VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

Big Data Analytics (23CS6PEBDA)

Submitted by

Neelvani Varsha Vittal (1BM23CS412)

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
Feb-2024 to July-2024

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**" carried out by **Neelvani Varsha Vittal (1BM23CS412),** 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 2024. The Lab report has been approved as it satisfies the academic requirements in respect of a **Big Data Analytics – (23CS6PEBDA)** work prescribed for the said degree.

Ramya K M
Assistant Professor
Department of CSE
BMSCE, Bengaluru

Dr. Kavitha Sooda
Professor and Head
Department of CSE
BMSCE, Bengaluru

`

Index Sheet

9	Using RDD and FlatMap count how many times each word appears in a file and write out a list of words whose count is strictly greater than 4 using Spark.	24
10	Write a simple streaming program in Spark to receive text data streams on a particular port, perform basic text cleaning (like white space removal, stop words removal, lemmatization, etc.), and print the cleaned text on the screen. (Open Ended Question).	25

Course Outcome

CO1	Apply the concept of NoSQL, Hadoop or Spark for a given task
CO2	Analyse big data analytics mechanisms that can be applied to obtain solution for a given problem.
CO3	Design and implement solutions using data analytics mechanisms for a given problem.

Github Link: https://github.com/NeelvaniVarsha/BDALab.git

Lab 1
MongoDB- CRUD Operations Demonstration (Practice and Self Study)

```
Atlas atlas-11p8k4-shard-0 [primary] myDB> use studentDB;
switched to db studentDB
Atlas atlas-11p8k4-shard-0 [primary] studentDB> show collections;

Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.createCollection("Student");
{ ok: 1 }
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.drop();
true
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.createCollection("Student");
{ ok: 1 }
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.insert({_id:1, StudName:"Michelle Jacintha", Grade:"VII", Hobbies:"Intern
et Surfing"});
{ acknowledged: true, insertedIds: { '0': 1 } }
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.insert({_id:2, StudName:"Lando Norris", Grade:"VII", Hobbies:"Racing"});
{ acknowledged: true, insertedIds: { '0': 2 } }
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.insert({_id:3, StudName:"Aryan David", Grade:"VII", Hobbies:"Skating"});
{ acknowledged: true, insertedIds: { '0': 3 } }
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.insert({_id:4, StudName:"Gukesh D", Grade:"VII", Hobbies:"Chess"});
{ acknowledged: true, insertedIds: { '0': 4 } }
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find();
[
  {
    _id: 1,
    StudName: 'Michelle Jacintha',
    Grade: 'VII',
    Hobbies: 'Internet Surfing'
  },
  { _id: 2, StudName: 'Lando Norris', Grade: 'VII', Hobbies: 'Racing' },
  { _id: 3, StudName: 'Aryan David', Grade: 'VII', Hobbies: 'Skating' },
  { _id: 4, StudName: 'Gukesh D', Grade: 'VII', Hobbies: 'Chess' }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find({StudName:"Aryan David"});
[
  { _id: 3, StudName: 'Aryan David', Grade: 'VII', Hobbies: 'Skating' }
]
```

```
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find({},{StudName:1, Grade:1, _id:0});
[
  { StudName: 'Michelle Jacintha', Grade: 'VII' },
  { StudName: 'Lando Norris', Grade: 'VII' },
  { StudName: 'Aryan David', Grade: 'VII' },
  { StudName: 'Gukesh D', Grade: 'VII' }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find({},{StudName:1, Grade:1, _id:2});
[
  { _id: 1, StudName: 'Michelle Jacintha', Grade: 'VII' },
  { _id: 2, StudName: 'Lando Norris', Grade: 'VII' },
  { _id: 3, StudName: 'Aryan David', Grade: 'VII' },
  { _id: 4, StudName: 'Gukesh D', Grade: 'VII' }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find({_id:2},{StudName:1, Grade:1, _id:1});
[ { _id: 2, StudName: 'Lando Norris', Grade: 'VII' } ]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find({Grade:{$eq:"VII"}}).pretty();
[
  {
    _id: 1,
    StudName: 'Michelle Jacintha',
    Grade: 'VII',
    Hobbies: 'Internet Surfing'
  },
  { _id: 2, StudName: 'Lando Norris', Grade: 'VII', Hobbies: 'Racing' },
  { _id: 3, StudName: 'Aryan David', Grade: 'VII', Hobbies: 'Skating' },
  { _id: 4, StudName: 'Gukesh D', Grade: 'VII', Hobbies: 'Chess' }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find({Hobbies:{$in:['Chess', 'Skating']}}).pretty();
[
  { _id: 3, StudName: 'Aryan David', Grade: 'VII', Hobbies: 'Skating' },
  { _id: 4, StudName: 'Gukesh D', Grade: 'VII', Hobbies: 'Chess' }
]
```

```
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find({StudName:/^M/}).pretty();
[
  {
    _id: 1,
    StudName: 'Michelle Jacintha',
    Grade: 'VII',
    Hobbies: 'Internet Surfing'
  }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find({StudName:/e/}).pretty();
[
  {
    _id: 1,
    StudName: 'Michelle Jacintha',
    Grade: 'VII',
    Hobbies: 'Internet Surfing'
  },
  { _id: 4, StudName: 'Gukesh D', Grade: 'VII', Hobbies: 'Chess' }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find({Hobbies:{$nin:['Chess', 'Skating']}}).pretty();
[
  {
    _id: 1,
    StudName: 'Michelle Jacintha',
    Grade: 'VII',
    Hobbies: 'Internet Surfing'
  },
  { _id: 2, StudName: 'Lando Norris', Grade: 'VII', Hobbies: 'Racing' }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find({StudName:/s$/}).pretty();
[
  { _id: 2, StudName: 'Lando Norris', Grade: 'VII', Hobbies: 'Racing' }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find({$or:[{_id:3}, {_id:4}]});
[
  { _id: 3, StudName: 'Aryan David', Grade: 'VII', Hobbies: 'Skating' },
  { _id: 4, StudName: 'Gukesh D', Grade: 'VII', Hobbies: 'Chess' }
]
```

```
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.update({_id:3}, {$set:{Location:null}});
DeprecationWarning: Collection.update() is deprecated. Use updateOne, updateMany, or bulkWrite.
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find({Location:{$eq:null}});
[
  {
    _id: 1,
    StudName: 'Michelle Jacintha',
    Grade: 'VII',
    Hobbies: 'Internet Surfing'
  },
  { _id: 2, StudName: 'Lando Norris', Grade: 'VII', Hobbies: 'Racing' },
  {
    _id: 3,
    StudName: 'Aryan David',
    Grade: 'VII',
    Hobbies: 'Skating',
    Location: null
  },
  { _id: 4, StudName: 'Gukesh D', Grade: 'VII', Hobbies: 'Chess' }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.update({_id:4}, {$set:{Location:null}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

```
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find({Location:{$eq:null, $exists:true}});
[
  {
    _id: 3,
    StudName: 'Aryan David',
    Grade: 'VII',
    Hobbies: 'Skating',
    Location: null
  },
  {
    _id: 4,
    StudName: 'Gukesh D',
    Grade: 'VII',
    Hobbies: 'Chess',
    Location: null
  }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.update({_id:4}, {$unset:{Location:null}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.update({_id:3}, {$unset:{Location:null}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find({Location:{$eq:null, $exists:true}});

Atlas atlas-11p8k4-shard-0 [primary] studentDB>
```

```
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.count();
DeprecationWarning: Collection.count() is deprecated. Use countDocuments or estimatedDocumentCount.
4
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.count({Grade:"VII"});
4
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find().sort({StudName:1}).pretty();
[
  { _id: 3, StudName: 'Aryan David', Grade: 'VII', Hobbies: 'Skating' },
  { _id: 4, StudName: 'Gukesh D', Grade: 'VII', Hobbies: 'Chess' },
  { _id: 2, StudName: 'Lando Norris', Grade: 'VII', Hobbies: 'Racing' },
  {
    _id: 1,
    StudName: 'Michelle Jacintha',
    Grade: 'VII',
    Hobbies: 'Internet Surfing'
  }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find().sort({StudName:-1}).pretty();
[
  {
    _id: 1,
    StudName: 'Michelle Jacintha',
    Grade: 'VII',
    Hobbies: 'Internet Surfing'
  },
  { _id: 2, StudName: 'Lando Norris', Grade: 'VII', Hobbies: 'Racing' },
  { _id: 4, StudName: 'Gukesh D', Grade: 'VII', Hobbies: 'Chess' },
  { _id: 3, StudName: 'Aryan David', Grade: 'VII', Hobbies: 'Skating' }
]
```

```
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find().sort({Grade:1,Hobbies:-1}).pretty();
[
  { _id: 3, StudName: 'Aryan David', Grade: 'VII', Hobbies: 'Skating' },
  { _id: 2, StudName: 'Lando Norris', Grade: 'VII', Hobbies: 'Racing' },
  {
    _id: 1,
    StudName: 'Michelle Jacintha',
    Grade: 'VII',
    Hobbies: 'Internet Surfing'
  },
  { _id: 4, StudName: 'Gukesh D', Grade: 'VII', Hobbies: 'Chess' }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find().sort({Grade:1,Hobbies:1}).pretty();
[
  { _id: 4, StudName: 'Gukesh D', Grade: 'VII', Hobbies: 'Chess' },
  {
    _id: 1,
    StudName: 'Michelle Jacintha',
    Grade: 'VII',
    Hobbies: 'Internet Surfing'
  },
  { _id: 2, StudName: 'Lando Norris', Grade: 'VII', Hobbies: 'Racing' },
  { _id: 3, StudName: 'Aryan David', Grade: 'VII', Hobbies: 'Skating' }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find().skip(2).pretty();
[
  { _id: 3, StudName: 'Aryan David', Grade: 'VII', Hobbies: 'Skating' },
  { _id: 4, StudName: 'Gukesh D', Grade: 'VII', Hobbies: 'Chess' }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find().skip(1).pretty().sort({StudName:1})
[
  { _id: 4, StudName: 'Gukesh D', Grade: 'VII', Hobbies: 'Chess' },
  { _id: 2, StudName: 'Lando Norris', Grade: 'VII', Hobbies: 'Racing' },
  {
    _id: 1,
    StudName: 'Michelle Jacintha',
    Grade: 'VII',
    Hobbies: 'Internet Surfing'
  }
```

```
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find().skip(1).pretty().sort({StudName:1})
[
  { _id: 4, StudName: 'Gukesh D', Grade: 'VII', Hobbies: 'Chess' },
  { _id: 2, StudName: 'Lando Norris', Grade: 'VII', Hobbies: 'Racing' },
  {
    _id: 1,
    StudName: 'Michelle Jacintha',
    Grade: 'VII',
    Hobbies: 'Internet Surfing'
  }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find().pretty().skip(db.Student.count()-2);
[
  { _id: 3, StudName: 'Aryan David', Grade: 'VII', Hobbies: 'Skating' },
  { _id: 4, StudName: 'Gukesh D', Grade: 'VII', Hobbies: 'Chess' }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.find().pretty().skip(2).limit(3);
[
  { _id: 3, StudName: 'Aryan David', Grade: 'VII', Hobbies: 'Skating' },
  { _id: 4, StudName: 'Gukesh D', Grade: 'VII', Hobbies: 'Chess' }
]
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.update({_id:1}, {$set:{age:20}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-11p8k4-shard-0 [primary] studentDB> db.Student.update({_id:2}, {$set:{age:21}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

```
Atlas atlas-11p8k4-shard-0 [primary] customerDB> db.customers.aggregate([
... { $group:{
...         _id: "custid",
...         minAccBal:{$min: "$Balance"},
...         maxAccBal:{$max: "$Balance"}
...   }
... }
... ]);
[ { _id: 'custid', minAccBal: null, maxAccBal: null } ]
Atlas atlas-11p8k4-shard-0 [primary] customerDB> db.customers.aggregate([ { $group: { _id: "$custid", minAccBal: { $min: "$Balance"
}, maxAccBal: { $max: "$Balance" } } }] );
[
  { _id: 3, minAccBal: null, maxAccBal: null },
  { _id: 2, minAccBal: null, maxAccBal: null },
  { _id: 1, minAccBal: null, maxAccBal: null }
]
Atlas atlas-11p8k4-shard-0 [primary] customerDB>
```

Lab 2

Perform the following DB operations using Cassandra.
a)Create a keyspace by name Employee
b) Create a column family by name Employee-Info with attributes Emp_Id Primary Key, Emp_Name, Designation, Date_of_Joining, Salary,Dept_Name
c) Insert the values into the table in batch
d) Update Employee name and Department of Emp-Id 121
e) Sort the details of Employee records based on salary
f) Alter the schema of the table Employee_Info to add a column Projects which stores a set of Projects done by the corresponding Employee.
g) Update the altered table to add project names.
h) Create a TTL of 15 seconds to display the values of Employees.

```
varsha28@Ubuntu1:~/apache-cassandra-5.0.4$ cd bin
varsha28@Ubuntu1:~/apache-cassandra-5.0.4/bin$ ./cqlsh
Connection error: ('Unable to connect to any servers', {'127.0.0.1:9042': ConnectionRefusedError(111, "Tried connecting to [('127.0.
0.1', 9042)]. Last error: Connection refused")})
varsha28@Ubuntu1:~/apache-cassandra-5.0.4/bin$ cqlsh
Connected to Test Cluster at 127.0.0.1:9042.
[cqlsh 5.0.1 | Cassandra 5.0.4 | CQL spec 3.4.7 | Native protocol v5]
Use HELP for help.
cqlsh> CREATE KEYSPACE Employee WITH REPLICATION={'class':'SimpleStrategy','replication_factor':1};
cqlsh> DESCRIBE KEYSPACES;

system_virtual_schema  system_auth   system              employee
system_schema          system_views  system_distributed  system_traces

cqlsh> SELECT * FROM system_schema.keyspaces;

 keyspace_name      | durable_writes | replication
--------------------+----------------+-------------------------------------------------------------------------------------
        system_auth |           True | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '1'}
      system_schema |           True |                                 {'class': 'org.apache.cassandra.locator.LocalStrategy'}
 system_distributed |           True | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '3'}
             system |           True |                                 {'class': 'org.apache.cassandra.locator.LocalStrategy'}
      system_traces |           True | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '2'}
           employee |           True | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '1'}

(6 rows)
cqlsh> USE employee;
cqlsh:employee> CREATE TABLE empinfo(
            ... empid INT PRIMARY KEY,
            ... empname TEXT,
            ... designation TEXT,
            ... dateofjoining DATE,
            ... salary DOUBLE,
            ... deptname TEXT
            ... );
```

```
cqlsh:employee> BEGIN BATCH
            ... INSERT INTO empinfo(empid, empname, designation, dateofjoining, salary, deptname)
            ... VALUES(121, 'Aarohi Shirke', 'Developer', '2020-01-15', 55000, 'IT');
            ... INSERT INTO empinfo(empid, empname, designation, dateofjoining, salary, deptname)
            ... VALUES(122, 'Neil Sawant', 'Manager', '2018-03-10', 75000, 'HR');
            ... INSERT INTO empinfo(empid, empname, designation, dateofjoining, salary, deptname)
            ... VALUES(123, 'Sharayu Shivalkar', 'Analyst', '2021-07-22', 50000, 'Finance');
            ... APPLY BATCH;
cqlsh:employee> UPDATE empinfo
            ... SET empname='Aarohi Sawant', deptname='R&D'
            ... WHERE empid=121;
cqlsh:employee> SELECT * FROM empinfo;

 empid | dateofjoining | deptname | designation | empname           | salary
-------+---------------+----------+-------------+-------------------+--------
   123 |    2021-07-22 |  Finance |     Analyst | Sharayu Shivalkar |  50000
   122 |    2018-03-10 |       HR |     Manager |       Neil Sawant |  75000
   121 |    2020-01-15 |      R&D |   Developer |     Aarohi Sawant |  55000

(3 rows)
cqlsh:employee> ALTER TABLE empinfo ADD projects SET <TEXT>;
cqlsh:employee> SELECT * FROM empinfo;

 empid | dateofjoining | deptname | designation | empname           | projects | salary
-------+---------------+----------+-------------+-------------------+----------+--------
   123 |    2021-07-22 |  Finance |     Analyst | Sharayu Shivalkar |     null |  50000
   122 |    2018-03-10 |       HR |     Manager |       Neil Sawant |     null |  75000
   121 |    2020-01-15 |      R&D |   Developer |     Aarohi Sawant |     null |  55000

(3 rows)
cqlsh:employee> UPDATE empinfo
            ... SET projects={'ERP System', 'HR Portal'}
            ... WHERE empid=122;
```

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

 empid | dateofjoining | deptname | designation | empname           | projects                    | salary
-------+---------------+----------+-------------+-------------------+-----------------------------+--------
   123 |    2021-07-22 |  Finance |     Analyst | Sharayu Shivalkar |                        null |  50000
   122 |    2018-03-10 |       HR |     Manager |       Neil Sawant | {'ERP System', 'HR Portal'} |  75000
   121 |    2020-01-15 |      R&D |   Developer |     Aarohi Sawant |                        null |  55000

(3 rows)
cqlsh:employee> INSERT INTO empinfo(empid, empname, designation, dateofjoining, salary, deptname)
            ... VALUES(124, 'Ritvik Ranade', 'Tester', '2022-05-01', 45000, 'QA') USING TTL 15;
cqlsh:employee> SELECT * FROM empinfo;

 empid | dateofjoining | deptname | designation | empname           | projects                    | salary
-------+---------------+----------+-------------+-------------------+-----------------------------+--------
   123 |    2021-07-22 |  Finance |     Analyst | Sharayu Shivalkar |                        null |  50000
   122 |    2018-03-10 |       HR |     Manager |       Neil Sawant | {'ERP System', 'HR Portal'} |  75000
   121 |    2020-01-15 |      R&D |   Developer |     Aarohi Sawant |                        null |  55000
   124 |    2022-05-01 |       QA |      Tester |     Ritvik Ranade |                        null |  45000

(4 rows)
cqlsh:employee> SELECT * FROM empinfo;

 empid | dateofjoining | deptname | designation | empname           | projects                    | salary
-------+---------------+----------+-------------+-------------------+-----------------------------+--------
   123 |    2021-07-22 |  Finance |     Analyst | Sharayu Shivalkar |                        null |  50000
   122 |    2018-03-10 |       HR |     Manager |       Neil Sawant | {'ERP System', 'HR Portal'} |  75000
   121 |    2020-01-15 |      R&D |   Developer |     Aarohi Sawant |                        null |  55000

(3 rows)
```

Lab 3

Perform the following DB operations using Cassandra.

a)Create a keyspace by name Library

b) 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

c) Insert the values into the table in batch

d) Display the details of the table created and increase the value of the counter

e) Write a query to show that a student with id 112 has taken a book "BDA" 2 times.

f) Export the created column to a csv file

g) Import a given csv dataset from local file system into Cassandra column family

```
cqlsh:employee> CREATE KEYSPACE Library WITH REPLICATION={'class':'SimpleStrategy', 'replication_factor':1};
cqlsh:employee> USE Library;
cqlsh:library> SHOW KEYSPACES;
Improper SHOW command.
cqlsh:library> DESCRIBE KEYSPACES;

system_virtual_schema  system_auth   system   system_distributed  system_traces
system_schema          system_views  library  employee

cqlsh:library> CREATE TABLE libinfo(
           ... studid INT PRIMARY KEY;
SyntaxException: line 2:22 mismatched input ';' expecting ')' (...libinfo(studid INT PRIMARY KEY[;])
cqlsh:library> CREATE TABLE libinfo(
           ... studid INT PRIMARY KEY,
           ... studname TEXT,
           ... bookname TEXT,
           ... bookid TEXT,
           ... dateofissue DATE
           ... );
```

```
cqlsh:library> CREATE TABLE bookcounter(
           ... studid INT,
           ... bookname TEXT,
           ... counterval COUNTER,
           ... PRIMARY KEY((studid), bookname)
           ... );
```

```
cqlsh:library> BEGIN BATCH
           ... INSERT INTO libinfo(studid, studname, bookname, bookid, dateofissue)
           ... VALUES(112, 'Rahul', 'BDA', 'B101', '2024-04-01');
           ... INSERT INTO libinfo(studid, studname, bookname, bookid, dateofissue)
           ... VALUES(113, 'Neha', 'ML', 'B102', '2024-04-02');
           ... INSERT INTO libinfo(studid, studname, bookname, bookid, dateofissue)
           ... VALUES(114, 'Aarohi', 'CC', 'B103', '2024-04-03');
           ... INSERT INTO libinfo(studid, studname, bookname, bookid, dateofissue)
           ... VALUES(115, 'Neil', 'RML', 'B104', '2024-04-04');
           ... APPLY BATCH;
cqlsh:library> UPDATE bookcounter
           ... SET counterval = counterval + 1
           ... WHERE studid=112 AND bookname='BDA';
cqlsh:library> UPDATE bookcounter
           ... SET counterval = counterval + 1
           ... WHERE studid=112 AND bookname='BDA';
cqlsh:library> SELECT * FROM bookcounter;

 studid | bookname | counterval
--------+----------+------------
    112 |      BDA |          2

(1 rows)
```

IMPORT:

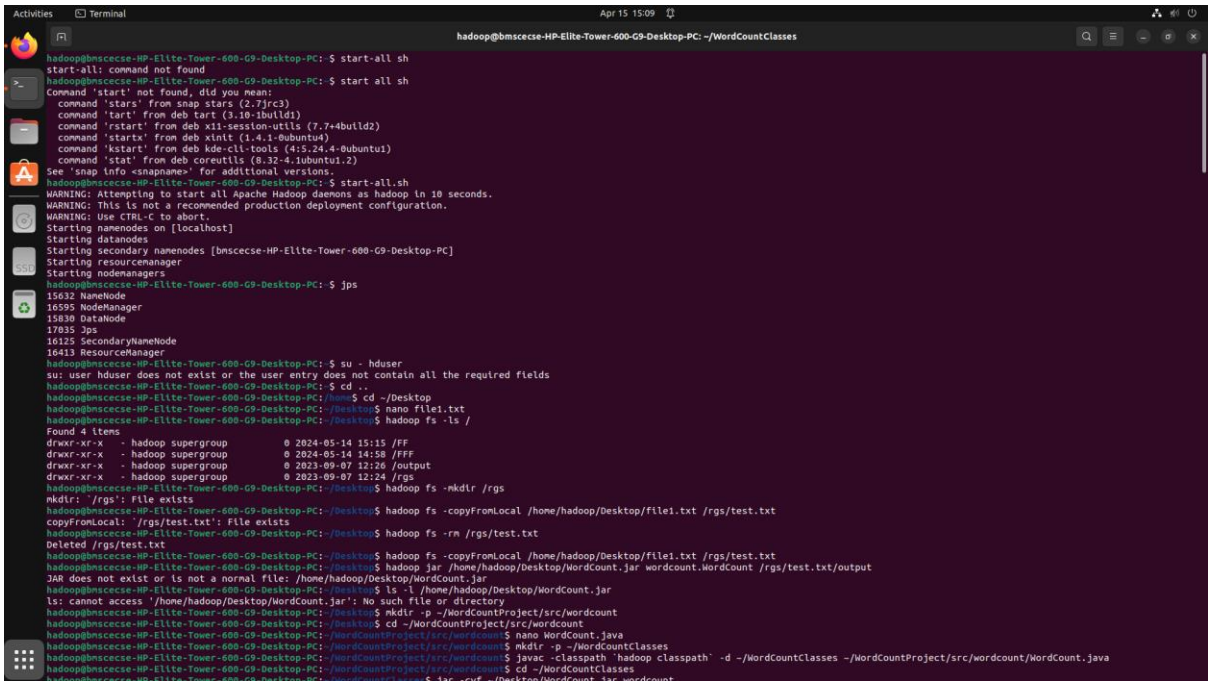
COPY libinfo TO 'libinfo.csv' WITH HEADER=TRUE;

EXPORT:

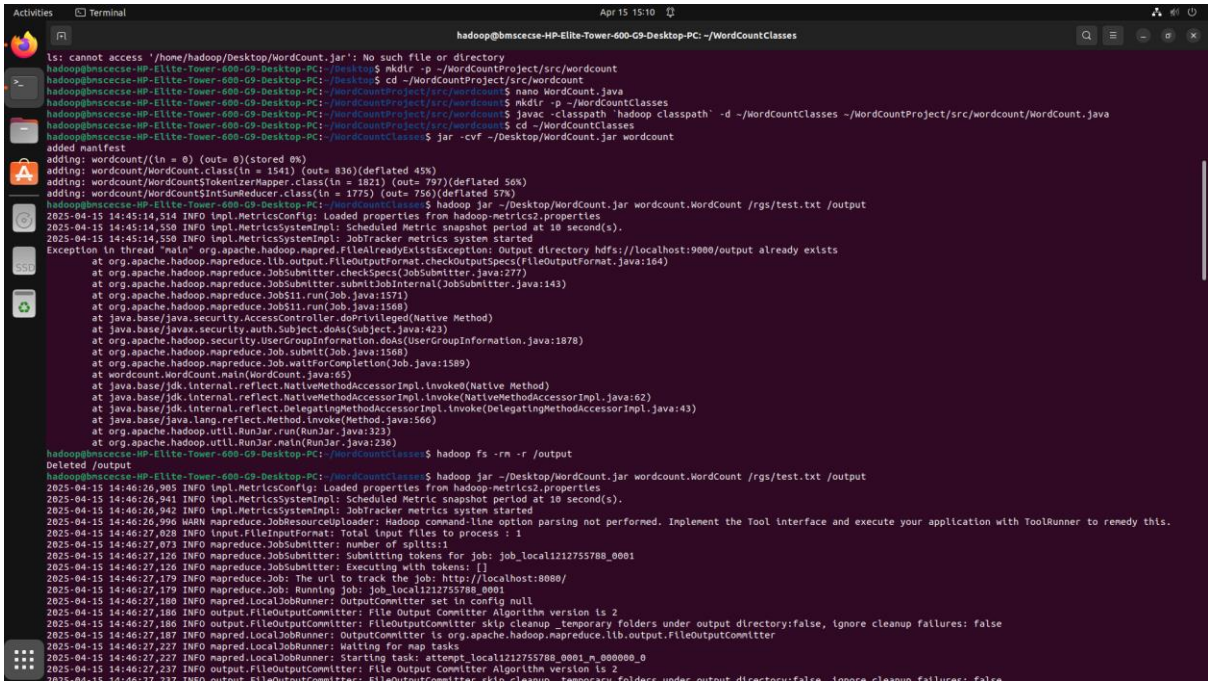
COPY libinfo(studid, studname, bookname, bookid, dateofissue) FROM 'libinfo.csv' WITH HEADER=TRUE;

Lab 4

Execution of HDFS Commands for interaction with Hadoop Environment. (Minimum 10 commands to be executed)



```
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ start-all sh
start-all: command not found
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ start all sh
Command 'start' not found, did you mean:
  command 'stars' from snap stars (2.7jrc3)
  command 'tart' from deb tart (3.10-1build1)
  command 'rstart' from deb x11-session-utils (7.7+4build2)
  command 'startx' from deb xinit (1.4.1-0ubuntu4)
  command 'kstart' from deb kde-cli-tools (4:5.24.4-0ubuntu1)
  command 'stat' from deb coreutils (8.32-4.1ubuntu1.2)
See 'snap info <snapname>' for additional versions.
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as hadoop in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [bmscecse-HP-Elite-Tower-600-G9-Desktop-PC]
Starting resourcemanager
Starting nodemanagers
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ jps
15632 NameNode
16595 NodeManager
15830 DataNode
17035 Jps
16125 SecondaryNameNode
16413 ResourceManager
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ su - hduser
su: user hduser does not exist or the user entry does not contain all the required fields
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ cd ..
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:/home$ cd ~/Desktop
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Desktop$ nano file1.txt
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Desktop$ hadoop fs -ls /
Found 4 items
drwxr-xr-x   - hadoop supergroup          0 2024-05-14 15:15 /FF
drwxr-xr-x   - hadoop supergroup          0 2024-05-14 14:58 /FFF
drwxr-xr-x   - hadoop supergroup          0 2023-09-07 12:26 /output
drwxr-xr-x   - hadoop supergroup          0 2023-09-07 12:24 /rgs
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Desktop$ hadoop fs -mkdir /rgs
mkdir: `/rgs': File exists
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Desktop$ hadoop fs -copyFromLocal /home/hadoop/Desktop/file1.txt /rgs/test.txt
copyFromLocal: `/rgs/test.txt': File exists
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Desktop$ hadoop fs -rm /rgs/test.txt
Deleted /rgs/test.txt
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Desktop$ hadoop fs -copyFromLocal /home/hadoop/Desktop/file1.txt /rgs/test.txt
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Desktop$ hadoop jar /home/hadoop/Desktop/WordCount.jar wordcount.WordCount /rgs/test.txt/output
JAR does not exist or is not a normal file: /home/hadoop/Desktop/WordCount.jar
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Desktop$ ls -l /home/hadoop/Desktop/WordCount.jar
ls: cannot access '/home/hadoop/Desktop/WordCount.jar': No such file or directory
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Desktop$ mkdir -p ~/WordCountProject/src/wordcount
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Desktop$ cd ~/WordCountProject/src/wordcount
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountProject/src/wordcount$ nano WordCount.java
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountProject/src/wordcount$ mkdir -p ~/WordCountClasses
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountProject/src/wordcount$ javac -classpath `hadoop classpath` -d ~/WordCountClasses ~/WordCountProject/src/wordcount/WordCount.java
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountProject/src/wordcount$ cd ~/WordCountClasses
```



```
ls: cannot access '/home/hadoop/Desktop/WordCount.jar': No such file or directory
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Desktop$ mkdir -p ~/WordCountProject/src/wordcount
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/Desktop$ cd ~/WordCountProject/src/wordcount
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountProject/src/wordcount$ nano WordCount.java
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountProject/src/wordcount$ mkdir -p ~/WordCountClasses
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountProject/src/wordcount$ javac -classpath `hadoop classpath` -d ~/WordCountClasses ~/WordCountProject/src/wordcount/WordCount.java
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountProject/src/wordcount$ cd ~/WordCountClasses
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountClasses$ jar -cvf ~/Desktop/WordCount.jar wordcount
added manifest
adding: wordcount/(in = 0) (out= 0)(stored 0%)
adding: wordcount/WordCount.class(in = 1541) (out= 836)(deflated 45%)
adding: wordcount/WordCount$TokenizerMapper.class(in = 1821) (out= 797)(deflated 56%)
adding: wordcount/WordCount$IntSumReducer.class(in = 1775) (out= 756)(deflated 57%)
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountClasses$ hadoop jar ~/Desktop/WordCount.jar wordcount.WordCount /rgs/test.txt /output
2025-04-15 14:45:14,514 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2025-04-15 14:45:14,550 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2025-04-15 14:45:14,550 INFO impl.MetricsSystemImpl: JobTracker metrics system started
Exception in thread "main" org.apache.hadoop.mapred.FileAlreadyExistsException: Output directory hdfs://localhost:9000/output already exists
        at org.apache.hadoop.mapreduce.lib.output.FileOutputFormat.checkOutputSpecs(FileOutputFormat.java:164)
        at org.apache.hadoop.mapreduce.JobSubmitter.checkSpecs(JobSubmitter.java:277)
        at org.apache.hadoop.mapreduce.JobSubmitter.submitJobInternal(JobSubmitter.java:143)
        at org.apache.hadoop.mapreduce.Job$11.run(Job.java:1571)
        at org.apache.hadoop.mapreduce.Job$11.run(Job.java:1568)
        at java.base/java.security.AccessController.doPrivileged(Native Method)
        at java.base/javax.security.auth.Subject.doAs(Subject.java:423)
        at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1878)
        at org.apache.hadoop.mapreduce.Job.submit(Job.java:1568)
        at org.apache.hadoop.mapreduce.Job.waitForCompletion(Job.java:1589)
        at wordcount.WordCount.main(WordCount.java:65)
        at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
        at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.base/java.lang.reflect.Method.invoke(Method.java:566)
        at org.apache.hadoop.util.RunJar.run(RunJar.java:323)
        at org.apache.hadoop.util.RunJar.main(RunJar.java:236)
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountClasses$ hadoop fs -rm -r /output
Deleted /output
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountClasses$ hadoop jar ~/Desktop/WordCount.jar wordcount.WordCount /rgs/test.txt /output
2025-04-15 14:46:26,905 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2025-04-15 14:46:26,941 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2025-04-15 14:46:26,942 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2025-04-15 14:46:26,996 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
2025-04-15 14:46:27,028 INFO input.FileInputFormat: Total input files to process : 1
2025-04-15 14:46:27,073 INFO mapreduce.JobSubmitter: number of splits:1
2025-04-15 14:46:27,126 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local1212755788_0001
2025-04-15 14:46:27,126 INFO mapreduce.JobSubmitter: Executing with tokens: []
2025-04-15 14:46:27,179 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
2025-04-15 14:46:27,179 INFO mapreduce.Job: Running job: job_local1212755788_0001
2025-04-15 14:46:27,180 INFO mapred.LocalJobRunner: OutputCommitter set in config null
2025-04-15 14:46:27,186 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2025-04-15 14:46:27,186 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
2025-04-15 14:46:27,187 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
2025-04-15 14:46:27,227 INFO mapred.LocalJobRunner: Waiting for map tasks
2025-04-15 14:46:27,227 INFO mapred.LocalJobRunner: Starting task: attempt_local1212755788_0001_m_000000_0
2025-04-15 14:46:27,237 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
```

```
Activities   Terminal                                   Apr 15 15:10
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC: ~/WordCountClasses
2025-04-15 14:46:27,331 INFO mapred.Task: Task 'attempt_local1212755788_0001_m_000000_0' done.
2025-04-15 14:46:27,333 INFO mapred.Task: Final Counters for attempt_local1212755788_0001_m_000000_0: Counters: 24
        File System Counters
                FILE: Number of bytes read=3484
                FILE: Number of bytes written=643788
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=89
                HDFS: Number of bytes written=0
                HDFS: Number of read operations=5
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=1
                HDFS: Number of bytes read erasure-coded=0
        Map-Reduce Framework
                Map input records=5
                Map output records=20
                Map output bytes=169
                Map output materialized bytes=115
                Input split bytes=99
                Combine input records=20
                Combine output records=10
                Spilled Records=10
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=0
                Total committed heap usage (bytes)=526385152
        File Input Format Counters
                Bytes Read=89
2025-04-15 14:46:27,333 INFO mapred.LocalJobRunner: Finishing task: attempt_local1212755788_0001_m_000000_0
2025-04-15 14:46:27,334 INFO mapred.LocalJobRunner: map task executor complete.
2025-04-15 14:46:27,336 INFO mapred.LocalJobRunner: Waiting for reduce tasks
2025-04-15 14:46:27,336 INFO mapred.LocalJobRunner: Starting task: attempt_local1212755788_0001_r_000000_0
2025-04-15 14:46:27,339 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2025-04-15 14:46:27,339 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
2025-04-15 14:46:27,339 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
2025-04-15 14:46:27,340 INFO mapred.ReduceTask: Using ShuffleConsumerPlugin: org.apache.hadoop.mapreduce.task.reduce.Shuffle@4bc31898
2025-04-15 14:46:27,341 WARN impl.MetricsSystemImpl: JobTracker metrics system already initialized!
2025-04-15 14:46:27,348 INFO reduce.MergeManagerImpl: MergerManager: memoryLimit=5827985408, maxSingleShuffleLimit=1456996352, mergeThreshold=3846470400, ioSortFactor=10, memToMemMergeOutputsThreshold=10
2025-04-15 14:46:27,349 INFO reduce.EventFetcher: attempt_local1212755788_0001_r_000000_0 Thread started: EventFetcher for fetching Map Completion Events
2025-04-15 14:46:27,361 INFO reduce.LocalFetcher: localfetcher#1 about to shuffle output of map attempt_local1212755788_0001_m_000000_0 decomp: 111 len: 115 to MEMORY
2025-04-15 14:46:27,362 INFO reduce.InMemoryMapOutput: Read 111 bytes from map-output for attempt_local1212755788_0001_m_000000_0
2025-04-15 14:46:27,363 INFO reduce.MergeManagerImpl: closeInMemoryFile -> map-output of size: 111, inMemoryMapOutputs.size() -> 1, commitMemory -> 0, usedMemory ->111
2025-04-15 14:46:27,364 INFO reduce.EventFetcher: EventFetcher is interrupted.. Returning
2025-04-15 14:46:27,364 INFO mapred.LocalJobRunner: 1 / 1 copied.
2025-04-15 14:46:27,364 INFO reduce.MergeManagerImpl: finalMerge called with 1 in-memory map-outputs and 0 on-disk map-outputs
2025-04-15 14:46:27,366 INFO mapred.Merger: Merging 1 sorted segments
2025-04-15 14:46:27,366 INFO mapred.Merger: Down to the last merge-pass, with 1 segments left of total size: 105 bytes
2025-04-15 14:46:27,367 INFO reduce.MergeManagerImpl: Merged 1 segments, 111 bytes to disk to satisfy reduce memory limit
2025-04-15 14:46:27,367 INFO reduce.MergeManagerImpl: Merging 1 files, 115 bytes from disk
2025-04-15 14:46:27,367 INFO reduce.MergeManagerImpl: Merging 0 segments, 0 bytes from memory into reduce
2025-04-15 14:46:27,367 INFO mapred.Merger: Merging 1 sorted segments
2025-04-15 14:46:27,367 INFO mapred.Merger: Down to the last merge-pass, with 1 segments left of total size: 105 bytes
2025-04-15 14:46:27,368 INFO mapred.LocalJobRunner: 1 / 1 copied.
2025-04-15 14:46:27,393 INFO Configuration.deprecation: mapred.skip.on is deprecated. Instead, use mapreduce.job.skiprecords
```

```
Activities   Terminal                                   Apr 15 15:10
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC: ~/WordCountClasses
Deleted /output
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountClasses$ hadoop jar ~/Desktop/WordCount.jar wordcount.WordCount /rgs/test.txt /output
2025-04-15 14:46:26,905 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2025-04-15 14:46:26,941 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2025-04-15 14:46:26,942 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2025-04-15 14:46:26,996 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
2025-04-15 14:46:27,028 INFO input.FileInputFormat: Total input files to process : 1
2025-04-15 14:46:27,073 INFO mapreduce.JobSubmitter: number of splits:1
2025-04-15 14:46:27,126 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local1212755788_0001
2025-04-15 14:46:27,126 INFO mapreduce.JobSubmitter: Executing with tokens: []
2025-04-15 14:46:27,179 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
2025-04-15 14:46:27,179 INFO mapreduce.Job: Running job: job_local1212755788_0001
2025-04-15 14:46:27,180 INFO mapred.LocalJobRunner: OutputCommitter set in config null
2025-04-15 14:46:27,186 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2025-04-15 14:46:27,186 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
2025-04-15 14:46:27,187 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
2025-04-15 14:46:27,227 INFO mapred.LocalJobRunner: Waiting for map tasks
2025-04-15 14:46:27,227 INFO mapred.LocalJobRunner: Starting task: attempt_local1212755788_0001_m_000000_0
2025-04-15 14:46:27,237 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2025-04-15 14:46:27,237 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
2025-04-15 14:46:27,244 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
2025-04-15 14:46:27,246 INFO mapred.MapTask: Processing split: hdfs://localhost:9000/rgs/test.txt:0+89
2025-04-15 14:46:27,277 INFO mapred.MapTask: (EQUATOR) 0 kvi 26214396(104857584)
2025-04-15 14:46:27,277 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100
2025-04-15 14:46:27,277 INFO mapred.MapTask: soft limit at 83886080
2025-04-15 14:46:27,277 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600
2025-04-15 14:46:27,277 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
2025-04-15 14:46:27,279 INFO mapred.MapTask: Map output collector class = org.apache.hadoop.mapred.MapTask$MapOutputBuffer
2025-04-15 14:46:27,319 INFO mapred.LocalJobRunner:
2025-04-15 14:46:27,320 INFO mapred.MapTask: Starting flush of map output
2025-04-15 14:46:27,320 INFO mapred.MapTask: Spilling map output
2025-04-15 14:46:27,320 INFO mapred.MapTask: bufstart = 0; bufend = 169; bufvoid = 104857600
2025-04-15 14:46:27,320 INFO mapred.MapTask: kvstart = 26214396(104857584); kvend = 26214320(104857280); length = 77/6553600
2025-04-15 14:46:27,325 INFO mapred.MapTask: Finished spill 0
2025-04-15 14:46:27,329 INFO mapred.Task: Task:attempt_local1212755788_0001_m_000000_0 is done. And is in the process of committing
2025-04-15 14:46:27,331 INFO mapred.LocalJobRunner: map
2025-04-15 14:46:27,331 INFO mapred.Task: Task 'attempt_local1212755788_0001_m_000000_0' done.
2025-04-15 14:46:27,333 INFO mapred.Task: Final Counters for attempt_local1212755788_0001_m_000000_0: Counters: 24
        File System Counters
                FILE: Number of bytes read=3484
                FILE: Number of bytes written=643788
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=89
                HDFS: Number of bytes written=0
                HDFS: Number of read operations=5
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=1
                HDFS: Number of bytes read erasure-coded=0
        Map-Reduce Framework
                Map input records=5
                Map output records=20
                Map output bytes=169
                Map output materialized bytes=115
                Input split bytes=99
```

```
Activities   Terminal                                   Apr 15 15:11
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC: ~/WordCountClasses
2025-04-15 14:46:27,447 INFO mapred.LocalJobRunner: Finishing task: attempt_local1212755788_0001_r_000000_0
2025-04-15 14:46:27,447 INFO mapred.LocalJobRunner: reduce task executor complete.
2025-04-15 14:46:28,182 INFO mapreduce.Job: Job job_local1212755788_0001 running in uber mode : false
2025-04-15 14:46:28,182 INFO mapreduce.Job:  map 100% reduce 100%
2025-04-15 14:46:28,183 INFO mapreduce.Job: Job job_local1212755788_0001 completed successfully
2025-04-15 14:46:28,186 INFO mapreduce.Job: Counters: 36
        File System Counters
                FILE: Number of bytes read=7230
                FILE: Number of bytes written=1287691
                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=15
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=4
                HDFS: Number of bytes read erasure-coded=0
        Map-Reduce Framework
                Map input records=5
                Map output records=20
                Map output bytes=169
                Map output materialized bytes=115
                Input split bytes=99
                Combine input records=20
                Combine output records=10
                Reduce input groups=10
                Reduce shuffle bytes=115
                Reduce input records=10
                Reduce output records=10
                Spilled Records=20
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=0
                Total committed heap usage (bytes)=1052770304
        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
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountClasses$ hadoop fs -cat /output/part-00000
cat: `/output/part-00000': No such file or directory
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountClasses$ hadoop fs -ls /output
Found 2 items
-rw-r--r--   1 hadoop supergroup          0 2025-04-15 14:46 /output/_SUCCESS
-rw-r--r--   1 hadoop supergroup         69 2025-04-15 14:46 /output/part-r-00000
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountClasses$ hadoop fs -cat /output/part-r-00000
are     1
brother 1
```

```
Activities   Terminal                    Apr 15 15:11
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC: ~/WordCountClasses
                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=15
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=4
                HDFS: Number of bytes read erasure-coded=0
        Map-Reduce Framework
                Map input records=5
                Map output records=20
                Map output bytes=169
                Map output materialized bytes=115
                Input split bytes=99
                Combine input records=20
                Combine output records=10
                Reduce input groups=10
                Reduce shuffle bytes=115
                Reduce input records=10
                Reduce output records=10
                Spilled Records=20
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=0
                Total committed heap usage (bytes)=1052770304
        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
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountClasses$ hadoop fs -cat /output/part-00000
cat: `/output/part-00000': No such file or directory
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountClasses$ hadoop fs -ls /output
Found 2 items
-rw-r--r--   1 hadoop supergroup          0 2025-04-15 14:46 /output/_SUCCESS
-rw-r--r--   1 hadoop supergroup         69 2025-04-15 14:46 /output/part-r-00000
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountClasses$ hadoop fs -cat /output/part-r-00000
are     1
brother 1
family  1
hi      1
how     5
is      4
job     1
sister  1
you     1
your    4
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/WordCountClasses$
```

```
Activities   Terminal                    Apr 15 15:11
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC: ~/WordCountClasses
2025-04-15 14:46:27,367 INFO reduce.MergeManagerImpl: Merged 1 segments, 111 bytes to disk to satisfy reduce memory limit
2025-04-15 14:46:27,367 INFO reduce.MergeManagerImpl: Merging 1 files, 115 bytes from disk
2025-04-15 14:46:27,367 INFO reduce.MergeManagerImpl: Merging 0 segments, 0 bytes from memory into reduce
2025-04-15 14:46:27,367 INFO mapred.Merger: Merging 1 sorted segments
2025-04-15 14:46:27,367 INFO mapred.Merger: Down to the last merge-pass, with 1 segments left of total size: 105 bytes
2025-04-15 14:46:27,368 INFO mapred.LocalJobRunner: 1 / 1 copied.
2025-04-15 14:46:27,393 INFO Configuration.deprecation: mapred.skip.on is deprecated. Instead, use mapreduce.job.skiprecords
2025-04-15 14:46:27,430 INFO mapred.Task: Task:attempt_local1212755788_0001_r_000000_0 is done. And is in the process of committing
2025-04-15 14:46:27,432 INFO mapred.LocalJobRunner: 1 / 1 copied.
2025-04-15 14:46:27,432 INFO mapred.Task: Task attempt_local1212755788_0001_r_000000_0 is allowed to commit now
2025-04-15 14:46:27,446 INFO output.FileOutputCommitter: Saved output of task 'attempt_local1212755788_0001_r_000000_0' to hdfs://localhost:9000/output
2025-04-15 14:46:27,447 INFO mapred.LocalJobRunner: reduce > reduce
2025-04-15 14:46:27,447 INFO mapred.Task: Task 'attempt_local1212755788_0001_r_000000_0' done.
2025-04-15 14:46:27,447 INFO mapred.Task: Final Counters for attempt_local1212755788_0001_r_000000_0: Counters: 30
        File System Counters
                FILE: Number of bytes read=3746
                FILE: Number of bytes written=643903
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=89
                HDFS: Number of bytes written=69
                HDFS: Number of read operations=10
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=3
                HDFS: Number of bytes read erasure-coded=0
        Map-Reduce Framework
                Combine input records=0
                Combine output records=0
                Reduce input groups=10
                Reduce shuffle bytes=115
                Reduce input records=10
                Reduce output records=10
                Spilled Records=10
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=0
                Total committed heap usage (bytes)=526385152
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Output Format Counters
                Bytes Written=69
2025-04-15 14:46:27,447 INFO mapred.LocalJobRunner: Finishing task: attempt_local1212755788_0001_r_000000_0
2025-04-15 14:46:27,447 INFO mapred.LocalJobRunner: reduce task executor complete.
2025-04-15 14:46:28,182 INFO mapreduce.Job: Job job_local1212755788_0001 running in uber mode : false
2025-04-15 14:46:28,182 INFO mapreduce.Job:  map 100% reduce 100%
2025-04-15 14:46:28,183 INFO mapreduce.Job: Job job_local1212755788_0001 completed successfully
2025-04-15 14:46:28,186 INFO mapreduce.Job: Counters: 36
        File System Counters
                FILE: Number of bytes read=7330
```

Lab 5

Implement Wordcount program on Hadoop framework

Driver code

```
// 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.FileInputFor
mat; import
org.apache.hadoop.mapred.FileOutputFo
rmat; 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);
conf.setJobName("WordCount");

       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("Job Exit Code: " + exitCode);
    }
}
```

Mapper Code

```
// 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.MapReduce
Base; 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 reporter)            throws IOException {
        String line = value.toString();

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

Reducer Code

```
// Importing libraries
import
java.io.IOException;
import java.util.Iterator;
```

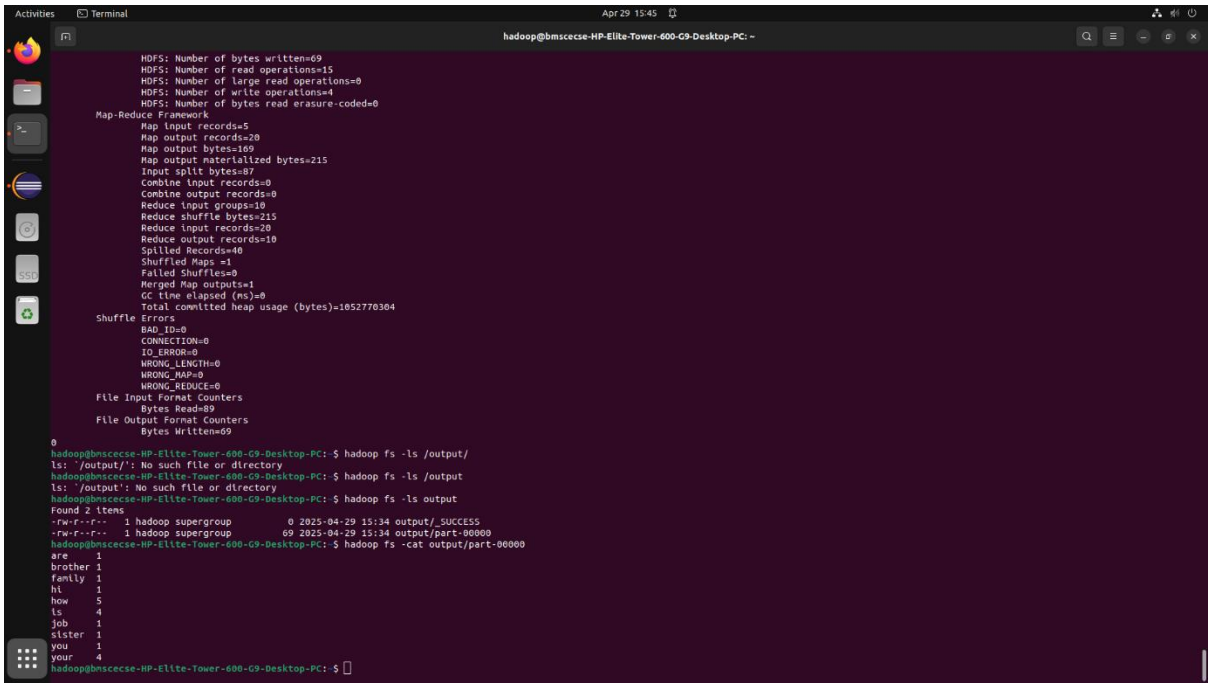
```
import
org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import
org.apache.hadoop.mapred.MapReduce
Base; import
org.apache.hadoop.mapred.OutputColle
ctor; import
org.apache.hadoop.mapred.Reducer;
import org.apache.hadoop.mapred.Reporter;

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

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

        // Counting the frequency of each
word         while (values.hasNext()) {
            count += values.next().get();
        }

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



Lab 6

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. b) find the mean max temperature for every month.

Driver Code

```
package temp;

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.lib.input.FileInputForm
at; import
org.apache.hadoop.mapreduce.lib.output.FileOutputFo
rmat;

public class AverageDriver {

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

      if (args.length != 2) {
         System.err.println("Please enter both input and output parameters.");
         System.exit(-1);
      }

      // Creating a configuration and job instance
      Configuration conf = new Configuration();
      Job job = Job.getInstance(conf, "Average Calculation");

      job.setJarByClass(AverageDriver.class);

      // Input and output paths
      FileInputFormat.addInputPath(job, new Path(args[0]));
      FileOutputFormat.setOutputPath(job, new Path(args[1]));

      // Setting mapper and reducer classes
job.setMapperClass(AverageMapper.class);
      job.setReducerClass(AverageReducer.class);
```

```
        // Output key and value types
job.setOutputKeyClass(Text.class);
        job.setOutputValueClass(IntWritable.class);

        // Submitting the job and waiting for it to complete
        System.exit(job.waitForCompletion(true) ? 0 : 1);
    }
}
```

Mapper Code

```
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;

    @Override
    public void map(LongWritable key, Text value,
Context context)            throws IOException,
InterruptedException {

        String line = value.toString();

        // Extract year from fixed
position              String year =
line.substring(15, 19);           int
temperature;

        // Determine if there's a '+' sign        if
(line.charAt(87) == '+') {            temperature =
Integer.parseInt(line.substring(88, 92));
        } else {
            temperature = Integer.parseInt(line.substring(87, 92));
        }

        // Quality check character
        String quality = line.substring(92, 93);

        // Only emit if data is valid
```

```
        if (temperature != MISSING && quality.matches("[01459]")) {
            context.write(new Text(year), new IntWritable(temperature));
        }
    }
}
```

Reducer Code

```
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> {

    @Override
    public void reduce(Text key, Iterable<IntWritable> values,
                    Context context) throws IOException, InterruptedException {

        int sumTemp = 0;
        int count = 0;

        for (IntWritable value : values) {
sumTemp += value.get();
            count++;
        }

        if (count > 0) {            int average =
sumTemp / count;
context.write(key, new
IntWritable(average));
        }
    }
}
```

```
hadoop@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC: $ hadoop jar /home/hadoop/Desktop/AverageTemperature.jar AverageDriver /weather/test.txt /weather/output
2025-05-06 14:59:23,239 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2025-05-06 14:59:23,279 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2025-05-06 14:59:23,279 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2025-05-06 14:59:23,340 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
2025-05-06 14:59:23,393 INFO input.FileInputFormat: Total input files to process : 1
2025-05-06 14:59:23,422 INFO mapreduce.JobSubmitter: number of splits:1
2025-05-06 14:59:23,487 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local91822813_0001
2025-05-06 14:59:23,488 INFO mapreduce.JobSubmitter: Executing with tokens: []
2025-05-06 14:59:23,560 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
2025-05-06 14:59:23,560 INFO mapreduce.Job: Running job: job_local91822813_0001
2025-05-06 14:59:23,561 INFO mapred.LocalJobRunner: OutputCommitter set in config null
2025-05-06 14:59:23,564 INFO output.PathOutputCommitterFactory: No output committer factory defined, defaulting to FileOutputCommitterFactory
2025-05-06 14:59:23,565 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2025-05-06 14:59:23,565 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
2025-05-06 14:59:23,565 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
2025-05-06 14:59:23,602 INFO mapred.LocalJobRunner: Waiting for map tasks
2025-05-06 14:59:23,603 INFO mapred.LocalJobRunner: Starting task: attempt_local91822813_0001_m_000000_0
2025-05-06 14:59:23,615 INFO output.PathOutputCommitterFactory: No output committer factory defined, defaulting to FileOutputCommitterFactory
2025-05-06 14:59:23,615 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2025-05-06 14:59:23,615 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
2025-05-06 14:59:23,622 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
2025-05-06 14:59:23,624 INFO mapred.MapTask: Processing split: hdfs://localhost:9000/weather/test.txt:0+888190
2025-05-06 14:59:23,658 INFO mapred.MapTask: (EQUATOR) 0 kvi 26214396(104857584)
2025-05-06 14:59:23,658 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100
2025-05-06 14:59:23,658 INFO mapred.MapTask: soft limit at 83886080
2025-05-06 14:59:23,658 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600
2025-05-06 14:59:23,658 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
2025-05-06 14:59:23,660 INFO mapred.MapTask: Map output collector class = org.apache.hadoop.mapred.MapTask$MapOutputBuffer
```

Lab 7

For a given Text file, Create a Map Reduce program to sort the content in an alphabetic order listing only top 10 maximum occurrences of words.

Driver Code (TopNDriver.java)

```
package samples.topn;

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.lib.input.FileInputFor
mat; import
org.apache.hadoop.mapreduce.lib.output.FileOutputF
ormat;

public class TopNDriver {

   public static void main(String[] args) throws
Exception {        if (args.length != 3) {
          System.err.println("Usage: TopNDriver <in> <temp-
out> <final-out>");            System.exit(2);
       }

       Configuration conf = new Configuration();

       // === Job 1: Word Count ===
       Job wcJob = Job.getInstance(conf, "word count");
wcJob.setJarByClass(TopNDriver.class);
wcJob.setMapperClass(WordCountMapper.class);
wcJob.setCombinerClass(WordCountReducer.class);
wcJob.setReducerClass(WordCountReducer.class);
wcJob.setOutputKeyClass(Text.class);
       wcJob.setOutputValueClass(IntWritable.class);

       FileInputFormat.addInputPath(wcJob, new Path(args[0]));
       Path tempDir = new Path(args[1]);
       FileOutputFormat.setOutputPath(wcJob, tempDir);

       if (!wcJob.waitForCompletion(true)) {
          System.exit(1);
       }

       // === Job 2: Top N ===
```

```
        Job topJob = Job.getInstance(conf, "top 10 words");
topJob.setJarByClass(TopNDriver.class);
topJob.setMapperClass(TopNMapper.class);
topJob.setReducerClass(TopNReducer.class);
topJob.setMapOutputKeyClass(IntWritable.class);
topJob.setMapOutputValueClass(Text.class);
topJob.setOutputKeyClass(Text.class);
        topJob.setOutputValueClass(IntWritable.class);

        FileInputFormat.addInputPath(topJob, tempDir);
        FileOutputFormat.setOutputPath(topJob, new
Path(args[2]));

        System.exit(topJob.waitForCompletion(true) ? 0 : 1);
    }
}
```

Mapper Code (TopNMapper.java)

```
package samples.topn;

import java.io.IOException;

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, IntWritable, Text> {

    private IntWritable count = new
IntWritable();     private Text word = new
Text();

    @Override
    protected void map(Object key, Text value, Context
context)        throws IOException, InterruptedException
{

        // input line: word \t count
        String[] parts =
value.toString().split("\\t");        if
(parts.length == 2) {
word.set(parts[0]);
            count.set(Integer.parseInt(parts[1]));
// emit count → word, so Hadoop sorts by
count
            context.write(count, word);
```

```
        }
    } }
```

Reducer Code (TopNReducer.java)

```
package samples.topn;

import java.io.IOException; import
java.util.ArrayList; import
java.util.Collections; import
java.util.List; import java.util.Map;
import java.util.TreeMap;

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

public class TopNReducer
   extends Reducer<IntWritable, Text, Text, IntWritable> {

   // TreeMap with descending order of keys (counts)     private
TreeMap<Integer, List<String>> countMap =
      new TreeMap<>(Collections.reverseOrder());

   @Override
   protected void reduce(IntWritable key, Iterable<Text> values, Context context)
throws IOException, InterruptedException {

      int cnt = key.get();
      List<String> words = countMap.getOrDefault(cnt, new ArrayList<>());         for
(Text w : values) {
         words.add(w.toString());
      }
      countMap.put(cnt, words);
   }

   @Override
   protected void cleanup(Context context)
      throws IOException, InterruptedException {

      // collect top 10 word→count pairs
      List<WordCount> topList = new ArrayList<>();         int seen = 0;         for
(Map.Entry<Integer, List<String>> entry : countMap.entrySet()) {             int cnt =
entry.getKey();             for (String w : entry.getValue()) {
            topList.add(new WordCount(w, cnt));
            seen++;                if (seen ==
10) break;
         }
         if (seen == 10) break;
      }

      // sort these 10 entries alphabetically by word
      Collections.sort(topList, (a, b) -> a.word.compareTo(b.word));

      // emit final top 10 in alphabetical order         for (WordCount wc : topList)
{            context.write(new Text(wc.word), new IntWritable(wc.count));
      }
   }

   // helper class     private static class
WordCount {
      String word;
      int count;
      WordCount(String w, int c) { word = w; count = c; }
   }
}
```

Mapper Code (WordCountMapper.java)

```
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 WordCountMapper     extends
Mapper<Object, Text, Text, IntWritable> {

   private final static IntWritable ONE = new
IntWritable(1);     private Text word = new
Text();     // characters to normalize into spaces
   private String tokens = "[_|$#<>\\^=\\[\\]\\*/\\\\,;,.\\-:()?!\"']";

   @Override     protected void map(Object key,
Text value, Context context)        throws
IOException, InterruptedException {

       // clean & tokenize
       String clean = value.toString()
                       .toLowerCase()
                       .replaceAll(tokens, " ");
       StringTokenizer itr = new
StringTokenizer(clean);        while
(itr.hasMoreTokens()) {
word.set(itr.nextToken().trim());
           context.write(word, ONE);
       }
   }
}
```

Reducer Code (WordCountReducer.java)

```
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 WordCountReducer     extends
Reducer<Text, IntWritable, Text, IntWritable> {

   @Override     protected void reduce(Text key,
Iterable<IntWritable> values, Context context)         throws
IOException, InterruptedException {

       int sum = 0;         for
(IntWritable val : values) {
           sum += val.get();
       }
       context.write(key, new IntWritable(sum));
   }
}
```

```
C:\hadoop-3.3.0\sbin>jps
11072 DataNode
20528 Jps
5620 ResourceManager
15532 NodeManager
6140 NameNode

C:\hadoop-3.3.0\sbin>hdfs dfs -mkdir /input_dir

C:\hadoop-3.3.0\sbin>hdfs dfs -ls /
Found 1 items
drwxr-xr-x   - Anusree supergroup          0 2021-05-08 19:46 /input_dir

C:\hadoop-3.3.0\sbin>hdfs dfs -copyFromLocal C:\input.txt /input_dir

C:\hadoop-3.3.0\sbin>hdfs dfs -ls /input_dir
Found 1 items
-rw-r--r--   1 Anusree supergroup         36 2021-05-08 19:48 /input_dir/input.txt

C:\hadoop-3.3.0\sbin>hdfs dfs -cat /input_dir/input.txt
hello
world
hello
hadoop
bye
```


```
C:\hadoop-3.3.0\sbin>jps
11072 DataNode
20528 Jps
5620 ResourceManager
15532 NodeManager
6140 NameNode

C:\hadoop-3.3.0\sbin>hdfs dfs -mkdir /input_dir

C:\hadoop-3.3.0\sbin>hdfs dfs -ls /
Found 1 items
drwxr-xr-x   - Anusree supergroup          0 2021-05-08 19:46 /input_dir

C:\hadoop-3.3.0\sbin>hdfs dfs -copyFromLocal C:\input.txt /input_dir

C:\hadoop-3.3.0\sbin>hdfs dfs -ls /input_dir
Found 1 items
-rw-r--r--   1 Anusree supergroup         36 2021-05-08 19:48 /input_dir/input.txt

C:\hadoop-3.3.0\sbin>hdfs dfs -cat /input_dir/input.txt
hello
world
hello
hadoop
bye
```

Lab 8

Write a Scala program to print numbers from 1 to 100 using for loop.

```
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ spark-shell
25/05/20 11:28:13 WARN Utils: Your hostname, bmscecse-HP-Elite-Tower-600-G9-Desktop-PC resolves to a loopback address: 127.0.1.1; using 10.124.3.80 instead (on interface eno1)
25/05/20 11:28:13 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.spark.unsafe.Platform (file:/opt/spark/jars/spark-unsafe_2.12-3.0.3.jar) to constructor java.nio.DirectByteBuffer(long,int)
WARNING: Please consider reporting this to the maintainers of org.apache.spark.unsafe.Platform
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
25/05/20 11:28:13 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
Spark context Web UI available at http://10.124.3.80:4040
Spark context available as 'sc' (master = local[*], app id = local-1747720695950).
Spark session available as 'spark'.
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 3.0.3
      /_/

Using Scala version 2.12.10 (OpenJDK 64-Bit Server VM, Java 11.0.26)
Type in expressions to have them evaluated.
Type :help for more information.

scala> for (i <- 1 to 100) print(i + " ")
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 6
5 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100
```

Lab 9

Using RDD and FlatMap count how many times each word appears in a file and write out a list of words whose count is strictly greater than 4 using Spark.

```
scala> val rdd = spark.sparkContext.textFile("file:/home/bmscecse/Desktop/scala")
rdd: org.apache.spark.rdd.RDD[String] = file:/home/bmscecse/Desktop/scala MapPartitionsRDD[1] at textFile at <console>:23

scala> val counts = rdd.flatMap(_.split("\\s+")).map(word => (word.toLowerCase, 1)).reduceByKey(_ + _).filter(_._2 > 4)
counts: org.apache.spark.rdd.RDD[(String, Int)] = MapPartitionsRDD[5] at filter at <console>:25

scala> counts.collect().foreach{ case (word, count) => println(s"$word $count") }
spark 6

scala>
```

Lab 10

Write a simple streaming program in Spark to receive text data streams on a particular port, perform basic text cleaning (like white space removal, stop words removal, lemmatization, etc.), and print the cleaned text on the screen. (Open Ended Question).

```
# Install NLTK and download required data
(run once) !pip install nltk

import nltk
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('wordnet')

from pyspark.sql import SparkSession
from pyspark.sql.functions import col, lower,
regexp_replace, split, explode, udf from pyspark.sql.types
import ArrayType, StringType from pyspark.ml.feature
import StopWordsRemover from nltk.stem import
WordNetLemmatizer

# Initialize SparkSession spark =
SparkSession.builder.appName("TextProcessing").getO
rCreate()

# Define your input lines
lines = [
    "Hello, I hate you.",
    "I hate that I love you.",
    "Don't want to, but I can't put",
    "nobody else above you."
]

# Create DataFrame from lines
df = spark.createDataFrame(lines, "string").toDF("value")

# Step 1: Lowercase and remove punctuation df_clean =
df.select(regexp_replace(lower(col("value")), "[^a-zA-Z\\s]",
"").alias("cleaned"))

# Step 2: Tokenize the cleaned text df_tokens =
df_clean.select(split(col("cleaned"),
"\\s+").alias("tokens"))

# Step 3: Remove stop words
```

```
remover = StopWordsRemover(inputCol="tokens", outputCol="filtered") df_filtered = remover.transform(df_tokens)

# Step 4: Lemmatization using NLTK WordNetLemmatizer with UDF lemmatizer = WordNetLemmatizer()

def lemmatize_words(words):
    return [lemmatizer.lemmatize(word) for word in words]

lemmatize_udf = udf(lemmatize_words, ArrayType(StringType()))
df_lemmatized = df_filtered.withColumn("lemmatized", lemmatize_udf(col("filtered")))

df_lemmatized.select(explode(col("lemmatized")).alias("word")).show(truncate=False)
```

```
Requirement already satisfied: nltk in /usr/local/lib/python3.11/dist-packages (3.9.1)
Requirement already satisfied: click in /usr/local/lib/python3.11/dist-packages (from nltk) (8.2.0)
Requirement already satisfied: joblib in /usr/local/lib/python3.11/dist-packages (from nltk) (1.5.0)
Requirement already satisfied: regex>=2021.8.3 in /usr/local/lib/python3.11/dist-packages (from nltk) (2024.11.6)
Requirement already satisfied: tqdm in /usr/local/lib/python3.11/dist-packages (from nltk) (4.67.1)
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
[nltk_data] Downloading package wordnet to /root/nltk_data...
+------+
|word  |
+------+
|hello |
|hate  |
|hate  |
|love  |
|dont  |
|want  |
|cant  |
|put   |
|nobody|
|else  |
+------+
```