "InternetSurfing"
}
{ "_id" : 3, "Grade" : "vii", "StudName" : "Ayan", "Hobbies" : "skating"
}
> db.Student.find({Hobbies:{$in:['Chess','Skating']}}).pretty();
> db.Student.find({Hobbies:{$in:['Skating']}}).pretty();
> db.Student.find({Hobbies:{$in:['skating']}}).pretty();
{ "_id" : 3, "Grade" : "vii", "StudName" : "Ayan", "Hobbies" : "skating"
}
> db.Student.find({StudName:/^M/}).pretty();
{
        "_id" : 1,
        "StudName" : "Megha",
        "Grade" : "vii",
        "Hobbies" : "InternetSurfing"
}
> db.Student.find({StudName:/e/}).pretty();
{
        "_id" : 1,
        "StudName" : "Megha",
        "Grade" : "vii",
        "Hobbies" : "InternetSurfing"
}
> db.Student.count();
2
> db.Student.find().sort({StudName:-1}).pretty();
{
        "_id" : 1,
        "StudName" : "Megha",
        "Grade" : "vii",
        "Hobbies" : "InternetSurfing"
}
{ "_id" : 3, "Grade" : "vii", "StudName" : "Ayan", "Hobbies" : "skating"
}
> db.Student.save({StudName:"Vamsi",Greade:"vi"})
```

```
WriteResult({ "nInserted" : 1 })
> db.Students.update({_id:4},{$set:{Location:"Network"}})
WriteResult({ "nMatched" : 0, "nUpserted" : 0, "nModified" : 0 })
> db.Students.update({_id:4},{$unset:{Location:"Network"}})
WriteResult({ "nMatched" : 0, "nUpserted" : 0, "nModified" : 0 })
> db.Student.find({_id:1},{StudName:1,Grade:1,_id:0});
{ "StudName" : "Megha", "Grade" : "vii" }
> db.Student.find({Grade:{$ne:'VII'}}).pretty();
{
        "_id" : 1,
        "StudName" : "Megha",
        "Grade" : "vii",
        "Hobbies" : "InternetSurfing"
}
{ "_id" : 3, "Grade" : "vii", "StudName" : "Ayan", "Hobbies" : "skating" }
{
        "_id" : ObjectId("6253f413e88b8c9e787b194e"),
        "StudName" : "Vamsi",
        "Greade" : "vi"
}
> db.Student.find({StudName:/s$/}).pretty();
> db.Students.update({_id:3},{$set:{Location:null}})
WriteResult({ "nMatched" : 0, "nUpserted" : 0, "nModified" : 0 })
```

```
> db.Students.count()
0
> db.Students.count({Grade:"VII"})
0
> db.Student.find({Grade:"VII"}).limit(3).pretty();
> db.Student.update({_id:3},{$set:{Location:null}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.Student.count({Grade:"VII"})
0
> db.Students.count({Grade:"vii"})
0
> db.Student.count()
3
> db.Student.count({Grade:"vii"})
2
> db.Student.find({Grade:"vii"}).limit(3).pretty();
{
        "_id" : 1,
        "StudName" : "Megha",
        "Grade" : "vii",
        "Hobbies" : "InternetSurfing"
}
{
        "_id" : 3,
        "Grade" : "vii",
        "StudName" : "Ayan",
        "Hobbies" : "skating",
        "Location" : null
}
> db.Student.find().sort({StudName:1}).pretty();
{
        "_id" : 3,
        "Grade" : "vii",
        "StudName" : "Ayan",
        "Hobbies" : "skating",
        "Location" : null
}
{
        "_id" : 1,
        "StudName" : "Megha",
        "Grade" : "vii",
        "Hobbies" : "InternetSurfing"
}
{
        "_id" : ObjectId("6253f413e88b8c9e787b194e"),
        "StudName" : "Vamsi",
        "Greade" : "vi"
}
> db.Student.find().skip(2).pretty()
{
        "_id" : ObjectId("6253f413e88b8c9e787b194e"),
        "StudName" : "Vamsi",
        "Greade" : "vi"
}
> db.food.insert( { _id:1, fruits:['grapes','mango','apple';] } )
2022-04-11T15:05:51.894+0530 E QUERY    [thread1] SyntaxError:
missing ] after element list @(shell):1:57
> db.food.insert({_id:1,fruits:['grapes','mango','apple']})
WriteResult({ "nInserted" : 1 })
> db.food.insert({_id:2,fruits:['grapes','mango','cherry']})
WriteResult({ "nInserted" : 1 })
> db.food.insert({_id:3,fruits:['banana','mango']})
WriteResult({ "nInserted" : 1 })
> db.food.find({fruits:['grapes','mango','apple']}).pretty();
{ "_id" : 1, "fruits" : [ "grapes", "mango", "apple" ] }
> db.food.find({'fruits.1':'grapes'})
> db.food.find({"fruits":{$size:2}})
{ "_id" : 3, "fruits" : [ "banana", "mango" ] }
> db.food.find({_id:1},{"fruits":{$slice:2}})
{ "_id" : 1, "fruits" : [ "grapes", "mango" ] }
```

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

{ "_id" : 1, "fruits" : [ "grapes", "mango", "apple" ] }

{ "_id" : 2, "fruits" : [ "grapes", "mango", "cherry" ] }

> 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 })

>

> db.createCollection("Customers");

{

        "ok" : 0,

        "errmsg" : "a collection 'bhuvana.Customers' already exists",

        "code" : 48,

        "codeName" : "NamespaceExists"

}

db.Customers.insert({_custID:1,AcctBal:'100000',AcctType:"saving"});

WriteResult({ "nInserted" : 1 })

>
db.Customers.aggregate({$group:{_id:"$custID",TotAccBal:{$sum:"$A
ccBal"}}});

{ "_id" : null, "TotAccBal" : 0 }

db.Customers.aggregate({$match:{AcctType:"saving"}},{$group:{_id:"
$custID",TotAccBal:{$sum:"$AccBal"}}});
```

```
{ "_id" : null, "TotAccBal" : 0 }
```

```
db.Customers.aggregate({$match:{AcctType:"saving"}},{$group:{_id:"$custID",TotAccBal:{$sum:"$AccBal"}}},{$match:{TotAccBal:{$gt:1200}}});
```

LAB 2:

Using Cassandra create a database for Employees

```
cqlsh:employee> CREATE KEYSPACE employee WITH
REPLICATION={ 'class' : 'SimpleStrategy', 'replication_factor' : 1};

cqlsh:employee> USE employee;

cqlsh:employee> create table employee_info(emp_id int PRIMARY
KEY, emp_name text,

        ... designation text, date_of_joining timestamp, salary double
PRIMARY KEY, dept_name text);

cqlsh:employee> CREATE TABLE employee_info(emp_id int,
emp_name text, designation text, date_of_joining timestamp, salary
double, dept_name text, PRIMARY KEY(emp_id, salary));

cqlsh:employee> BEGIN BATCH INSERT INTO

        ...
employee_info(emp_id,emp_name,designation,date_of_joining,salary,de
pt_name)

        ... VALUES(100,'Jogesh','MANAGER','2021-09-
11',30000,'TESTING');

        ... INSERT INTO

        ...
employee_info(emp_id,emp_name,designation,date_of_joining,salary,de
pt_name)

        ... VALUES(111,'Tamara','ASSOCIATE','2021-06-
22',25000,'DEVELOPING');
```

```
      ... INSERT INTO
      ... employee_info(emp_id,emp_name,designation,date_of_joining,salary,dept_name)
      ... VALUES(121,'Elenor','MANAGER','2021-03-30',35000,'HR');
      ... INSERT INTO
      ... employee_info(emp_id,emp_name,designation,date_of_joining,salary,dept_name)
      ... VALUES(115,'Charu','ASSISTANT','2021-12-30',20000,'DEVELOPING');
      ... INSERT INTO
      ... employee_info(emp_id,emp_name,designation,date_of_joining,salary,dept_name)
      ... VALUES(105,'Santosh','ASSOCIATE','2021-06-25',25000,'TESTING');
      ... APPLY BATCH;
cqlsh:employee> SELECT * FROM employee_info
      ... ;

 emp_id | salary | date_of_joining                 | dept_name  | designation | emp_name
--------+--------+---------------------------------+------------+-------------+----------
```

```
   105 |  25000 | 2021-06-24 18:30:00.000000+0000 |    TESTING |
ASSOCIATE |    Santosh

   111 |  25000 | 2021-06-21 18:30:00.000000+0000 | DEVELOPING |
ASSOCIATE |    Tamara

   121 |  35000 | 2021-03-29 18:30:00.000000+0000 |         HR |
MANAGER |    Elenor

   115 |  20000 | 2021-12-29 18:30:00.000000+0000 | DEVELOPING |
ASSISTANT |    Charu

   100 |  30000 | 2021-09-10 18:30:00.000000+0000 |    TESTING |
MANAGER |    Jogesh


(5 rows)

cqlsh:employee> UPDATE employee_info SET emp_name = 'Jayesh',
dept_name = 'DEVELOPING' WHERE emp_id = 121;

cqlsh:employee> UPDATE employee_info SET emp_name = 'Jayesh',
dept_name = 'DEVELOPING' WHERE emp_id = 121 AND salary =
35000;

cqlsh:employee> SELECT * FROM employee_info ;


 emp_id | salary | date_of_joining                 | dept_name  | designation |
emp_name

--------+--------+---------------------------------+------------+-------------+-----
-----

   105 |  25000 | 2021-06-24 18:30:00.000000+0000 |    TESTING |
ASSOCIATE |    Santosh

   111 |  25000 | 2021-06-21 18:30:00.000000+0000 | DEVELOPING |
ASSOCIATE |    Tamara
```

```
    121 |  35000 | 2021-03-29 18:30:00.000000+0000 | DEVELOPING |
MANAGER |    Jayesh

    115 |  20000 | 2021-12-29 18:30:00.000000+0000 | DEVELOPING |
ASSISTANT |    Charu

    100 |  30000 | 2021-09-10 18:30:00.000000+0000 |    TESTING |
MANAGER |    Jogesh


(5 rows)

cqlsh:employee> SELECT * FROM employee_info WHERE emp_id in
(105, 111, 121, 115, 100) order by salary;

cqlsh:employee> paging off

Disabled Query paging.

cqlsh:employee> SELECT * FROM employee_info WHERE emp_id in
(105, 111, 121, 115, 100) order by salary;


 emp_id | salary | date_of_joining                 | dept_name  | designation |
emp_name

--------+--------+---------------------------------+------------+-------------+-----
-----

    115 |  20000 | 2021-12-29 18:30:00.000000+0000 | DEVELOPING |
ASSISTANT |    Charu

    105 |  25000 | 2021-06-24 18:30:00.000000+0000 |    TESTING |
ASSOCIATE |    Santosh

    111 |  25000 | 2021-06-21 18:30:00.000000+0000 | DEVELOPING |
ASSOCIATE |    Tamara
```

```
   100 |  30000 | 2021-09-10 18:30:00.000000+0000 |    TESTING |
MANAGER |    Jogesh

   121 |  35000 | 2021-03-29 18:30:00.000000+0000 | DEVELOPING |
MANAGER |    Jayesh


(5 rows)

cqlsh:employee> ALTER TABLE employee_info ADD projects text;

cqlsh:employee> UPDATE employee_info SET projects = 'Chat App'
WHERE emp_id = 111;

cqlsh:employee> UPDATE employee_info SET projects = 'Chat App'
WHERE emp_id = 111 and salary = 25000;

cqlsh:employee> UPDATE employee_info SET projects = 'Discord Bot'
WHERE emp_id = 115 and salary = 20000;

cqlsh:employee> UPDATE employee_info SET projects = 'Campus
Portal' WHERE emp_id = 105 and salary = 25000;

cqlsh:employee> UPDATE employee_info SET projects = 'YouTube
Downloader' WHERE emp_id = 100 and salary = 30000;

cqlsh:employee> UPDATE employee_info SET projects = 'Library
Management System ' WHERE emp_id = 121 and salary = 35000;

cqlsh:employee> SELECT * FROM employee_infor

            ... ;

cqlsh:employee> SELECT * FROM employee_info ;


 emp_id | salary | date_of_joining                 | dept_name  | designation |
emp_name | projects
```

```
--------+--------+---------------------------------+------------+-------------+-----
-----+----------------------------

    105 |  25000 | 2021-06-24 18:30:00.000000+0000 |    TESTING |
ASSOCIATE |  Santosh |            Campus Portal

    111 |  25000 | 2021-06-21 18:30:00.000000+0000 | DEVELOPING |
ASSOCIATE |  Tamara  |                 Chat App

    121 |  35000 | 2021-03-29 18:30:00.000000+0000 | DEVELOPING |
MANAGER |  Jayesh  |  Library Management System

    115 |  20000 | 2021-12-29 18:30:00.000000+0000 | DEVELOPING |
ASSISTANT |  Charu   |              Discord Bot

    100 |  30000 | 2021-09-10 18:30:00.000000+0000 |    TESTING |
MANAGER |  Jogesh  |        YouTube Downloader


(5 rows)

cqlsh:employee> INSERT INTO

            ...
employee_info(emp_id,emp_name,designation,date_of_joining,salary,de
pt_name)

            ...

            ... ;

cqlsh:employee> INSERT INTO

            ...
employee_info(emp_id,emp_name,designation,date_of_joining,salary,de
pt_name)

            ... VALUES(110,'SAM','ASSOCIATE','2021-01-
11',28000,'TESTING') USING TTL 15;
```

```
cqlsh:employee> SELECT TTL(emp_name) from employee_info
WHERE emp_id = 110;


 ttl(emp_name)
---------------
            3


(1 rows)
cqlsh:employee> SELECT * FROM employee_info;


 emp_id | salary | date_of_joining                 | dept_name  | designation |
emp_name | projects
--------+--------+---------------------------------+------------+-------------+-----
-----+---------------------------
    105 |  25000 | 2021-06-24 18:30:00.000000+0000 |    TESTING |
ASSOCIATE |  Santosh |             Campus Portal
    111 |  25000 | 2021-06-21 18:30:00.000000+0000 | DEVELOPING |
ASSOCIATE |  Tamara  |                  Chat App
    121 |  35000 | 2021-03-29 18:30:00.000000+0000 | DEVELOPING |
MANAGER |  Jayesh  | Library Management System
    115 |  20000 | 2021-12-29 18:30:00.000000+0000 | DEVELOPING |
ASSISTANT |  Charu   |               Discord Bot
    100 |  30000 | 2021-09-10 18:30:00.000000+0000 |    TESTING |
MANAGER |  Jogesh  |        YouTube Downloader


(5 rows)
```

LAB3:

Using Cassandra create a Library database

```
cqlsh:library> CREATE KEYSPACE library WITH replication =
{'class':

'SimpleStrategy','replication_factor':1}; cqlsh:library> USE library ;

cqlsh:library> CREATE TABLE Library_info(stud_id int, stud_name
text,  book_name text, book_id text, date_of_issue timestamp,
counter_value counter, PRIMARY KEY(stud_id,stud_name,
book_name, book_id, date_of_issue));

cqlsh:library> BEGIN COUNTER BATCH

        ... UPDATE library_info set counter_value +=1 where stud_id =
111 and stud_name = 'Manju' and book_name = 'Human Behaviour' and
book_id = '52e43' and date_of_issue = '2021-09-12';

        ... UPDATE library_info set counter_value +=1 where stud_id =
112 and stud_name = 'Kishore' and book_name = 'Engineering
Mathematics-1' and book_id = '52e44' and date_of_issue = '2021-04-10';

        ... UPDATE library_info set counter_value +=1 where stud_id =
113 and stud_name = 'Maitri' and book_name = 'Dan Brown and
book_id = '52e45' and date_of_issue = '2021-02-01';

        ... UPDATE library_info set counter_value +=1 where stud_id =
114 and stud_name = 'Ramesh' and book_name = 'EME' and book_id =
'52e46' and date_of_issue = '2021-04-03';

        ... APPLY BATCH;

cqlsh:library> SELECT * FROM library_info ;
```

```
 stud_id | stud_name | book_name                | book_id | date_of_issue
| counter_value

---------+-----------+-------------------------+---------+------------------------
--------+--------------

     114 |  Ramesh |   EME                     |   52e46 | 2021-04-02
18:30:00.000000+0000 |           1

     111 |  Manju  |   Human Behaviour          |   52e43 | 2021-09-11
18:30:00.000000+0000 |           1

     113 |  Maitri |   Dan Brown                |   52e45 | 2021-01-31
18:30:00.000000+0000 |           1

     112 |  Kishore|   Engineering Mathematics-1 |   52e44 | 2021-04-09
18:30:00.000000+0000 |           1


(4 rows)


cqlsh:library> UPDATE library_info set counter_value += 1 where
stud_id = 112 and stud_name = 'Kishore' and book_name = 'Engineering
Mathematics-1' and book_id = '52e44' and date_of_issue = '2021-04-09';
cqlsh:library> SELECT * FROM library_info ;


 stud_id | stud_name | book_name                | book_id | date_of_issue
| counter_value

---------+-----------+-------------------------+---------+------------------------
--------+--------------

     114 |  Ramesh |   EME                     |   52e46 | 2021-04-02
18:30:00.000000+0000 |           1
```

```
    111 |  Manoj  |   Human Behaviour          |   52e43 | 2021-09-11
18:30:00.000000+0000 |           1

    113 |  Maitri |   Dan Brown                |   52e45 | 2021-01-31
18:30:00.000000+0000 |           1

    112 |  Kishore|   Engineering Mathematics-1 |   52e44 | 2021-04-09
18:30:00.000000+0000 |           2


cqlsh:library> copy library_info(stud_id,stud_name, book_name,
book_id, date_of_issue,counter_value) to 'library_info.csv' ;

Using 11 child processes


Starting copy of library.library_info with columns [stud_id, stud_name,
book_name, book_id, date_of_issue, counter_value]. Processed: 6 rows;
Rate:      39 rows/s; Avg. rate:      39 rows/s 6 rows exported to 1 files in
0.165 seconds.


cqlsh:library> copy library_info(stud_id,stud_name, book_name,
book_id, date_of_issue,counter_value) from 'library_info.csv' ;

Using 11 child processes


Starting copy of library.library_info with columns [stud_id, stud_name,
book_name, book_id, date_of_issue, counter_value]. Processed: 6 rows;
Rate:      10 rows/s; Avg. rate:      15 rows/s 6 rows imported from 1 files
in 0.392 seconds (0 skipped).
```

LAB4:

Wordcount Program for a given text file using Hadoop

```
hduser@bmsce-Precision-T1700:~$ 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:~$ jps
7747 NodeManager
```

```
7045 DataNode
7416 ResourceManager
7257 SecondaryNameNode
6874 NameNode
7886 Jps
hduser@bmsce-Precision-T1700:~$ mkdir
mkdir: missing operand
Try 'mkdir --help' for more information.
hduser@bmsce-Precision-T1700:~$ hdfs dfs -mkdir/hadoop
-mkdir/hadoop: Unknown command
hduser@bmsce-Precision-T1700:~$ hdfs dfs -mkdir/lab6
-mkdir/lab6: Unknown command
hduser@bmsce-Precision-T1700:~$ hdfs dfs -mkdir /lab6
hduser@bmsce-Precision-T1700:~$ hadoop fs -ls /
Found 6 items
drwxr-xr-x   - hduser supergroup          0 2022-05-31 09:45 /ff
drwxr-xr-x   - hduser supergroup          0 2022-05-31 09:16 /j
drwxr-xr-x   - hduser supergroup          0 2022-06-01 09:31 /lab6
drwxr-xr-x   - hduser supergroup          0 2022-05-31 09:57 /ss
drwxrwxr-x   - hduser supergroup          0 2019-08-01 16:19 /tmp
drwxr-xr-x   - hduser supergroup          0 2019-08-01 16:03 /user
hduser@bmsce-Precision-T1700:~$ hdfs dfs -put
/home/hduser/Desktop/Welcome.txt/abc/WC.txt
put: `.': No such file or directory
```

```
hduser@bmsce-Precision-T1700:~$ hdfs dfs -put
/home/hduser/Desktop/Welcome.txt/abc/WC.txt

put: `.': No such file or directory

hduser@bmsce-Precision-T1700:~$ hdfs dfs -put
/home/hduser/Desktop/Welcome.txt /lab6/WC.txt

hduser@bmsce-Precision-T1700:~$ sudo nano xyz.txt

[sudo] password for hduser:

hduser@bmsce-Precision-T1700:~$ sudo nano xyz.txt

hduser@bmsce-Precision-T1700:~$ hadoop fs -copyFromlocal xyz.txt
/lab6

-copyFromlocal: Unknown command

hduser@bmsce-Precision-T1700:~$ hadoop fs -copyFromLocal xyz.txt
/lab6

hduser@bmsce-Precision-T1700:~$ hadoop fs -ls/lab6

-ls/lab6: Unknown command

hduser@bmsce-Precision-T1700:~$ hadoop fs -ls /lab6

Found 2 items

-rw-r--r--   1 hduser supergroup          0 2022-06-01 09:40 /lab6/WC.txt

-rw-r--r--   1 hduser supergroup         24 2022-06-01 09:45 /lab6/xyz.txt

hduser@bmsce-Precision-T1700:~$ hadoop fs -cat /lab6/xyz.txt

Hello My name is Anitej

hduser@bmsce-Precision-T1700:~$ hdfs dfs -get /lab6/xyz.txt
/home/hduser/Downloads/WWC.txt

get: `/home/hduser/Downloads/WWC.txt': File exists
```

```
hduser@bmsce-Precision-T1700:~$ hdfs dfs -get /lab6/WC.txt
/home/hduser/Downloads/WWC.txt

get: `/home/hduser/Downloads/WWC.txt': File exists

hduser@bmsce-Precision-T1700:~$ hdfs dfs -get /lab6/xyz.txt
/home/hduser/Downloads/WWC.txt

get: `/home/hduser/Downloads/WWC.txt': File exists

hduser@bmsce-Precision-T1700:~$ hdfs dfs -copyToLocal /lab6/xyz.xt
/home/hduser/Desktop

copyToLocal: `/lab6/xyz.xt': No such file or directory

hduser@bmsce-Precision-T1700:~$ hdfs dfs -copyToLocal /lab6/xyz.txt
/home/hduser/Desktop

hduser@bmsce-Precision-T1700:~$ hdfs dfs -cat /lab6/xyz.txt

Hello My name is Anitej

hduser@bmsce-Precision-T1700:~$ hdfs dfs -copyToLocal /lab6/xyz.txt
/home/hduser/Desktop

copyToLocal: `/home/hduser/Desktop/xyz.txt': File exists

hduser@bmsce-Precision-T1700:~$ hdoop fs -mv /lab6 /FFF

hdoop: command not found

hduser@bmsce-Precision-T1700:~$ hadoop fs -mv /lab6 /FFF

hduser@bmsce-Precision-T1700:~$ hadoop fs -ls /FFF

Found 2 items

-rw-r--r--   1 hduser supergroup          0 2022-06-01 09:40 /FFF/WC.txt

-rw-r--r--   1 hduser supergroup         24 2022-06-01 09:45 /FFF/xyz.txt

hduser@bmsce-Precision-T1700:~$ hadoop fs -cp /lab6/  /LLL

cp: `/lab6/': No such file or directory
```

```
hduser@bmsce-Precision-T1700:~$ hadoop fs -cp /CSE/  /LLL
cp: `/CSE/': No such file or directory
hduser@bmsce-Precision-T1700:~$ hadoop fs -cp /lab6/  /LLL
cp: `/lab6/': No such file or directory
hduser@bmsce-Precision-T1700:~$ hadoop fs -cp /lab6/ LLL
cp: `LLL': No such file or directory
```

LAB5:

Wordcount Program for a given text file using Map Reduce

Mapper Code: You have to copy paste this program into the WCMapper Java Class file.

```
// 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&lt;LongWritable,
Text, Text,

IntWritable&gt; {
// Map function
public void map(LongWritable key, Text value,
OutputCollector&lt;Text,
IntWritable&gt; output, Reporter rep) throws IOException
{
String line = value.toString();
// Splitting the line on spaces
for (String word : line.split(&quot; &quot;))
{
if (word.length() &gt; 0)
{
output.collect(new Text(word), new IntWritable(1));
} } } }
```

Reducer Code: You have to copy paste this program into the WCReducer Java Class file

```
// 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&lt;Text,
IntWritable, Text, IntWritable&gt; {

// Reduce function
public void reduce(Text key, Iterator&lt;IntWritable&gt; value,

OutputCollector&lt;Text, IntWritable&gt; 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: You have to copy paste this program into the WCDriver Java Class file.

```
// 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 &lt; 2)
{
System.out.println(&quot;Please give valid inputs&quot;);
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);
}
```

HDFS EXECUTION:

```
hduser@bmsce-Precision-T1700:~$ 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: namenode running as process 10473. Stop it first.
hduser@localhost's password:
```

```
localhost: datanode running as process 10644. Stop it first.
Starting secondary namenodes [0.0.0.0]
hduser@0.0.0.0's password:
0.0.0.0: secondarynamenode running as process 10857. Stop it first.
starting yarn daemons
resourcemanager running as process 9796. Stop it first.
hduser@localhost's password:
localhost: nodemanager running as process 10160. Stop it first.
hduser@bmsce-Precision-T1700:~$ jps
10160 NodeManager
7441 org.eclipse.equinox.launcher_1.5.600.v20191014-2022.jar
9796 ResourceManager
12692 org.eclipse.equinox.launcher_1.5.600.v20191014-2022.jar
10644 DataNode
10857 SecondaryNameNode
10473 NameNode
15100 Jps

hduser@bmsce-Precision-T1700:~$ hadoop fs -ls /
Found 10 items
drwxr-xr-x   - hduser supergroup          0 2019-10-23 09:52 /sample
drwxr-xr-x   - hduser supergroup          0 2019-10-23 10:33 /test
drwxr-xr-x   - hduser supergroup          0 2022-06-14 10:50 /tmp1
```

```
drwxr-xr-x   - hduser supergroup          0 2019-10-23 09:58 /output
drwxr-xr-x   - hduser supergroup          0 2022-06-15 10:27 /rgs
drwxr-xr-x   - hduser supergroup          0 2019-10-23 11:09 /stud
drwxr-xr-x   - hduser supergroup          0 2019-10-23 15:50 /testing
drwxrwxr-x   - hduser supergroup          0 2019-10-23 11:24 /tmp
drwxr-xr-x   - hduser supergroup          0 2019-08-01 16:03 /user

hduser@bmsce-Precision-T1700:~$ hadoop fs -mkdir /1BM19CS194


hduser@bmsce-Precision-T1700:~$ hadoop fs -copyFromLocal
/home/hduser/Desktop/sample1.txt /1BM19CS194/test.txt


hduser@bmsce-Precision-T1700:~$ hdfs dfs -cat /1BM19CS194/test.txt
hi my name is anitej
my age is twenty one
my country is india
my surname is prasad


hduser@bmsce-Precision-T1700:~$ hadoop jar
/home/hduser/Documents/wordCount.jar wordCount.WCDriver
/1BM19CS194/test.txt /1BM19CS194/output
22/06/15 10:27:53 INFO Configuration.deprecation: session.id is
deprecated. Instead, use dfs.metrics.session-id
22/06/15 10:27:53 INFO jvm.JvmMetrics: Initializing JVM Metrics with
processName=JobTracker, sessionId=
22/06/15 10:27:53 INFO jvm.JvmMetrics: Cannot initialize JVM
Metrics with processName=JobTracker, sessionId= - already initialized

22/06/15 10:27:53 WARN mapreduce.JobSubmitter: Hadoop command-
line option parsing not performed. Implement the Tool interface and
execute your application with ToolRunner to remedy this.

22/06/15 10:27:53 INFO mapred.FileInputFormat: Total input paths to
process : 1

22/06/15 10:27:53 INFO mapreduce.JobSubmitter: number of splits:1

22/06/15 10:27:53 INFO mapreduce.JobSubmitter: Submitting tokens
for job: job_local1115189753_0001

22/06/15 10:27:53 INFO mapreduce.Job: The url to track the job:
http://localhost:8080/

22/06/15 10:27:53 INFO mapred.LocalJobRunner: OutputCommitter set
in config null

22/06/15 10:27:53 INFO mapreduce.Job: Running job:
job_local1115189753_0001

22/06/15 10:27:53 INFO mapred.LocalJobRunner: OutputCommitter is
org.apache.hadoop.mapred.FileOutputCommitter

22/06/15 10:27:53 INFO mapred.LocalJobRunner: Waiting for map
tasks

22/06/15 10:27:53 INFO mapred.LocalJobRunner: Starting task:
attempt_local1115189753_0001_m_000000_0

22/06/15 10:27:53 INFO mapred.Task:  Using
ResourceCalculatorProcessTree : [ ]

22/06/15 10:27:53 INFO mapred.MapTask: Processing split:
hdfs://localhost:54310/rgs/test.txt:0+89

22/06/15 10:27:53 INFO mapred.MapTask: numReduceTasks: 1
```

```
22/06/15 10:27:54 INFO mapred.MapTask: (EQUATOR) 0 kvi
26214396(104857584)

22/06/15 10:27:54 INFO mapred.MapTask: mapreduce.task.io.sort.mb:
100

22/06/15 10:27:54 INFO mapred.MapTask: soft limit at 83886080

22/06/15 10:27:54 INFO mapred.MapTask: bufstart = 0; bufvoid =
104857600

22/06/15 10:27:54 INFO mapred.MapTask: kvstart = 26214396; length
= 6553600

22/06/15 10:27:54 INFO mapred.MapTask: Map output collector class =
org.apache.hadoop.mapred.MapTask$MapOutputBuffer

22/06/15 10:27:54 INFO mapred.LocalJobRunner:

22/06/15 10:27:54 INFO mapred.MapTask: Starting flush of map output

22/06/15 10:27:54 INFO mapred.MapTask: Spilling map output

22/06/15 10:27:54 INFO mapred.MapTask: bufstart = 0; bufend = 169;
bufvoid = 104857600

22/06/15 10:27:54 INFO mapred.MapTask: kvstart =
26214396(104857584); kvend = 26214320(104857280); length =
77/6553600

22/06/15 10:27:54 INFO mapred.MapTask: Finished spill 0

22/06/15 10:27:54 INFO mapred.Task:
Task:attempt_local1115189753_0001_m_000000_0 is done. And is in
the process of committing

22/06/15 10:27:54 INFO mapred.LocalJobRunner:
hdfs://localhost:54310/rgs/test.txt:0+89

22/06/15 10:27:54 INFO mapred.Task: Task
'attempt_local1115189753_0001_m_000000_0' done.
```

```
22/06/15 10:27:54 INFO mapred.LocalJobRunner: Finishing task:
attempt_local1115189753_0001_m_000000_0

22/06/15 10:27:54 INFO mapred.LocalJobRunner: map task executor
complete.

22/06/15 10:27:54 INFO mapred.LocalJobRunner: Waiting for reduce
tasks

22/06/15 10:27:54 INFO mapred.LocalJobRunner: Starting task:
attempt_local1115189753_0001_r_000000_0

22/06/15 10:27:54 INFO mapred.Task:  Using
ResourceCalculatorProcessTree : [ ]

22/06/15 10:27:54 INFO mapred.ReduceTask: Using
ShuffleConsumerPlugin:
org.apache.hadoop.mapreduce.task.reduce.Shuffle@1bc68cd5

22/06/15 10:27:54 INFO reduce.MergeManagerImpl: MergerManager:
memoryLimit=334338464, maxSingleShuffleLimit=83584616,
mergeThreshold=220663392, ioSortFactor=10,
memToMemMergeOutputsThreshold=10

22/06/15 10:27:54 INFO reduce.EventFetcher:
attempt_local1115189753_0001_r_000000_0 Thread started:
EventFetcher for fetching Map Completion Events

22/06/15 10:27:54 INFO reduce.LocalFetcher: localfetcher#1 about to
shuffle output of map attempt_local1115189753_0001_m_000000_0
decomp: 211 len: 215 to MEMORY

22/06/15 10:27:54 INFO reduce.InMemoryMapOutput: Read 211 bytes
from map-output for attempt_local1115189753_0001_m_000000_0

22/06/15 10:27:54 INFO reduce.MergeManagerImpl:
closeInMemoryFile -> map-output of size: 211,
inMemoryMapOutputs.size() -> 1, commitMemory -> 0, usedMemory -
>211
```

```
22/06/15 10:27:54 INFO reduce.EventFetcher: EventFetcher is interrupted.. Returning
22/06/15 10:27:54 INFO mapred.LocalJobRunner: 1 / 1 copied.
22/06/15 10:27:54 INFO reduce.MergeManagerImpl: finalMerge called with 1 in-memory map-outputs and 0 on-disk map-outputs
22/06/15 10:27:54 INFO mapred.Merger: Merging 1 sorted segments
22/06/15 10:27:54 INFO mapred.Merger: Down to the last merge-pass, with 1 segments left of total size: 205 bytes
22/06/15 10:27:54 INFO reduce.MergeManagerImpl: Merged 1 segments, 211 bytes to disk to satisfy reduce memory limit
22/06/15 10:27:54 INFO reduce.MergeManagerImpl: Merging 1 files, 215 bytes from disk
22/06/15 10:27:54 INFO reduce.MergeManagerImpl: Merging 0 segments, 0 bytes from memory into reduce
22/06/15 10:27:54 INFO mapred.Merger: Merging 1 sorted segments
22/06/15 10:27:54 INFO mapred.Merger: Down to the last merge-pass, with 1 segments left of total size: 205 bytes
22/06/15 10:27:54 INFO mapred.LocalJobRunner: 1 / 1 copied.
22/06/15 10:27:54 INFO mapred.Task: Task:attempt_local1115189753_0001_r_000000_0 is done. And is in the process of committing
22/06/15 10:27:54 INFO mapred.LocalJobRunner: 1 / 1 copied.
22/06/15 10:27:54 INFO mapred.Task: Task attempt_local1115189753_0001_r_000000_0 is allowed to commit now
22/06/15 10:27:54 INFO output.FileOutputCommitter: Saved output of task 'attempt_local1115189753_0001_r_000000_0' to
```

```
hdfs://localhost:54310/rgs/output/_temporary/0/task_local1115189753_
0001_r_000000

22/06/15 10:27:54 INFO mapred.LocalJobRunner: reduce > reduce

22/06/15 10:27:54 INFO mapred.Task: Task
'attempt_local1115189753_0001_r_000000_0' done.

22/06/15 10:27:54 INFO mapred.LocalJobRunner: Finishing task:
attempt_local1115189753_0001_r_000000_0

22/06/15 10:27:54 INFO mapred.LocalJobRunner: reduce task executor
complete.

22/06/15 10:27:54 INFO mapreduce.Job: Job
job_local1115189753_0001 running in uber mode : false

22/06/15 10:27:54 INFO mapreduce.Job:  map 100% reduce 100%

22/06/15 10:27:54 INFO mapreduce.Job: Job
job_local1115189753_0001 completed successfully

22/06/15 10:27:54 INFO mapreduce.Job: Counters: 38

        File System Counters

                FILE: Number of bytes read=8614

                FILE: Number of bytes written=510599

                FILE: Number of read operations=0

                FILE: Number of large read operations=0

                FILE: Number of write operations=0

                HDFS: Number of bytes read=178

                HDFS: Number of bytes written=69

                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=5
                Map output records=20
                Map output bytes=169
                Map output materialized bytes=215
                Input split bytes=87
                Combine input records=0
                Combine output records=0
                Reduce input groups=10
                Reduce shuffle bytes=215
                Reduce input records=20
                Reduce output records=10
                Spilled Records=40
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=1
                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=89
        File Output Format Counters
                Bytes Written=69
0
hduser@bmsce-Precision-T1700:~$ hdfs dfs -cat
/1BM19CS194/output/part-00000
age     1
anitej  1
country         1
india   1
is      4
my      4
name    1
one     1
prasad          1
surname         1
twenty 1
```

LAB6:

For a given Text file, create a Map Reduce program to sort the content in an alphabetic order listing only top ‘n’ maximum occurrence of words.

Driver-TopN.class

```
package samples.topn;
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(&quot;Usage: TopN &lt;in&gt; &lt;out&gt;&quot;);
System.exit(2);
}
Job job = Job.getInstance(conf);
job.setJobName(&quot;Top N&quot;);
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&lt;Object, Text, Text,
IntWritable&gt; {
private static final IntWritable one = new IntWritable(1);
private Text word = new Text();
private String tokens = &quot;[_|$#&lt;&gt;\\^=\\[\\]\\*/\\\\,;,.\\-
:()?!\&quot;&#39;]&quot;;
```

```
public void map(Object key, Text value, Mapper&lt;Object, Text, Text,
IntWritable&gt;.Context
context) throws IOException, InterruptedException {
String cleanLine =
value.toString().toLowerCase().replaceAll(this.tokens, &quot; &quot;);
StringTokenizer itr = new StringTokenizer(cleanLine);
while (itr.hasMoreTokens()) {
this.word.set(itr.nextToken().trim());
context.write(this.word, one);
}
}
}
}
```

TopNCombiner.class

```
package samples.topn;
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&lt;Text, IntWritable,
Text, IntWritable&gt; {
```

```
public void reduce(Text key, Iterable&lt;IntWritable&gt; values,
Reducer&lt;Text, IntWritable,

Text, IntWritable&gt;.Context context) throws IOException,
InterruptedException {

int sum = 0;

for (IntWritable val : values)

sum += val.get();

context.write(key, new IntWritable(sum));

}

}
```

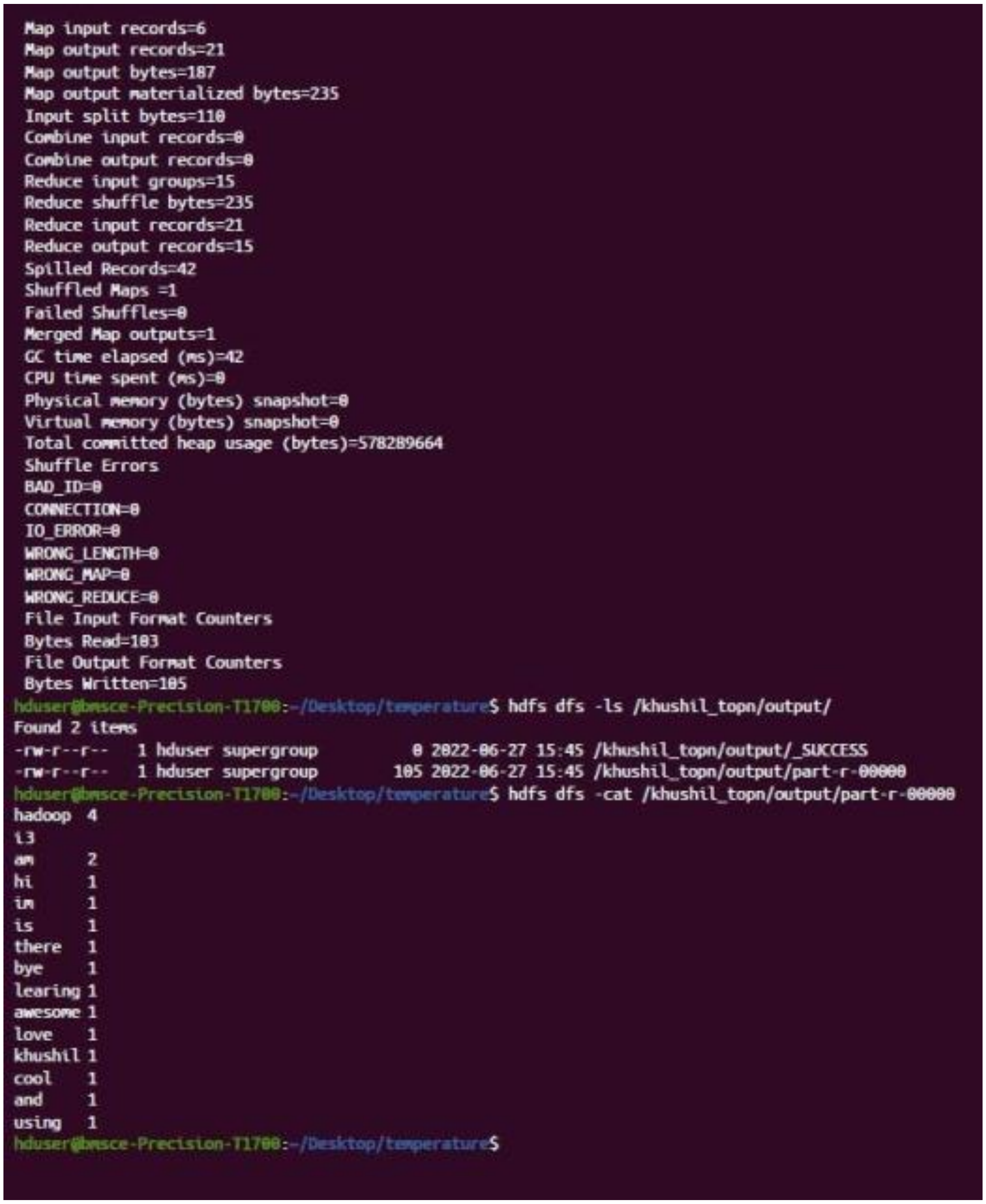
TopNMapper.class

```
package samples.topn;

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&lt;Object, Text, Text,
IntWritable&gt; {

private static final IntWritable one = new IntWritable(1);

private Text word = new Text();

private String tokens = &quot;[_|$#&lt;&gt;\\^=\\[\\]\\*/\\\\,;,.\\-
:()?!\&quot;&#39;]&quot;;

public void map(Object key, Text value, Mapper&lt;Object, Text, Text,
IntWritable&gt;.Context
context) throws IOException, InterruptedException {
String cleanLine =
value.toString().toLowerCase().replaceAll(this.tokens, &quot; &quot;);
StringTokenizer itr = new StringTokenizer(cleanLine);
while (itr.hasMoreTokens()) {
this.word.set(itr.nextToken().trim());
context.write(this.word, one);
}
}
}
```

TopNReducer.class

```
package samples.topn;
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&lt;Text, IntWritable, Text,
IntWritable&gt; {
private Map&lt;Text, IntWritable&gt; countMap = new
HashMap&lt;&gt;();
public void reduce(Text key, Iterable&lt;IntWritable&gt; values,
Reducer&lt;Text, IntWritable,

Text, IntWritable&gt;.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&lt;Text, IntWritable, Text,
IntWritable&gt;.Context context)

throws IOException, InterruptedException {

Map&lt;Text, IntWritable&gt; sortedMap =
MiscUtils.sortByValues(this.countMap);

int counter = 0;

for (Text key : sortedMap.keySet()) {

if (counter++ == 20)

break;

context.write(key, sortedMap.get(key));

}

}

}
```

HDFS EXECTUION:



```
 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$
```

LAB7:

From the following link extract the weather data

https://github.com/tomwhite/hadoop-book/tree/master/input/ncdc/all

Create a Map Reduce program to

a) find average temperature for each year from NCDC data set.

AverageDriver

```
package temp;
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(&quot;Please Enter the input and output parameters&quot;);
System.exit(-1);
}
Job job = new Job();
job.setJarByClass(AverageDriver.class);
job.setJobName(&quot;Max temperature&quot;);
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

```
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&lt;LongWritable, Text,
Text, IntWritable&gt; {
public static final int MISSING = 9999;
public void map(LongWritable key, Text value,
Mapper&lt;LongWritable, Text, Text,
IntWritable&gt;.Context context) throws IOException,
InterruptedException {
int temperature;
String line = value.toString();
String year = line.substring(15, 19);
if (line.charAt(87) == &#39;+&#39;) {
temperature = Integer.parseInt(line.substring(88, 92));
} else {
temperature = Integer.parseInt(line.substring(87, 92));
}
String quality = line.substring(92, 93);
if (temperature != 9999 &amp;&amp;
quality.matches(&quot;[01459]&quot;))
context.write(new Text(year), new IntWritable(temperature));
}
}
```

AverageReducer

```
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&lt;Text, IntWritable,
Text, IntWritable&gt; {

public void reduce(Text key, Iterable&lt;IntWritable&gt; values,
Reducer&lt;Text, IntWritable,

Text, IntWritable&gt;.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));

}

}
```

HDFS EXECUTION:

```
C:\WINDOWS\system32>cd c:\hadoop_new\sbin

c:\hadoop_new\sbin>start-all.cmd

This script is Deprecated. Instead use start-dfs.cmd and start-yarn.cmd
starting yarn daemons

c:\hadoop_new\sbin>cd c:\hadoop_new\share\hadoop\mapreduce
```

```
c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -mkdir /tempAverage

c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -put

E:\Desktop\temp1.txt \tempAverage

c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -put

E:\Desktop\temp2.txt \tempAverage

c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -cat
\tempAverage\temp1.txt

0067011990999991950051507004+68750+023550FM-
12+038299999V0203301N0067
1220001CN9999999N9+00001+99999999999

0043011990999991950051512004+68750+023550FM-
12+038299999V0203201N0067
1220001CN9999999N9+00221+99999999999

0043011990999991950051518004+68750+023550FM-
12+038299999V0203201N0026

1220001CN9999999N9-00111+99999999999

0043012650999991949032412004+62300+010750FM-
12+048599999V0202701N0046
1220001CN0500001N9+01111+99999999999

0043012650999991949032418004+62300+010750FM-
12+048599999V0202701N0046

1220001CN0500001N9+00781+99999999999
c:\hadoop_new\share\hadoop\mapreduce>hadoop jar

E:\Desktop\temperatureAverage.jar temperature.AverageDriver
\tempAverage

\tempAverageOutput
```

```
2022-06-22 14:31:05,036 INFO client.RMProxy: Connecting to
ResourceManager at

/0.0.0.0:8032

2022-06-22 14:31:07,049 WARN mapreduce.JobResourceUploader:
Hadoop command-line option parsing not performed. Implement the
Tool interface and execute your application with ToolRunner to remedy
this.

2022-06-22 14:31:07,159 INFO mapreduce.JobResourceUploader:
Disabling Erasure

Coding for path: /tmp/hadoop-
yarn/staging/Admin/.staging/job_1620636818881_0001 2021-05-10
14:31:08,149 INFO input.FileInputFormat: Total input files to process :
2 2021-05-10 14:31:08,697 INFO mapreduce.JobSubmitter: number of
splits:2 2021-05-10 14:31:09,122 INFO Configuration.deprecation:
yarn.resourcemanager.system-metrics-publisher.enabled is deprecated.
Instead, use yarn.system-metrics-publisher.enabled

2022-06-22 14:31:10,026 INFO mapreduce.JobSubmitter: Submitting
tokens for job: job_1620636818881_0001

2022-06-22 14:31:10,031 INFO mapreduce.JobSubmitter: Executing
with tokens: []

2022-06-22 14:31:10,923 INFO conf.Configuration: resource-types.xml
not found 2022-06-22 14:31:10,924 INFO resource.ResourceUtils:
Unable to find 'resource-types.xml'.

2022-06-22 14:31:11,755 INFO impl.YarnClientImpl: Submitted
application application_1620636818881_0001

2022-06-22 14:31:12,063 INFO mapreduce.Job: The url to track the job:
http://DESKTOP-
CK14PFH:8088/proxy/application_1620636818881_0001/ 2021-05-10
```

```
14:31:12,068 INFO mapreduce.Job: Running job:
job_1620636818881_0001

2022-06-22 14:31:43,855 INFO mapreduce.Job: Job
job_1620636818881_0001 running in uber mode : false

2022-06-22 14:31:43,876 INFO mapreduce.Job:  map 0% reduce 0%

2022-06-22 14:32:25,710 INFO mapreduce.Job:  map 50% reduce 0%

2022-06-22 14:32:26,732 INFO mapreduce.Job:  map 100% reduce 0%

2022-06-22 14:32:56,193 INFO mapreduce.Job:  map 100% reduce
100% 2022-06-22 14:33:04,369 INFO mapreduce.Job: Job
job_1620636818881_0001 completed successfully

2022-06-22 14:33:04,843 INFO mapreduce.Job: Counters: 53

File System Counters

FILE: Number of bytes read=72265

FILE: Number of bytes written=784759

FILE: Number of read operations=0

FILE: Number of large read operations=0

FILE: Number of write operations=0

HDFS: Number of bytes read=895504 HDFS: Number of bytes
written=23

HDFS: Number of read operations=11

HDFS: Number of large read operations=0

HDFS: Number of write operations=2

Job Counters

Launched map tasks=2

Launched reduce tasks=1
```

```
Data-local map tasks=2
Total time spent by all maps in occupied slots (ms)=79302
Total time spent by all reduces in occupied slots (ms)=27416
Total time spent by all map tasks (ms)=79302
Total time spent by all reduce tasks (ms)=27416
Total vcore-milliseconds taken by all map tasks=79302
Total vcore-milliseconds taken by all reduce tasks=27416
Total megabyte-milliseconds taken by all map tasks=81205248
Total megabyte-milliseconds taken by all reduce tasks=28073984
Map-Reduce Framework
Map input records=6570
Map output records=6569
Map output bytes=59121
Map output materialized bytes=72271
Input split bytes=216
Combine input records=0
Combine output records=0
Reduce input groups=3
Reduce shuffle bytes=72271
Reduce input records=6569
Reduce output records=3
Spilled Records=13138
Shuffled Maps =2
```

```
Failed Shuffles=0
Merged Map outputs=2
GC time elapsed (ms)=350
CPU time spent (ms)=7667
Physical memory (bytes) snapshot=689139712
Virtual memory (bytes) snapshot=854798336
Total committed heap usage (bytes)=381157376
Peak Map Physical memory (bytes)=250327040
Peak Map Virtual memory (bytes)=309706752
Peak Reduce Physical memory (bytes)=190980096
Peak Reduce Virtual memory (bytes)=244252672
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=895288
File Output Format Counters
Bytes Written=23
c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -ls
\tempAverageOutput
```

```
Found 2 items
-rw-r--r--   1 Admin supergroup          0 2022-06-22 14:32
/tempAverageOutput/_SUCCESS
-rw-r--r--   1 Admin supergroup         23 2022-06-22 14:32
/tempAverageOutput/part-r-00000
c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -cat
\tempAverageOutput\part-r-00000
1901	46
1949 94
1950 3
```

LAB8:

Find the mean max temperature for every month

MeanMax

MeanMaxDriver.class

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

MeanMaxMapper.class

```
package meanmax;
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 MeanMaxMapper extends Mapper&lt;LongWritable, Text,
Text, IntWritable&gt; {
public static final int MISSING = 9999;
public void map(LongWritable key, Text value,
Mapper&lt;LongWritable, Text, Text,
IntWritable&gt;.Context context) throws IOException,
InterruptedException {
int temperature;
String line = value.toString();
String month = line.substring(19, 21);
if (line.charAt(87) == &#39;+&#39;) {
temperature = Integer.parseInt(line.substring(88, 92));
} else {
temperature = Integer.parseInt(line.substring(87, 92));
}
String quality = line.substring(92, 93);
if (temperature != 9999 &amp;&amp;
quality.matches(&quot;[01459]&quot;))
context.write(new Text(month), new IntWritable(temperature));
}
}
```

MeanMaxReducer.class

```
package meanmax;
import java.io.IOException;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;
public class MeanMaxReducer extends Reducer&lt;Text, IntWritable,
Text, IntWritable&gt; {
public void reduce(Text key, Iterable&lt;IntWritable&gt; values,
Reducer&lt;Text, IntWritable,
Text, IntWritable&gt;.Context context) throws IOException,
InterruptedException {
int max_temp = 0;
int total_temp = 0;
int count = 0;

int days = 0;
for (IntWritable value : values) {
int temp = value.get();
if (temp &gt; max_temp)
max_temp = temp;
count++;
if (count == 3) {
total_temp += max_temp;
max_temp = 0;
count = 0;
days++;
}
}
context.write(key, new IntWritable(total_temp / days));
}
}
```

HDFS EXECUTION:

```
c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -mkdir /tempMax
c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -put
E:\Desktop\temp1.txt \tempMax
c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -put
E:\Desktop\temp2.txt \tempMax
c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -cat
\tempMax\temp1.txt
0067011990999991950051507004+68750+023550FM-
12+038299999V0203301N0067
1220001CN9999999N9+00001+99999999999
0043011990999991950051512004+68750+023550FM-
12+038299999V0203201N0067
1220001CN9999999N9+00221+99999999999
0043011990999991950051518004+68750+023550FM-
12+038299999V0203201N0026
1220001CN9999999N9-00111+99999999999
0043012650999991949032412004+62300+010750FM-
12+048599999V0202701N0046
1220001CN0500001N9+01111+99999999999
0043012650999991949032418004+62300+010750FM-
12+048599999V0202701N0046
1220001CN0500001N9+00781+99999999999
c:\hadoop_new\share\hadoop\mapreduce>hadoop jar
```

```
E:\Desktop\temperatureMax.jar temperatureMax.TempDriver \tempMax
\tempMaxOutput

2022-06-22 15:19:31,366 INFO client.RMProxy: Connecting to
ResourceManager at

/0.0.0.0:8032

2022-06-22 15:19:33,482 WARN mapreduce.JobResourceUploader:
Hadoop command-line option parsing not performed. Implement the
Tool interface and execute your application with ToolRunner to remedy
this.

2022-06-22 15:19:33,591 INFO mapreduce.JobResourceUploader:
Disabling Erasure

Coding for path: /tmp/hadoop-
yarn/staging/Admin/.staging/job_1620636818881_0003

2022-06-22 15:19:34,660 INFO input.FileInputFormat: Total input files
to process : 2 2022-06-22 15:19:35,250 INFO mapreduce.JobSubmitter:
number of splits:2 2022-06-22 15:19:35,729 INFO
Configuration.deprecation: yarn.resourcemanager.system-metrics-
publisher.enabled is deprecated. Instead, use yarn.system-metrics-
publisher.enabled

2022-06-22 15:19:36,334 INFO mapreduce.JobSubmitter: Submitting
tokens for job: job_1620636818881_0003

2022-06-22 15:19:36,337 INFO mapreduce.JobSubmitter: Executing
with tokens: []

2022-06-22 15:19:36,859 INFO conf.Configuration: resource-types.xml
not found 2022-06-22 15:19:36,863 INFO resource.ResourceUtils:
Unable to find 'resource-types.xml'.

2022-06-22 15:19:37,153 INFO impl.YarnClientImpl: Submitted
application application_1620636818881_0003
```

```
2022-06-22 15:19:37,287 INFO mapreduce.Job: The url to track the job:
http://DESKTOP-
CK14PFH:8088/proxy/application_1620636818881_0003/ 2022-06-22
15:19:37,290 INFO mapreduce.Job: Running job:
job_1620636818881_0003

2022-06-22 15:20:03,295 INFO mapreduce.Job: Job
job_1620636818881_0003 running in uber mode : false

2022-06-22 15:20:03,327 INFO mapreduce.Job:  map 0% reduce 0%

2022-06-22 15:20:26,140 INFO mapreduce.Job:  map 100% reduce 0%

2022-06-22 15:20:55,633 INFO mapreduce.Job:  map 100% reduce
100% 2021-05-2022-06-22,752 INFO mapreduce.Job: Job
job_1620636818881_0003 completed successfully

2022-06-22 15:21:03,029 INFO mapreduce.Job: Counters: 53

File System Counters

FILE: Number of bytes read=59127

FILE: Number of bytes written=758459

FILE: Number of read operations=0

FILE: Number of large read operations=0

FILE: Number of write operations=0

HDFS: Number of bytes read=895496 HDFS: Number of bytes
written=81

HDFS: Number of read operations=11

HDFS: Number of large read operations=0

HDFS: Number of write operations=2

Job Counters

Launched map tasks=2
Launched reduce tasks=1
Data-local map tasks=2
Total time spent by all maps in occupied slots (ms)=40099
Total time spent by all reduces in occupied slots (ms)=26572
Total time spent by all map tasks (ms)=40099
Total time spent by all reduce tasks (ms)=26572
Total vcore-milliseconds taken by all map tasks=40099
Total vcore-milliseconds taken by all reduce tasks=26572
Total megabyte-milliseconds taken by all map tasks=41061376
Total megabyte-milliseconds taken by all reduce tasks=27209728
Map-Reduce Framework
Map input records=6570
Map output records=6569
Map output bytes=45983
Map output materialized bytes=59133
Input split bytes=208
Combine input records=0
Combine output records=0
Reduce input groups=12
Reduce shuffle bytes=59133
Reduce input records=6569
Reduce output records=12
Spilled Records=13138
```

```
Shuffled Maps =2
Failed Shuffles=0
Merged Map outputs=2
GC time elapsed (ms)=368
CPU time spent (ms)=14277
Physical memory (bytes) snapshot=654815232
Virtual memory (bytes) snapshot=1096699904
Total committed heap usage (bytes)=554696704
Peak Map Physical memory (bytes)=245723136
Peak Map Virtual memory (bytes)=432418816
Peak Reduce Physical memory (bytes)=186056704
Peak Reduce Virtual memory (bytes)=232816640
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=895288
File Output Format Counters
Bytes Written=81
```

```
c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -ls \tempMaxOutput

Found 2 items

-rw-r--r--   1 Admin supergroup          0 2022-06-22 15:20
/tempMaxOutput/_SUCCESS

-rw-r--r--   1 Admin supergroup         81 2022-06-22 15:20
/tempMaxOutput/part-r-00000
c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -cat
\tempMaxOutput\part-r-00000

01    44

02    17

03    111

04    194

05    256

06    278

07    317

08    283

09    211

10    156

11    89

12    117
```

LAB9:

Join Operation using MapReduce

```
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 {

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

```
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);
        }
    }
}
```

Posts.java

```
import java.io.IOException;

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;
}

@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);
    }
  }
}

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

// vv TextPairFirstComparator
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);
  }
}
// ^^ TextPairFirstComparator

// vv TextPair
}
```

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]));
    }
}
```

HDFS EXECUTION:

```
c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -mkdir /posts

c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -put

E:\Desktop\sampleposts.tsv \posts

c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -mkdir /users

c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -put

E:\Desktop\sampleusers.tsv \users

c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -cat
\posts\sampleposts.tsv

"2312" "Feedback on Audio Quality" "cs101 production audio"

"100005361" "<p>We are looking for feedback on the audio in our
videos. Tell

us what you think and try to be as <em>specific</em> as possible.</p>"

"question"

"\N" "\N" "2012-02-23 00:28:02.321344+00" "2" "" "\N"

"201398145" "2014-01-14 17:18:35.613939+00" "2960" "\N" "\N"
"524"

"f"

"2014856" "" "cs101 " "100022094" "<p>I also would like to know

the answer to this question. An 'open exam' sounds great, but on the
other hand

it also seems pretty easy to cheat now: solutions have been posted and
anybody

only interested in a certificate wouldn't have much of a problem getting
the

highest distinction. So where is the catch??</p>" "answer" "2014706"
```

```
"2014706" "2012-07-01 10:32:36.302782+00" "0" "" "\N"
"100022094" "2012-07-01 10:32:36.302782+00" "2020501" "\N" "\N"
"0" "f"
"2004004" "" "cs101 " "100018705" "<p>But then why even the
new variable q? Why not just modify the variable p?</p>" "comment"
"2003997" "2003993" "2012-05-03 21:07:52.028935+00" "2" ""
"\N" "100018705" "2012-05-03 21:07:52.028935+00" "2005150" "\N"
"\N" "0" "f"
c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -cat
\users\sampleusers.tsv
"100006402" "18" "0" "0" "0"
"100022094" "6354" "4" "12" "50"
"100018705" "76" "0" "3" "4"
"100005361" "36134" "73" "220" "333"
c:\hadoop_new\share\hadoop\mapreduce>hadoop jar E:\Desktop\Join.jar
JoinDriver \join \users \joinOutput
2022-06-12 12:28:44,441 INFO client.RMProxy: Connecting to
ResourceManager at /0.0.0.0:8032
2022-07-03 12:28:45,518 INFO client.RMProxy: Connecting to
ResourceManager at /0.0.0.0:8032
2022-07-03 12:28:46,975 INFO mapreduce.JobResourceUploader:
Disabling
Erasure Coding for path:
```

```
/tmp/hadoop-yarn/staging/Admin/.staging/job_1623480742672_0001
2022-07-
03 12:28:47,543 INFO mapred.FileInputFormat: Total input files to
process : 1
2022-07-03 12:28:47,635 INFO mapred.FileInputFormat: Total input
files to
process : 1
2022-07-03 12:28:48,092 INFO mapreduce.JobSubmitter: number of
splits:4
2022-07-03 12:28:53,031 INFO Configuration.deprecation:
yarn.resourcemanager.system-metrics-publisher.enabled is deprecated.
Instead,
use yarn.system-metrics-publisher.enabled
2022-07-03 12:28:53,944 INFO mapreduce.JobSubmitter: Submitting
tokens for
job: job_1623480742672_0001
2022-07-03 12:28:53,947 INFO mapreduce.JobSubmitter: Executing
with tokens:
[]
2022-07-03 12:28:54,424 INFO conf.Configuration: resource-types.xml
not found
2022-07-03 12:28:54,426 INFO resource.ResourceUtils: Unable to find
'resource-types.xml'.
2022-07-03 12:28:55,154 INFO impl.YarnClientImpl: Submitted
application
application_1623480742672_0001
```

```
2022-07-03 12:28:55,293 INFO mapreduce.Job: The url to track the job:

http://DESKTOP-
CK14PFH:8088/proxy/application_1623480742672_0001/

2022-07-03 12:28:55,295 INFO mapreduce.Job: Running job:

job_1623480742672_0001

2022-07-03 12:29:19,847 INFO mapreduce.Job: Job
job_1623480742672_0001

running in uber mode : false

2022-07-03 12:29:19,874 INFO mapreduce.Job: map 0% reduce 0%

2022-07-03 12:31:53,514 INFO mapreduce.Job: map 67% reduce 0%
2022-07-

03 12:31:59,518 INFO mapreduce.Job: map 83% reduce 0%

2022-07-03 12:32:00,667 INFO mapreduce.Job: map 100% reduce 0%

2022-07-03 12:33:23,194 INFO mapreduce.Job: map 100% reduce
100% 2022-

07-03 12:33:32,307 INFO mapreduce.Job: Job
job_1623480742672_0001

completed successfully

2022-07-03 12:33:32,532 INFO mapreduce.Job: Counters: 53

File System Counters

FILE: Number of bytes read=155

FILE: Number of bytes written=1071678

FILE: Number of read operations=0

FILE: Number of large read operations=0

FILE: Number of write operations=0
```

```
HDFS: Number of bytes read=2595
HDFS: Number of bytes written=71
HDFS: Number of read operations=17
HDFS: Number of large read operations=0
HDFS: Number of write operations=2
Job Counters
Launched map tasks=4
Launched reduce tasks=1
Data-local map tasks=4
Total time spent by all maps in occupied slots (ms)=630042
Total time spent by all reduces in occupied slots (ms)=80087
Total time spent by all map tasks (ms)=630042
Total time spent by all reduce tasks (ms)=80087
Total vcore-milliseconds taken by all map tasks=630042
Total vcore-milliseconds taken by all reduce tasks=80087
Total megabyte-milliseconds taken by all map tasks=645163008
Total megabyte-milliseconds taken by all reduce tasks=82009088
Map-Reduce Framework
Map input records=7
Map output records=7
Map output bytes=135
Map output materialized bytes=173
Input split bytes=750
```

```
Combine input records=0
Combine output records=0
Reduce input groups=4
Reduce shuffle bytes=173
Reduce input records=7
Reduce output records=3
Spilled Records=14
Shuffled Maps =4
Failed Shuffles=0
Merged Map outputs=4
GC time elapsed (ms)=903
CPU time spent (ms)=15864
Physical memory (bytes) snapshot=990265344
Virtual memory (bytes) snapshot=1415651328
Total committed heap usage (bytes)=663224320
Peak Map Physical memory (bytes)=219955200
Peak Map Virtual memory (bytes)=296644608
Peak Reduce Physical memory (bytes)=204709888
Peak Reduce Virtual memory (bytes)=246579200
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=71
c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -ls \joinOutput
Found 2 items
-rw-r--r-- 1 Admin supergroup 0 2022-07-03 12:33 /joinOutput/_SUCCESS
-rw-r--r-- 1 Admin supergroup 71 2022-07-03 12:33 /joinOutput/part-00000
c:\hadoop_new\share\hadoop\mapreduce>hdfs dfs -cat \joinOutput\part-00000
"100005361" "2" "36134"
"100018705" "2" "76"
"100022094" "0" "6354"
```

LAB10:

Scala Program for word count

```
object WordCount {

def main(args: Array[String]): Unit = {

val map = ReadFile.readFile()
ReadFile.printContent(map)

}

}
```

Output:

```
scala> result.foreach(println)
(working,1)
(BigData,2)
(is,1)
(Technologies,1)
(Anish,2)
(on,1)
(Hello,1)
```

LAB11:

Wordcount greater than 4 using 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()
}}
```

Output:

```
In,
5
I,
6
```