VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT on

Big Data Analytics (23CS6PCBDA)

Submitted by:

Ranjan Devi (1BM22CS219)

Under the Guidance of
Vikranth B.M.
Assistant Professor, BMSCE

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 March 2025 - June 2025

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 **Ranjan Devi (1BM22CS219),** 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 **Big Data Analytics –(23CS6PCBDA)** work prescribed for the said degree.

Vikranth B.M
Associate Professor
Department of CSE
BMSCE, Bengaluru

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

Table Of Contents

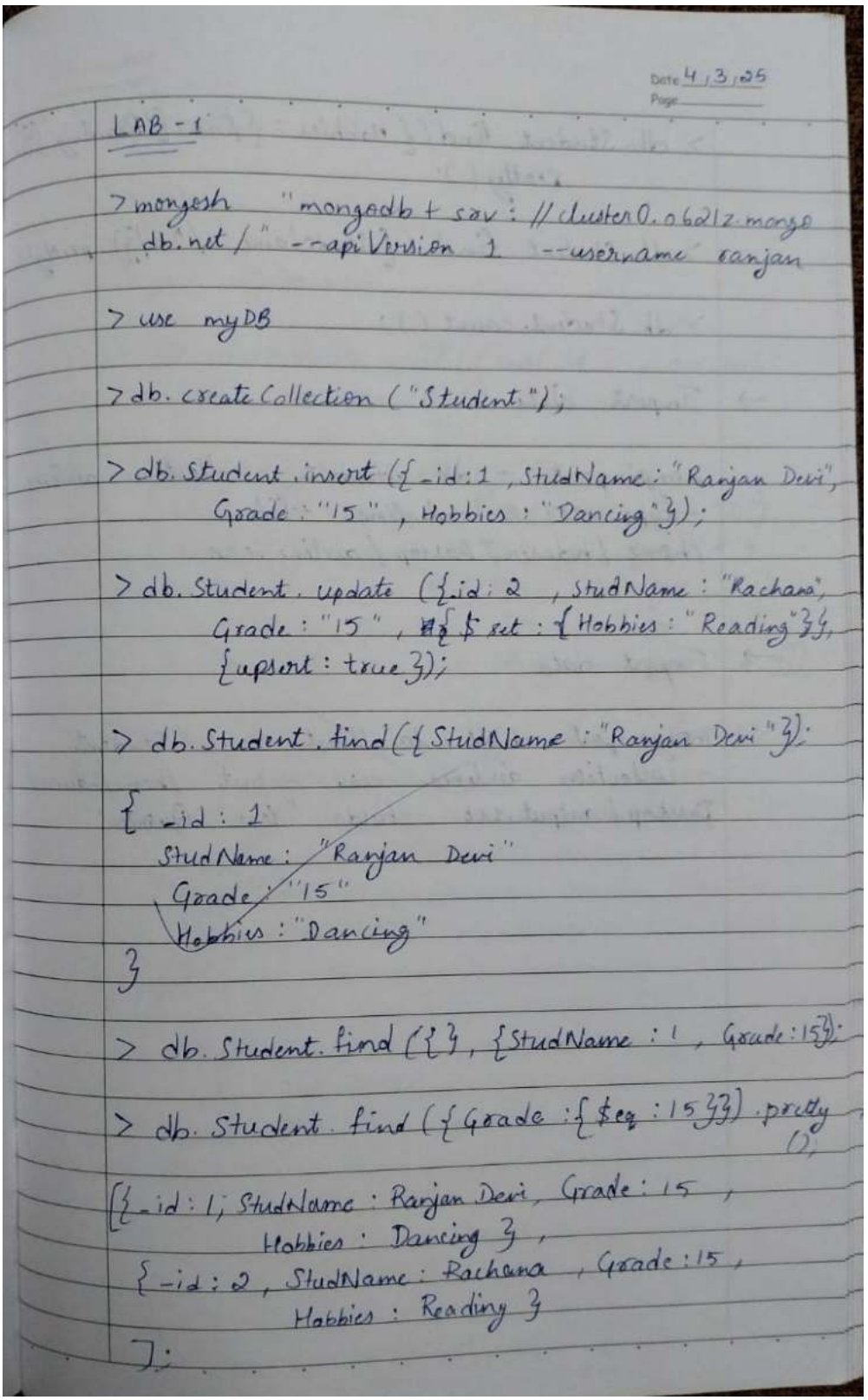
Github Link: https://github.com/ranjandevi219/BDA.git

Program 1

MongoDB- CRUD Operations Demonstration (Practice and Self Study)

- Created a database named **myDB** and verified its existence.
- Created and dropped collections like **Student** and **Students**.
- Inserted student data into collections.
- Performed **upsert** to insert or update a student record.
- Used to find queries with various filters: by name, grade, hobbies, regex, etc.
- Retrieved specific fields while suppressing _id.
- Counted total documents and documents with specific criteria.
- Sorted records in ascending and descending order.
- Imported data from a CSV file and exported data to a CSV file.
- Used save() to insert or replace documents.
- Added, removed, and set fields to null in documents.
- Retrieved limited records and skipped initial entries.
- Created a **food** collection with arrays and queried arrays by value, index, size, etc.
- Updated specific elements in an array.
- Practiced query optimizations using $in, $all, $ne, $regex, $slice, and more.

Observation:



Date 4/3/25
Page

LAB - 1

> mongosh "mongodb+srv://cluster0.o62lz.mongodb.net/" --apiVersion 1 --username ranjan

> use myDB

> db.createCollection("Student");

> db.Student.insert({_id:1, StudName: "Ranjan Devi", Grade: "15", Hobbies: "Dancing"});

> db.Student.update({_id: 2, StudName: "Rachana", Grade: "15", $set: {Hobbies: "Reading"}}, {upsert: true});

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

{
_id: 1
StudName: "Ranjan Devi"
Grade: "15"
Hobbies: "Dancing"
}

> db.Student.find({}, {StudName: 1, Grade: 15});

> db.Student.find({Grade: {$eq: 15}}).pretty();

[{_id: 1, StudName: Ranjan Devi, Grade: 15, Hobbies: Dancing},
{_id: 2, StudName: Rachana, Grade: 15, Hobbies: Reading}
];

```
> db.customerdb.aggregate ([{$group:
  {_id:"$cust_id", min_balance:{$min:
  "$acc_bal"}, max_balance:{$max:
  "$acc_bal"}}]);
```

(2) E-commerce platform.

```
> db.createCollection("Products");

> db.Products.insertMany([
  {product_id:"12345", name:"Smartphone",
  category:"Electronics", price:299.99, quantity:50,
  "  "  "  "
  "  "  "  "
]);

> db.createCollection("Users");

> db.Users.insertMany([
  {user_id:"789gh", name:"John",
  email:"john.doe@email.com", cart:[
  {product_id:"12345", quantity:2},
  {product_id:"23456", quantity:1}
  ], orders:[{order_id:"order123",
  order_date:ISODate("  "),
  "  "  "
  "  "  "
  "  "  "
},
]);
```

> db.createCollection("Orders");

```
>db.Orders.insertMany([
  { order_id: "order123",
    user_id: "789ghi",
    order_date: ISODate(" ");
    products: [{
      product_id: "12345", quantity: 1,
          price: 299.99},
    {product_id: "23456", quantity: 2,
          price: 49.99}],
    total_price: 399.97
}]);
```

→ Retrieve all products
```
>db.Products.find();
```

→ Retrieve products in specific category
```
>db.Products.find({category: "Electronics"});
```

→ Retrive products quantity greater than 0.
```
>db.Products.find({quantity: {$gt: 0}});
```

→ Retrieve sorted by price
```
>db.Products.find({}).sort({price:1});
```

→ Retrive price less than or equal to 100.
```
>db.Products.find({price: {$lte: 100}});
```

→
```
>db.Users.find({user_id: "123abc"},
          {orders:1});
```

→ Retrieve total price of orders placed by User.

```
db.Users.aggregate([{
  $match: { user_id: "123abc" }},
  { $unwind: "$orders" },
  { $group: { _id: "$user_id",
    totalprice: { $sum: "$orders.total_price" }
  }}]);
```

→ Calculate total number of products in each category.

```
db.Products.aggregate([{$group: { _id: "$category",
    total_products: { $sum: 1 }}}]);
```

OUTPUT:

```
[{ _id: 'Footwear', total: 1},
  { _id: 'Electronics', total: 3},
  { _id: 'Clothing', total: 1}
];
```

→ Calculate total price of products in each category

```
db.Products.aggregate([{$group: { _id:
"$category", total_price: { $sum: { $multiply:
["$price", "$quantity"] }}}
```

→ Find average price of products

```
db.Products.aggregate([{$group:{_id:null,
  average_price:{$avg:"$price"}}}]);
```

→ Find products with quantity less than 10.

```
db.Products.find({quantity:{$lt:10}});
```

→ Sort products by Price in Descending Order

```
db.Products.find({}).sort({price:-1});
```

→ Calculate Total Price of Orders Placed by each user.

```
db.Orders.aggregate([{$group:{_id:"user_id",
  total_spent:{$sum:"total_price"}}}]);
```

→ Find average total price of orders

```
db.Orders.aggregate([{$group:{_id:null,
  average_order_price:{$avg:"total_price"}}}]);
```

Code with Output:

```
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ mongosh
Current Mongosh Log ID: 6833f9c9126af1945c47586f
Connecting to:          mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.0.1
Using MongoDB:          7.0.2
Using Mongosh:          2.0.1
mongosh 2.5.1 is available for download: https://www.mongodb.com/try/download/shell

For mongosh info see: https://docs.mongodb.com/mongodb-shell/

------
   The server generated these startup warnings when booting
   2025-05-26T10:46:48.806+05:30: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongo
db.org/core/prodnotes-filesystem
   2025-05-26T10:46:50.937+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestric
ted
------

test> use MyDB
switched to db MyDB
MyDB> db
MyDB
MyDB> show dbs
admin           40.00 KiB
config          72.00 KiB
local           80.00 KiB
myNewDatabase   72.00 KiB
MyDB> db.createCollection("Student");
{ ok: 1 }
MyDB> db.Student.insert({_id:1,Name:"Preeti",Grade:"V",Hobbies:"Dancing"},{_id:2,Name:"Prajwal",Grade:"V",Hobbies:"Drawing"});
DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or bulkWrite.
{ acknowledged: true, insertedIds: { '0': 1 } }
MyDB> db.find();
TypeError: db.find is not a function
MyDB> db.Student.find();
[ { _id: 1, Name: 'Preeti', Grade: 'V', Hobbies: 'Dancing' } ]
MyDB> db.Student.insertMany({_id:2,Name:"Rachana",Grade:"V",Hobbies:"Painting"},{_id:3,Name:"Prajwal",Grade:"V",Hobbies:"Drawing"});
MongoInvalidArgumentError: Argument "docs" must be an array of documents
MyDB> db.Student.insertMany([{_id:2,Name:"Rachana",Grade:"V",Hobbies:"Painting"},{_id:3,Name:"Prajwal",Grade:"V",Hobbies:"Drawing"}]);
{ acknowledged: true, insertedIds: { '0': 2, '1': 3 } }
MyDB> db.Student.update({_id:2,Name:"Rachana",Grade:"V"},{$set:{Hobbies:"Singing"}},{upsert:true});
DeprecationWarning: Collection.update() is deprecated. Use updateOne, updateMany, or bulkWrite.
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
MyDB> db.Student.find();
[
  { _id: 1, Name: 'Preeti', Grade: 'V', Hobbies: 'Dancing' },
  { _id: 2, Name: 'Rachana', Grade: 'V', Hobbies: 'Singing' },
  { _id: 3, Name: 'Prajwal', Grade: 'V', Hobbies: 'Drawing' }
]
```

```
  { _id: 1, fruits: [ 'grapes', 'banana', 'apple' ] },
  {
    _id: 2,
    fruits: [ 'grapes', 'mango', 'cherry' ],
    price: [ { grapes: 80, mango: 100, cherry: 200 } ]
  },
  { _id: 3, fruits: [ 'banana', 'mango' ] }
]
MyDB> db.food.find().pretty();
[
  { _id: 1, fruits: [ 'grapes', 'banana', 'apple' ] },
  {
    _id: 2,
    fruits: [ 'grapes', 'mango', 'cherry' ],
    price: [ { grapes: 80, mango: 100, cherry: 200 } ]
  },
  { _id: 3, fruits: [ 'banana', 'mango' ] }
]
MyDB> db.createCollection("customer");
{ ok: 1 }
MyDB> var val=[{custid:1,accbal:100,acctype:"A"},{custid:1,accbal:100,acctype:"B"},{custid:2,accbal:200,accty
ype:"B"}];
Uncaught:
SyntaxError: Unexpected token, expected "," (1:144)

> 1 | var val=[{custid:1,accbal:100,acctype:"A"},{custid:1,accbal:100,acctype:"B"},{custid:2,accbal:200,accty
ype:"B"}];
    |                                                                                                    ^
  2 |

MyDB> var val=[{custid:1,accbal:100,acctype:"A"},{custid:1,accbal:100,acctype:"B"},{custid:2,accbal:200,accty
ype:"B"}];

MyDB> db.customer.insert(val);
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("683405cb126af1945c475870"),
    '1': ObjectId("683405cb126af1945c475871"),
    '2': ObjectId("683405cb126af1945c475872"),
    '3': ObjectId("683405cb126af1945c475873")
  }
}
MyDB> db.customer.aggregate({$group:{_id:'$custid',totalbal:{$sum:'$accbal'}}});
[ { _id: 1, totalbal: 200 }, { _id: 2, totalbal: 400 } ]
MyDB> db.customers.aggregate ( {$match:{AcctType:"S"}},{$group : { _id : "$custID",TotAccBal :
Uncaught:
SyntaxError: Unexpected character '"'. (1:43)

> 1 | db.Customers.aggregate ( {$match:{AcctType:"S"}},{$group : { _id : "$custID",TotAccBal :
```

```
  { _id: 1, fruits: [ 'grapes', 'banana', 'apple' ] },
  {
    _id: 2,
    fruits: [ 'grapes', 'mango', 'cherry' ],
    price: [ { grapes: 80, mango: 100, cherry: 200 } ]
  },
  { _id: 3, fruits: [ 'banana', 'mango' ] }
]
MyDB> db.food.find().pretty();
[
  { _id: 1, fruits: [ 'grapes', 'banana', 'apple' ] },
  {
    _id: 2,
    fruits: [ 'grapes', 'mango', 'cherry' ],
    price: [ { grapes: 80, mango: 100, cherry: 200 } ]
  },
  { _id: 3, fruits: [ 'banana', 'mango' ] }
]
MyDB> db.createCollection("customer");
{ ok: 1 }
MyDB> var val=[{custid:1,accbal:100,acctype:"A"},{custid:1,accbal:100,acctype:"B"},{custid:2,accbal:200,accty
ype:"B"});
Uncaught:
SyntaxError: Unexpected token, expected "," (1:144)

> 1 | var val=[{custid:1,accbal:100,acctype:"A"},{custid:1,accbal:100,acctype:"B"},{custid:2,accbal:200,accty
ype:"B"});
    |
      ^
  2 |

MyDB> var val=[{custid:1,accbal:100,acctype:"A"},{custid:1,accbal:100,acctype:"B"},{custid:2,accbal:200,accty
ype:"B"}];

MyDB> db.customer.insert(val);
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("683405cb126af1945c475870"),
    '1': ObjectId("683405cb126af1945c475871"),
    '2': ObjectId("683405cb126af1945c475872"),
    '3': ObjectId("683405cb126af1945c475873")
  }
}
MyDB> db.customer.aggregate({$group:{_id:'$custid',totalbal:{$sum:'$accbal'}}});
[ { _id: 1, totalbal: 200 }, { _id: 2, totalbal: 400 } ]
MyDB> db.Customers.aggregate ( {$match:{AcctType:"S"}},{$group : { _id : "$custID",TotAccBal :
Uncaught:
SyntaxError: Unexpected character '"'. (1:43)

> 1 | db.Customers.aggregate ( {$match:{AcctType:"S"}},{$group : { _id : "$custID",TotAccBal :
```

```
MyDB> db.customer.aggregate({$match:{acctype:'A'}},{$group:{_id:"$custid",totalbal:{$sum:'$accbal'}}});
[ { _id: 1, totalbal: 100 }, { _id: 2, totalbal: 200 } ]
MyDB> db.customer.aggregate({$match:{acctype:'A'}},{$group:{_id:"$custid",totalbal:{$sum:'$accbal'}}},{$mat$

MyDB> db.customer.aggregate({$match:{acctype:'A'}},{$group:{_id:"$custid",totalbal:{$sum:'$accbal'}}},{$match

MyDB> db.customer.aggregate({$match:{acctype:'A'}},{$group:{_id:"$custid",totalbal:{$sum:'$accbal'}}},{$match
[ { _id: 2, totalbal: 200 } ]
MyDB> S
```

Program 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 ofEmployees.

Observation:

25

Date / /
Page

CASANDRA :

```
create keyspace ;

create keyspace students with replication =
{'class' : 'SimpleStrategy', 'replication_factor' : 1};

Describe keyspaces ;

select * from system_schema.keyspaces ;

Use students ;

create table students_info (Roll_No int
  primary key , studname text , dob timestamp,
  last_exam_perc double);

describe tables ;

describe tables student_info ;
```

CRUD

```
Begin Batch
insert into students_info (Rollno , studname ,
    dob , last_exam_perc)
  values (1, 'Asha', '2012-03-12', 79.9)

insert into students_info (Rollno , studname , dob,
    last_exam_perc)
  values (2, 'Kiran', '2012-03-12', 89.9)
insert into students_info (Rollno , studname , dob,
    last_exam_perc)
```

values (3, 'Tarun', '2012-03-12', 78.9)

Apply Batch;

select * from students-info;

→

Roll_no	dob	last_exam_perc	studname
1	2012-03-12	79.9	Asha
2	2012-03-12	89.9	Kiran
3	2012-03-12	78.9	Tarun

select * from students-info where Rollno in (1, 2);

select * from students-info where studname = 'Asha';

create index on students-info (studname);

select Rollno, studname from students-info limit 2;

select Rollno as "USN" from students-info;

→

USN
1
2
3

update students-info set studname = 'David' where Rollno = 2;

delete last_exam-perc from students-info where Rollno = 2;

alter table students_info add hobby set <text>;

update students_info
set hobby = hobby + {"chess", "tennis"}
where RollNo = 1;

create table library (counter_value = counter_value + 1 where book_name = 'BDA' and studname = 'Jeet';

2); create table user login (id int primary key, pss text);

'); insert into userlogin (id, pss) values (1, 'infy') using TTL 30;

2; select TTL (pss) from userlogin where id = 1;

→ TTL (pss)
28

→ RollNo dob hobby language last_exam_perc
1 2012-03-11 {chess, tennis} null 79.9

→ studname
Asha

Code with Output:

```
cqlsh> CREATE KEYSPACE Student WITH REPLICATION= {'class':'SimpleStrategy','replication_factor':1};
cqlsh> describe keysapces;
'keysapces' not found in keyspaces
cqlsh> describe keyspaces;

student   system       system_distributed  system_traces  system_virtual_schema
students  system_auth  system_schema       system_views

cqlsh> use students;
cqlsh:students> create table st_info(rollno int primary key,name text,doj timestamp,percent double);
cqlsh:students> describe tables;

library_book  st_info  students_info  userlogin

cqlsh:students> describe table<st_info>;
Improper describe command.
cqlsh:students> describe table st_info;

CREATE TABLE students.st_info (
    rollno int PRIMARY KEY,
    doj timestamp,
    name text,
    percent double
) WITH additional_write_policy = '99p'
    AND bloom_filter_fp_chance = 0.01
    AND caching = {'keys': 'ALL', 'rows_per_partition': 'NONE'}
    AND cdc = false
    AND comment = ''
    AND compaction = {'class': 'org.apache.cassandra.db.compaction.SizeTieredCompactionStrategy', 'max_threshold': '32', 'min_thr
eshold': '4'}
    AND compression = {'chunk_length_in_kb': '16', 'class': 'org.apache.cassandra.io.compress.LZ4Compressor'}
    AND memtable = 'default'
    AND crc_check_chance = 1.0
    AND default_time_to_live = 0
    AND extensions = {}
    AND gc_grace_seconds = 864000
    AND max_index_interval = 2048
    AND memtable_flush_period_in_ms = 0
    AND min_index_interval = 128
    AND read_repair = 'BLOCKING'
    AND speculative_retry = '99p';
cqlsh:students> begin batch
           ... insert into st_info(rollno,name,doj,percent)
```

```
cqlsh:students> select * from st_info;

 rollno | doj                             | name    | percent
--------+---------------------------------+---------+---------
      1 | 2010-02-28 18:30:00.000000+0000 |  preeti |      90
      2 | 2010-03-19 18:30:00.000000+0000 | prajwal |      89
      4 | 2010-04-22 18:30:00.000000+0000 | rachana |      90

(3 rows)
cqlsh:students> select * from st_info where rollno in(1,2);

 rollno | doj                             | name    | percent
--------+---------------------------------+---------+---------
      1 | 2010-02-28 18:30:00.000000+0000 |  preeti |      90
      2 | 2010-03-19 18:30:00.000000+0000 | prajwal |      89

(2 rows)
cqlsh:students> select  * from st_info where name="preeti";
SyntaxException: line 1:42 no viable alternative at input ';' (...* from st_info where name=["preet]i";)
cqlsh:students> create index on st_info(name);
cqlsh:students> select  * from st_info where name="preeti";
SyntaxException: line 1:42 no viable alternative at input ';' (...* from st_info where name=["preet]i";)
cqlsh:students> select * from st_info where name='preeti';

 rollno | doj                             | name   | percent
--------+---------------------------------+--------+---------
      1 | 2010-02-28 18:30:00.000000+0000 | preeti |      90

(1 rows)
cqlsh:students> select rollno,name,percent from st_info limit 2;

 rollno | name    | percent
--------+---------+---------
      1 |  preeti |      90
      2 | prajwal |      89

(2 rows)
cqlsh:students> slect rollno as usn from st_info;
SyntaxException: line 1:0 no viable alternative at input 'slect' ([slect]...)
cqlsh:students> select rollno as usn from st_info;

 usn
-----
   1
```

```
 usn
-----
   1
   2
   4

(3 rows)
cqlsh:students> create table library(c_val counter,book_name varchar,stud_name varchar,primary key(book_name,stud_name
cqlsh:students> update library set c_val=c_val+1 where book_name='BDA' and stud_name='preeti';
cqlsh:students> create table userlogin(id int primary key,pass text);
AlreadyExists: Table 'students.userlogin' already exists
cqlsh:students> create table login(id int primary key,pass text);
cqlsh:students> insert into login(id,pass) values(1,'infy')using ttl 30;
cqlsh:students> select ttl(pass) from login where id=1;

 ttl(pass)
-----------
         3
```

Program 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

Observation:

15/4/25

Date / /
Page

CASSANDRA 2

:$ cqlsh

```
> create table Library_Info (
      Stud_Id INT, Stud_Name TEXT,
    Book_Name TEXT, Book_Id INT,
    Date_of_Issue DATE,
    PRIMARY KEY (Stud_ID, Book_Id));

>create table Library_Counter (
        Stud_Id INT PRIMARY KEY,
        Counter_value COUNTER
    );

> Begin UNLOGGED BATCH
    UPDATE Library_Counter
    SET Counter_value = Counter_value + 1
    WHERE Stud_Id = 101;

    UPDATE Library_Counter
    SET counter_value = Counter_value + 1
    WHERE Stud_Id = 102;
    APPLY BATCH;

> Begin UNLOGGED BATCH
    INSERT into Library_Info (Stud_Id, Stud_Name
      Book_Name, Book_Id, Date_of_Issue)
    VALUES (101, 'Alice', 'The Great Gatsby',
        501, '2025-04-10');

    INSERT into Library_Info (Stud_Id, Stud_Name
      Book_Name, Book_Id, Date_of_Issue)
```

Date / /
Page

```
VALUES (102, 'Bob', '1984', 502,
        '2025-04-11');
APPLY BATCH;
```

> SELECT * FROM Library_Info;

Stud_id	book_id	Stud_name	book_name	date
101	501	Alice	The Great Gatsby	2025-04-10
102	502	Bob	1984	2025-04-11

> select * FROM Library_Counter;

Stud_id	counter_value
101	1
102	1

> COPY Library_Info TO 'library_info.csv'
WITH HEADER = true;

Using 16 CHILD Processes
4 rows exported to 1 files in 0.100 seconds.

S. Pradeep
11/4/25

Code with Output:

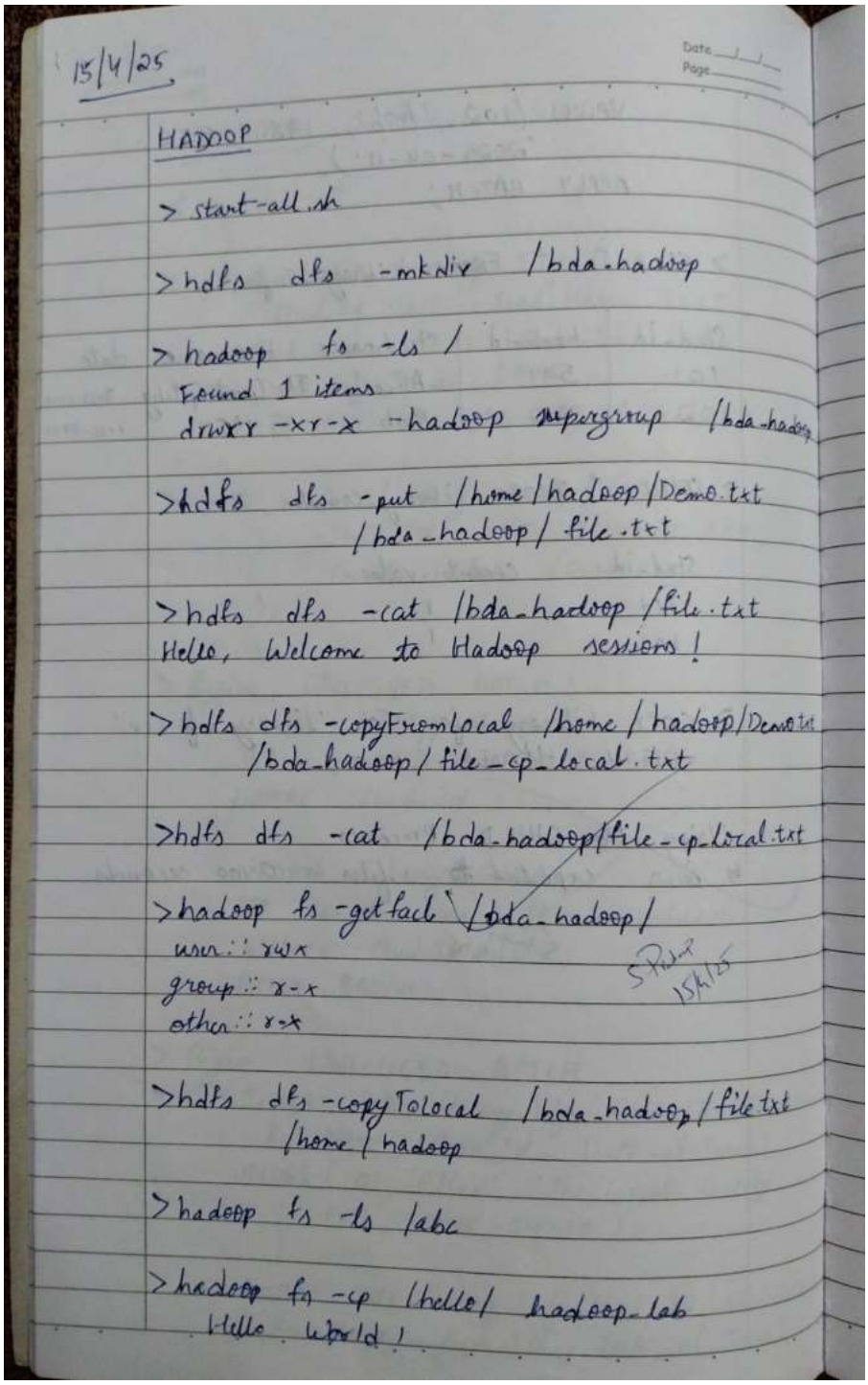
```
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.8 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> create keyspace library with replication={'class':'Simplestrategy','replication_factor':1};
ConfigurationException: Unable to find replication strategy class 'org.apache.cassandra.locator.Simplestrategy'
cqlsh> create keyspace library with replication={'class':'Simplestrategy','replication_factor':1};exit
ConfigurationException: Unable to find replication strategy class 'org.apache.cassandra.locator.Simplestrategy'
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.8 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> create keyspace library with replication={'class':'SimpleStrategy','replication_factor':1};exit
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.8 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> create keyspace library with replication={'class':'SimpleStrategy','replication_factor':1};
AlreadyExists: Keyspace 'library' already exists
cqlsh> exit
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.8 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> create keyspace libraries with replication={'class':'SimpleStrategy','replication_factor':1};
cqlsh> keyspaces
   ...
cqlsh> describe keyspaces;

libraries  students     system_distributed  system_views
library    system       system_schema       system_virtual_schema
student    system_auth  system_traces

cqlsh> use libraries;
cqlsh:libraries> create table l_info(sid int primary key, c_val counter, sname varchar,bname varchar,bid int,doi timestamp);
InvalidRequest: Error from server: code=2200 [Invalid query] message="Cannot mix counter and non counter columns in the same tabl
e"
cqlsh:libraries> create table l_info(sid int primary key, sname varchar,bname varchar,bid int,doi timestamp);
cqlsh:libraries> create table count(sid int primary key,c_val counter);
cqlsh:libraries> begin batch
             ... insert into l_info(sid,sname,bname,bid,doi)
             ... values(112,'alice','bda',1,'2020-03-03')
             ... insert into l_info(sid,sname,bname,bid,doi)
             ... values(113,'preeti','cn',2,'2020-03-04')
             ... apply batch;
cqlsh:libraries> update l_info
```

```
cqlsh:libraries> select * from l_info;

 sid | bid | bname | doi                                 | sname
-----+-----+-------+-------------------------------------+--------
 113 |   2 |    cn | 2020-03-03 18:30:00.000000+0000 | preeti
 112 |   1 |   bda | 2020-03-02 18:30:00.000000+0000 |  alice

(2 rows)
cqlsh:libraries> select * from count;

 sid | c_val
-----+-------
 112 |     1

(1 rows)
cqlsh:libraries>
```

Program 4

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

Observation:



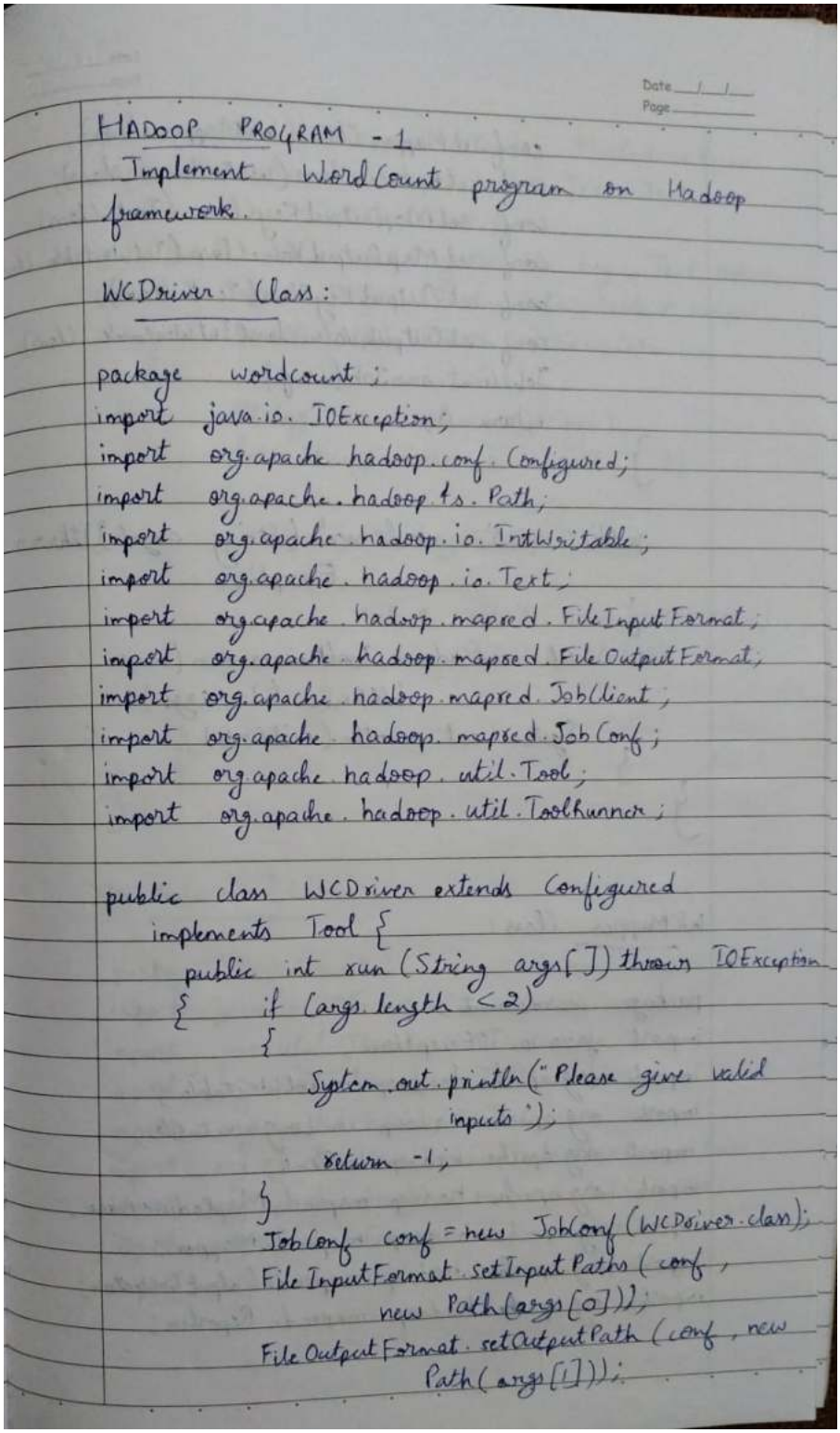
Code with Output:

```
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:~$ hdfs dfs -mkdir /bda_hadoop
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -ls /
Found 4 items
drwxr-xr-x   - hadoop supergroup          0 2025-04-15 15:07 /abc
drwxr-xr-x   - hadoop supergroup          0 2025-05-26 14:13 /bda_hadoop
drwxr-xr-x   - hadoop supergroup          0 2025-05-22 16:32 /pqr
drwxr-xr-x   - hadoop supergroup          0 2025-05-20 16:36 /rgs
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -put /home/hadoop/sample.txt /bda_hadoop/file.txt
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -copyFromLocal /home/hadoop/sample.txt /bda_hadoop/local.txt
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -cat /bda_hadoop/file.txt
hi how are you
how is your job
how is your family
how is your brother
how is your sister
eof
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -get /bda_hadoop/file.txt /home/hadoop/sample.txt
get: `/home/hadoop/sample.txt': File exists
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -get /bda_hadoop/file.txt /home/hadoop/get.txt
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hadoop fs -getfacl /bda_hadoop/
# file: /bda_hadoop
# owner: hadoop
# group: supergroup
user::rwx
group::r-x
other::r-x

hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -copyToLocal /bda_hadoop/file.txt /home/hadoop/tolocal.txt
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hadoop fs -cp /bda_hadoop /abc
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hadoop fs -ls /abc
Found 3 items
drwxr-xr-x   - hadoop supergroup          0 2025-05-26 14:28 /abc/bda_hadoop
-rw-r--r--   1 hadoop supergroup         55 2025-04-15 15:05 /abc/file.txt
-rw-r--r--   1 hadoop supergroup         55 2025-04-15 15:07 /abc/file_cp_.txt
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$
```

Program 5

Implement Wordcount program on Hadoop framework

Observation:



HADOOP PROGRAM - 1

Implement WordCount program on Hadoop framework.

WCDriver Class:

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

public class WCDriver extends Configured
    implements Tool {
    public int run (String args[]) throws IOException
    {   if (args.length < 2)
        {
            System.out.println("Please give valid
                    inputs");
            return -1;
        }
        JobConf conf = new JobConf(WCDriver.class);
        FileInputFormat.setInputPaths(conf,
                new Path(args[0]));
        FileOutputFormat.setOutputPath(conf, new
                Path(args[1]));
```

```
        conf.setMapperClass(WCMapper.class);
        conf.setReducerClass(WCReducer.class);
        conf.setMapOutputKeyClass(Text.class);
        conf.setMapOutputValueClass(IntWritable.class);
        conf.setOutputKeyClass(Text.class);
        conf.setOutputValueClass(IntWritable.class);
        JobClient.runJob(conf);
        return 0;
    }

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

WCMapper Class:

```
package wordcount;
import java.io.IOException;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.MapReduceBase;
import org.apache.hadoop.mapred.Mapper;
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reporter;
```

```
public class WCMapper extends MapReduceBase
implements Mapper <LongWritable, Text, Text,
                   IntWritable> {
    public void map (LongWritable key, Text value,
        OutputCollector <Text, IntWritable> output,
        Reporter rep) throws IOException
    {
        String line = value.toString();
        for (String word: line.split(" "))
        {
            if (word.length() > 0)
            {
                output.collect (new Text(word),
                    new IntWritable(1));
            }
        }
    }
}
```

WCReducer Class:

```
package wordcount;
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.OutCollector;
import org.apache.hadoop.mapred.Reducer;
import org.apache.hadoop.mapred.Reporter;
```

```
public class WCReducer extends MapReduceBase
implements Reducer <Text, IntWritable, Text,
        IntWritable> {
    public void reduce (Text key, Iterator
        <IntWritable> value, OutputCollector
        <Text, IntWritable> output,
            Reporter rep) throws IOException
    {
        int count = 0;
        while (value.hasNext())
        {
            IntWritable i = value.next();
            count += i.get();
        }
        output.collect (key, new IntWritable(count
    }
}
```

Input.txt
Hello World
Hello World
This is BDA Lab

Output:
Hello - 2
World - 2
This - 1
is - 1
BDA - 1
Lab - 1

Code with Output:

```
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]
localhost: namenode is running as process 12082.  Stop it first and ensure /tmp/hadoop-hadoop-namenode.pid file is empty before retry.
Starting datanodes
localhost: datanode is running as process 12255.  Stop it first and ensure /tmp/hadoop-hadoop-datanode.pid file is empty before retry.
Starting secondary namenodes [bmscecse-HP-Elite-Tower-600-G9-Desktop-PC]
bmscecse-HP-Elite-Tower-600-G9-Desktop-PC: secondarynamenode is running as process 12557.  Stop it first and ensure /tmp/hadoop-hadoop-secondarynamenode.pid file is empty before retry.
Starting resourcemanager
resourcemanager is running as process 12845.  Stop it first and ensure /tmp/hadoop-hadoop-resourcemanager.pid file is empty before retry.
Starting nodemanagers
localhost: nodemanager is running as process 13014.  Stop it first and ensure /tmp/hadoop-hadoop-nodemanager.pid file is empty before retry.
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC: $ jps
12082 NameNode
13014 NodeManager
15100 org.eclipse.equinox.launcher_1.6.1000.v20250227-1734.jar
17036 Jps
12845 ResourceManager
12557 SecondaryNameNode
12255 DataNode
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC: $ hdfs dfs -copyFromLocal /home/hadoop/sample.txt /bda_hadoop/input.txt
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC: $ hadoop jar /home/hadoop/Desktop/hdpwordcount.jar WCDriver /bda_hadoop/input.txt /bda_hadoop/output
Exception in thread "main" java.lang.ClassNotFoundException: WCDriver
        at java.base/java.net.URLClassLoader.findClass(URLClassLoader.java:476)
        at java.base/java.lang.ClassLoader.loadClass(ClassLoader.java:594)
        at java.base/java.lang.ClassLoader.loadClass(ClassLoader.java:527)
        at java.base/java.lang.Class.forName0(Native Method)
        at java.base/java.lang.Class.forName(Class.java:398)
        at org.apache.hadoop.util.RunJar.run(RunJar.java:321)
        at org.apache.hadoop.util.RunJar.main(RunJar.java:241)
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC: $ hadoop jar /home/hadoop/Desktop/hdpwordcount.jar hdpwordcount.WCDriver /bda_hadoop/input.txt /bda_hadoop/output
2025-05-26 14:40:01,404 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2025-05-26 14:40:01,440 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2025-05-26 14:40:01,440 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2025-05-26 14:40:01,446 WARN impl.MetricsSystemImpl: JobTracker metrics system already initialized!
2025-05-26 14:40:01,501 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
2025-05-26 14:40:01,545 INFO mapred.FileInputFormat: Total input files to process : 1
2025-05-26 14:40:01,567 INFO mapreduce.JobSubmitter: number of splits:1
2025-05-26 14:40:01,624 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local276129153_0001
2025-05-26 14:40:01,624 INFO mapreduce.JobSubmitter: Executing with tokens: []
2025-05-26 14:40:01,677 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
2025-05-26 14:40:01,679 INFO mapred.LocalJobRunner: OutputCommitter set in config null
2025-05-26 14:40:01,679 INFO mapreduce.Job: Running job: job_local276129153_0001
2025-05-26 14:40:01,680 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapred.FileOutputCommitter
2025-05-26 14:40:01,682 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2025-05-26 14:40:01,682 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ign
```

```
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC: $ hdfs dfs -ls /bda_hadoop/output
Found 2 items
-rw-r--r--   1 hadoop supergroup          0 2025-05-26 14:40 /bda_hadoop/output/_SUCCESS
-rw-r--r--   1 hadoop supergroup         75 2025-05-26 14:40 /bda_hadoop/output/part-00000
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC: $ hdfs dfs -cat /bda_hadoop/output/part-00000
are     1
brother 1
eof     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: $
```

Program 6

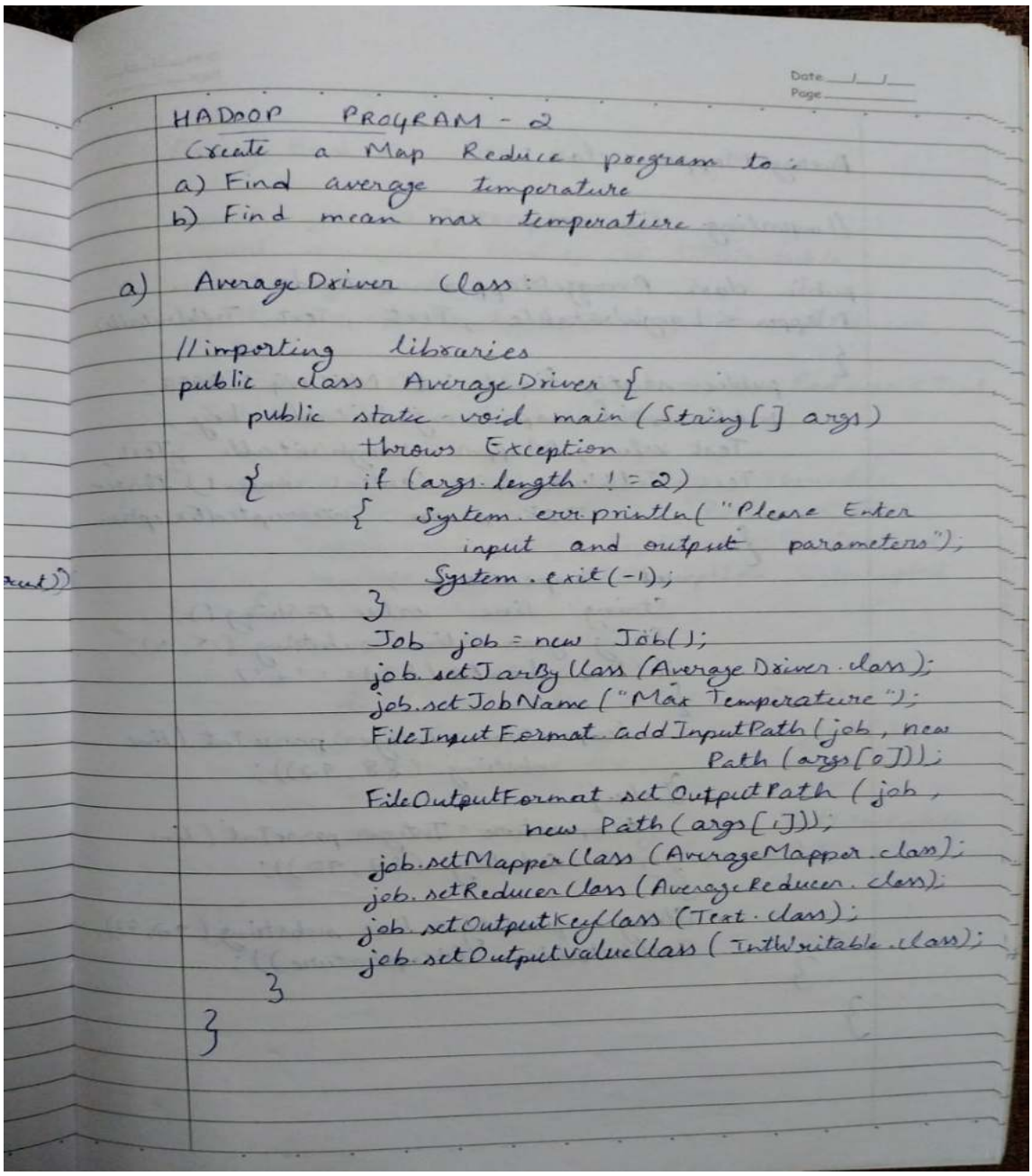
From the following link extract the weather data

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

a) Create a MapReduce program to find average temperature for each year from the NCDC data set.

b) find the mean max temperature for every month

Observation:



HADOOP PROGRAM - 2

Create a Map Reduce program to:

a) Find average temperature

b) Find mean max temperature

a) AverageDriver Class:

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

Date
Page

AverageMapper class:

```
//importing libraries

public class AverageMapper extends
Mapper <LongWritable , Text , Text, IntWritable>
{
    public static final int MISSING = 9999;
    public void map (LongWritable key,
      Text value, Mapper <LongWritable , Text,
      Text, IntWritable > .Context context) throws
              IOException , InterruptedException
      {
          int temperature ;
          String line = value.toString();
          String year = line.substring (15,19);
          if(line.charAt(87) == '+')
          {
              temperature = Integer.parseInt (line
                  .substring (88,92));
          }else{
              temperature = Integer.parseInt (line
                  .substring (87,92));
          }
          String quality = line.substring (92,93);
          IntWritable (temperature));
      }
}
```

Date __/__/__
Page

Average Reducer class:

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

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

b) MeanMaxDriver class :

```
//importing libraries

public class MeanMaxDriver
{
    public static void main (String [] args)
        throws Exception
    { if (args.length != 2)
        { System.err.println ("Please Enter");
          System.exit (-1);
        }
        Job job = new Job();
        job.setJarByClass (MeanMaxDriver.class);
        job.setJobName ("Max Temperature");
        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);
    }
}
```

MeanMaxMapper class :

```
// importing libraries

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

MeanMax Reducer class:

```
//importing libraries

public class MeanMaxReducer extends
Reducer<Text, IntWritable, Text, IntWritable>
{
    public void reduce (Text key, Iterable
    <IntWritable> values, Reducer<Text, IntWritable
     Text, IntWritable> Context context) throws
        IOException, InterruptedException
    {
        int max_temp = 0;
        int total_temp = 0;
        int count = 0;
        int days = 0;
        for (IntWritable value : values)
        {   int temp : value.get();
            if (temp > max_temp)
                max_temp = temp;
                count++;
            if (count == 3)
            {
                total_temp += max_temp;
                max_temp = 0;
                count = 0;
                days++;
            }
        }
    }
}
```

Code with Output:

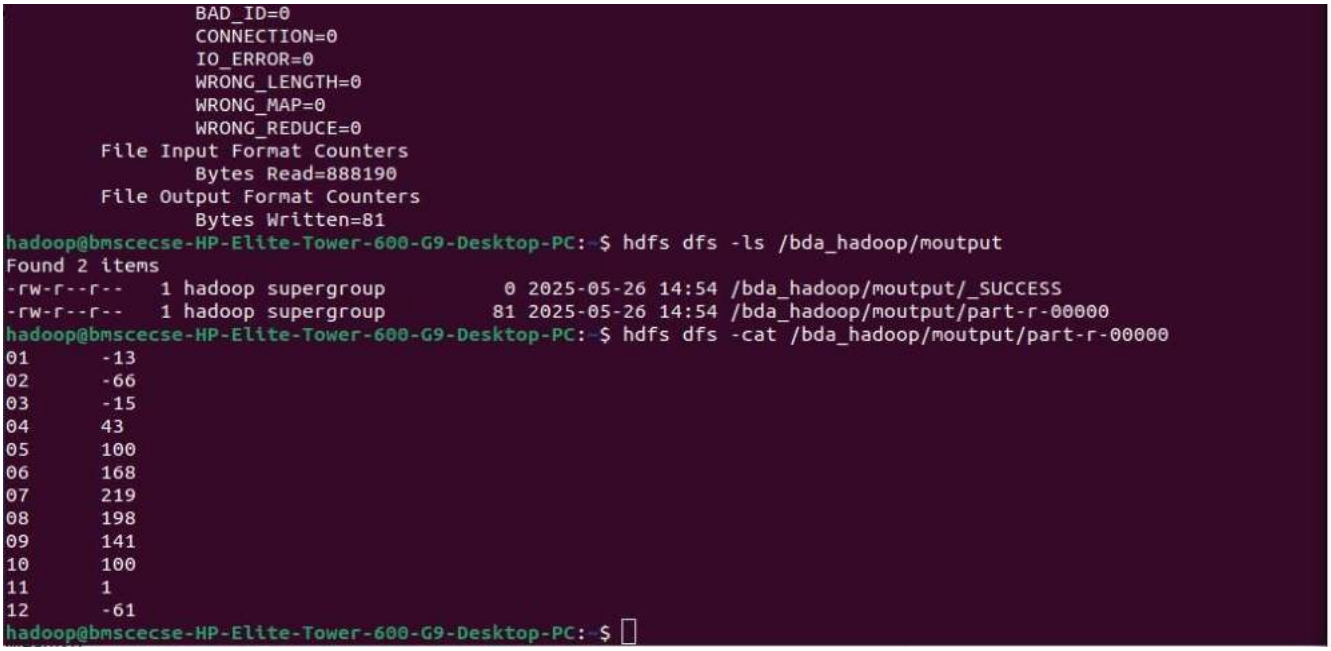
a) Average temperature

```
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ jps
12082 NameNode
17908 Jps
13014 NodeManager
15100 org.eclipse.equinox.launcher_1.6.1000.v20250227-1734.jar
12845 ResourceManager
12557 SecondaryNameNode
12255 DataNode
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -copyFromLocal /home/hadoop/Downloads/1901 /bda_hadoop/avinput.txt
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hadoop jar /home/hadoop/Desktop/WeatherAverage.jar WeatherAverage.AVDriver /bda_h
adoop/avinput.txt /bda_hadoop/avoutput
2025-05-26 14:49:09,290 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2025-05-26 14:49:09,327 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2025-05-26 14:49:09,327 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2025-05-26 14:49:09,380 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool inte
rface and execute your application with ToolRunner to remedy this.
2025-05-26 14:49:09,427 INFO input.FileInputFormat: Total input files to process : 1
2025-05-26 14:49:09,452 INFO mapreduce.JobSubmitter: number of splits:1
2025-05-26 14:49:09,510 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local1313646497_0001
2025-05-26 14:49:09,510 INFO mapreduce.JobSubmitter: Executing with tokens: []
2025-05-26 14:49:09,566 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
2025-05-26 14:49:09,566 INFO mapreduce.Job: Running job: job_local1313646497_0001
2025-05-26 14:49:09,567 INFO mapred.LocalJobRunner: OutputCommitter set in config null
2025-05-26 14:49:09,570 INFO output.PathOutputCommitterFactory: No output committer factory defined, defaulting to FileOutputCommitte
rFactory
2025-05-26 14:49:09,571 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2025-05-26 14:49:09,571 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:f
alse, ignore cleanup failures: false
2025-05-26 14:49:09,571 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
2025-05-26 14:49:09,620 INFO mapred.LocalJobRunner: Waiting for map tasks
2025-05-26 14:49:09,620 INFO mapred.LocalJobRunner: Starting task: attempt_local1313646497_0001_m_000000_0
2025-05-26 14:49:09,629 INFO output.PathOutputCommitterFactory: No output committer factory defined, defaulting to FileOutputCommitte
rFactory
2025-05-26 14:49:09,629 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2025-05-26 14:49:09,629 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:f
alse, ignore cleanup failures: false
2025-05-26 14:49:09,635 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
2025-05-26 14:49:09,637 INFO mapred.MapTask: Processing split: hdfs://localhost:9000/bda_hadoop/avinput.txt:0+888190
2025-05-26 14:49:09,666 INFO mapred.MapTask: (EQUATOR) 0 kvi 26214396(104857584)
2025-05-26 14:49:09,666 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100
2025-05-26 14:49:09,666 INFO mapred.MapTask: soft limit at 83886080
2025-05-26 14:49:09,666 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600
2025-05-26 14:49:09,666 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
2025-05-26 14:49:09,668 INFO mapred.MapTask: Map output collector class = org.apache.hadoop.mapred.MapTask$MapOutputBuffer
2025-05-26 14:49:09,730 INFO mapred.LocalJobRunner:
2025-05-26 14:49:09,731 INFO mapred.MapTask: Starting flush of map output
2025-05-26 14:49:09,731 INFO mapred.MapTask: Spilling map output
2025-05-26 14:49:09,731 INFO mapred.MapTask: bufstart = 0; bufend = 59076; bufvoid = 104857600
2025-05-26 14:49:09,731 INFO mapred.MapTask: kvstart = 26214396(104857584); kvend = 26188144(104752576); length = 26253/6553600
2025-05-26 14:49:09,739 INFO mapred.MapTask: Finished spill 0
2025-05-26 14:49:09,743 INFO mapred.Task: Task:attempt_local1313646497_0001_m_000000_0 is done. And is in the process of committing
2025-05-26 14:49:09,745 INFO mapred.LocalJobRunner: map
```

```
                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=888190
        File Output Format Counters
                Bytes Written=8
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -ls /bda_hadoop/avoutput
Found 2 items
-rw-r--r--   1 hadoop supergroup          0 2025-05-26 14:49 /bda_hadoop/avoutput/_SUCCESS
-rw-r--r--   1 hadoop supergroup          8 2025-05-26 14:49 /bda_hadoop/avoutput/part-r-00000
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -cat /bda_hadoop/avoutput/part-r-00000
1901    46
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$
```

b) Maximum temperature

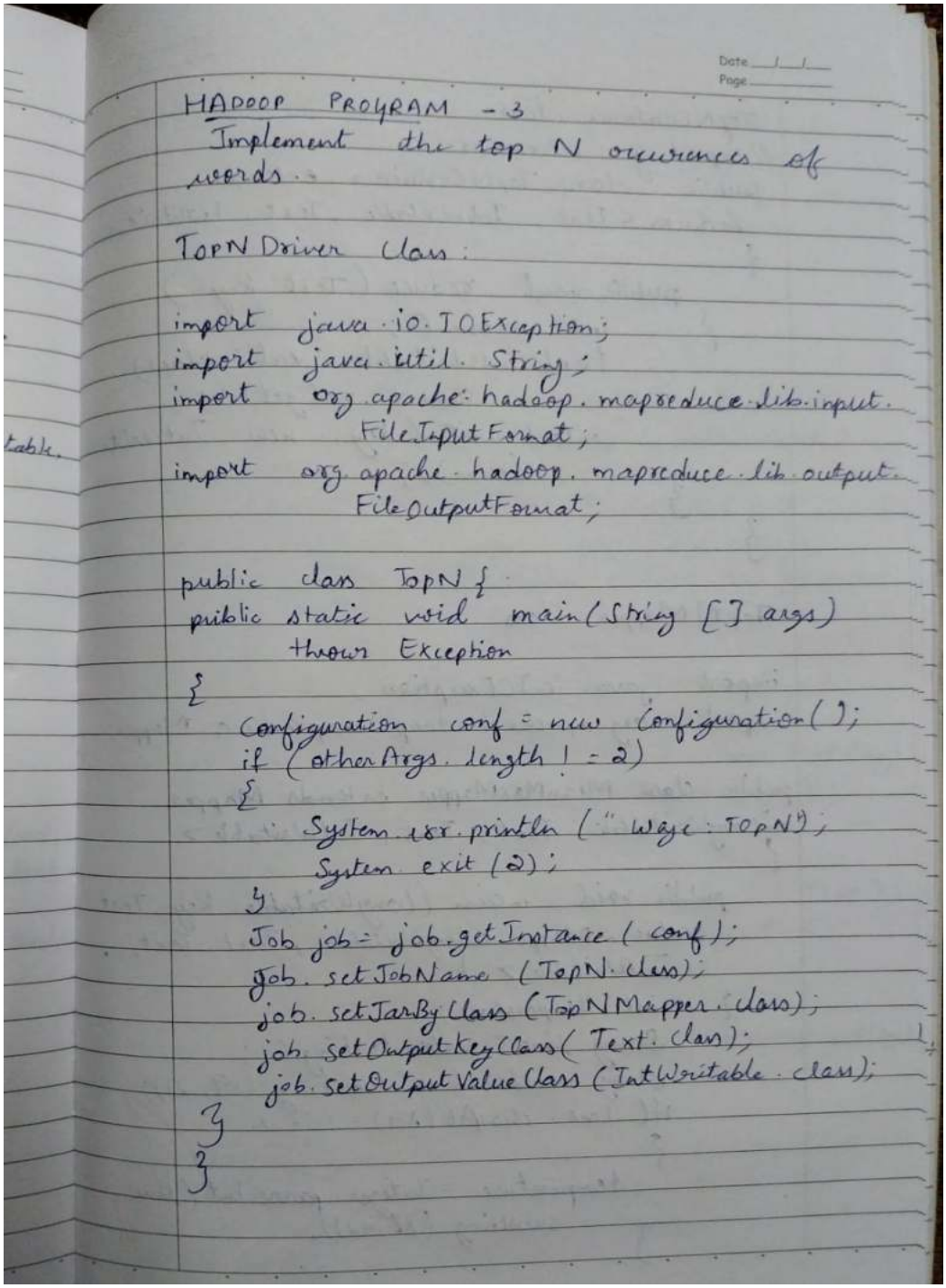
```
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ jps
18721 Jps
12082 NameNode
13014 NodeManager
15100 org.eclipse.equinox.launcher_1.6.1000.v20250227-1734.jar
12845 ResourceManager
12557 SecondaryNameNode
12255 DataNode
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -copyFromLocal /home/hadoop/Downloads/1901 /bda_ha
doop/minput.txt
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hadoop jar /home/hadoop/Desktop/meanTemp.jar Mean.MNDriver
/bda_hadoop/minput.txt /bda_hadoop/moutput
2025-05-26 14:54:41,993 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2025-05-26 14:54:42,029 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2025-05-26 14:54:42,029 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2025-05-26 14:54:42,083 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. I
mplement the Tool interface and execute your application with ToolRunner to remedy this.
2025-05-26 14:54:42,131 INFO input.FileInputFormat: Total input files to process : 1
2025-05-26 14:54:42,158 INFO mapreduce.JobSubmitter: number of splits:1
2025-05-26 14:54:42,216 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local862196817_0001
2025-05-26 14:54:42,216 INFO mapreduce.JobSubmitter: Executing with tokens: []
2025-05-26 14:54:42,272 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
2025-05-26 14:54:42,273 INFO mapreduce.Job: Running job: job_local862196817_0001
2025-05-26 14:54:42,273 INFO mapred.LocalJobRunner: OutputCommitter set in config null
2025-05-26 14:54:42,276 INFO output.PathOutputCommitterFactory: No output committer factory defined, defaulting
 to FileOutputCommitterFactory
2025-05-26 14:54:42,277 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2025-05-26 14:54:42,277 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders un
der output directory:false, ignore cleanup failures: false
2025-05-26 14:54:42,277 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.F
ileOutputCommitter
2025-05-26 14:54:42,319 INFO mapred.LocalJobRunner: Waiting for map tasks
2025-05-26 14:54:42,319 INFO mapred.LocalJobRunner: Starting task: attempt_local862196817_0001_m_000000_0
2025-05-26 14:54:42,328 INFO output.PathOutputCommitterFactory: No output committer factory defined, defaulting
 to FileOutputCommitterFactory
2025-05-26 14:54:42,329 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2025-05-26 14:54:42,329 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders un
der output directory:false, ignore cleanup failures: false
2025-05-26 14:54:42,335 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
2025-05-26 14:54:42,336 INFO mapred.MapTask: Processing split: hdfs://localhost:9000/bda_hadoop/minput.txt:0+88
8190
2025-05-26 14:54:42,366 INFO mapred.MapTask: (EQUATOR) 0 kvi 26214396(104857584)
2025-05-26 14:54:42,366 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100
2025-05-26 14:54:42,366 INFO mapred.MapTask: soft limit at 83886080
2025-05-26 14:54:42,366 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600
2025-05-26 14:54:42,366 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
2025-05-26 14:54:42,368 INFO mapred.MapTask: Map output collector class = org.apache.hadoop.mapred.MapTask$MapO
utputBuffer
2025-05-26 14:54:42,428 INFO mapred.LocalJobRunner:
2025-05-26 14:54:42,428 INFO mapred.MapTask: Starting flush of map output
2025-05-26 14:54:42,429 INFO mapred.MapTask: Spilling map output
2025-05-26 14:54:42,429 INFO mapred.MapTask: bufstart = 0; bufend = 45948; bufvoid = 104857600
```


BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0
File Input Format Counters
Bytes Read=888190
File Output Format Counters
Bytes Written=81
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -ls /bda_hadoop/moutput
Found 2 items
-rw-r--r-- 1 hadoop supergroup 0 2025-05-26 14:54 /bda_hadoop/moutput/_SUCCESS
-rw-r--r-- 1 hadoop supergroup 81 2025-05-26 14:54 /bda_hadoop/moutput/part-r-00000
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -cat /bda_hadoop/moutput/part-r-00000
01 -13
02 -66
03 -15
04 43
05 100
06 168
07 219
08 198
09 141
10 100
11 1
12 -61
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$

Program 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.

Observation:



HADOOP PROGRAM - 3
Implement the top N occurrences of words.

TopN Driver Class:

```
import java.io.IOException;
import java.util.String;
import org.apache.hadoop.mapreduce.lib.input.
        FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.
        FileOutputFormat;

public class TopN {
public static void main(String [] args)
      throws Exception
{
   Configuration conf = new Configuration();
   if (otherArgs.length != 2)
   {
      System.err.println("Usage: TopN");
      System.exit(2);
   }
   Job job = job.getInstance(conf);
   Job.setJobName(TopN.class);
   job.setJarByClass(TopNMapper.class);
   job.setOutputKeyClass(Text.class);
   job.setOutputValueClass(IntWritable.class);
}
}
```

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

```
TopNMapper.class

import java.io.IOException;
import org.apache.hadoop.mapreduce.Mapper;

public class MeanMaxMapper extends Mapper
<LongWritable Text, Text, IntWritable>
{
    public void main(LongWritable key, Text
      value, Mapper<LongWritable, Text, Text,
       IntWritable>)
    {   int temperature;
        String line = value.toString();
        string month = line.substring(19,21);
        if(line.charAt(87)=='+')
        {
            temperature = Integer.parseInt(line.
                substring(88,92));
        }
```

```
        String quality = line.substring(92,93);
        context.write(new Text(month), new
            IntWritable);
    }
}
```

TopNReducer.class

```
public class TopNReducer extends Reducer
<Text, IntWritable, Text, IntWritable>
{
    private Map<Text, IntWritable>
    countMap = new HashMap<>();
    public void reduce (Text key)
    {
        int sum = 0;
        for (IntWritable val: values)
            sum += val.get();
        this.countMap.put(new Text(key),
            new IntWritable(sum));
    }

    protected void cleanup (Reducer<Text>)
    {   for( Text key . SortedMap keyset <>)
        {   if(counter ++ == 20)
                break;
            context.write(key, sootedMap.get(key));
    } } }
```

OUTPUT:

Hello
world
hadoop.

Code with Output:

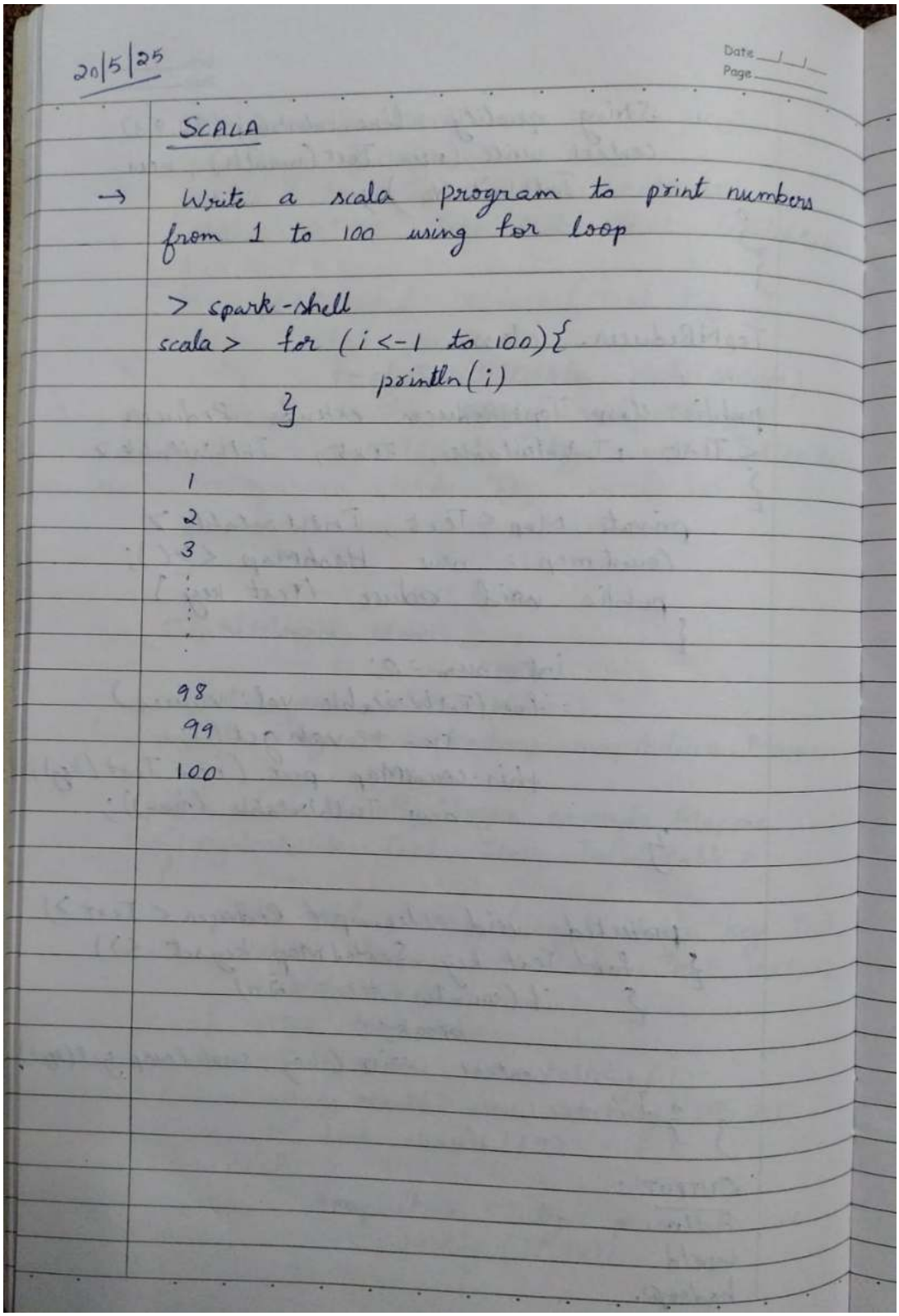
```
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ jps
12082 NameNode
19238 Jps
13014 NodeManager
15100 org.eclipse.equinox.launcher_1.6.1000.v20250227-1734.jar
12845 ResourceManager
12557 SecondaryNameNode
12255 DataNode
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -copyFromLocal /home/hadoop/Desktop/TopN.txt /bda_
haoop/tinput.txt
copyFromLocal: `/bda_haoop/tinput.txt': No such file or directory: `hdfs://localhost:9000/bda_haoop/tinput.txt'
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -copyFromLocal /home/hadoop/Desktop/TopN.txt /bda_
hadoop/tinput.txt
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hadoop jar /home/hadoop/Desktop/TopN.jar TopN.TNDriver /bda
_hadoop/tinput.txt /bda_hadoop/toutput
2025-05-26 14:59:03,334 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2025-05-26 14:59:03,372 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2025-05-26 14:59:03,372 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2025-05-26 14:59:03,426 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. I
mplement the Tool interface and execute your application with ToolRunner to remedy this.
2025-05-26 14:59:03,472 INFO input.FileInputFormat: Total input files to process : 1
2025-05-26 14:59:03,497 INFO mapreduce.JobSubmitter: number of splits:1
2025-05-26 14:59:03,554 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local1824101299_0001
2025-05-26 14:59:03,554 INFO mapreduce.JobSubmitter: Executing with tokens: []
2025-05-26 14:59:03,609 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
2025-05-26 14:59:03,610 INFO mapreduce.Job: Running job: job_local1824101299_0001
2025-05-26 14:59:03,610 INFO mapred.LocalJobRunner: OutputCommitter set in config null
2025-05-26 14:59:03,614 INFO output.PathOutputCommitterFactory: No output committer factory defined, defaulting
 to FileOutputCommitterFactory
2025-05-26 14:59:03,614 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2025-05-26 14:59:03,614 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders un
der output directory:false, ignore cleanup failures: false
2025-05-26 14:59:03,614 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.F
ileOutputCommitter
2025-05-26 14:59:03,654 INFO mapred.LocalJobRunner: Waiting for map tasks
2025-05-26 14:59:03,655 INFO mapred.LocalJobRunner: Starting task: attempt_local1824101299_0001_m_000000_0
2025-05-26 14:59:03,664 INFO output.PathOutputCommitterFactory: No output committer factory defined, defaulting
 to FileOutputCommitterFactory
2025-05-26 14:59:03,664 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2025-05-26 14:59:03,664 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders un
der output directory:false, ignore cleanup failures: false
2025-05-26 14:59:03,670 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
2025-05-26 14:59:03,672 INFO mapred.MapTask: Processing split: hdfs://localhost:9000/bda_hadoop/tinput.txt:0+95
2025-05-26 14:59:03,701 INFO mapred.MapTask: (EQUATOR) 0 kvi 26214396(104857584)
2025-05-26 14:59:03,701 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100
2025-05-26 14:59:03,701 INFO mapred.MapTask: soft limit at 83886080
2025-05-26 14:59:03,701 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600
2025-05-26 14:59:03,701 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
2025-05-26 14:59:03,702 INFO mapred.MapTask: Map output collector class = org.apache.hadoop.mapred.MapTask$MapO
utputBuffer
2025-05-26 14:59:03,738 INFO mapred.LocalJobRunner:
2025-05-26 14:59:03,739 INFO mapred.MapTask: Starting flush of map output
```

```
        File System Counters
                FILE: Number of bytes read=10682
                FILE: Number of bytes written=1291808
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=190
                HDFS: Number of bytes written=40
                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=3
                Map output records=15
                Map output bytes=154
                Map output materialized bytes=190
                Input split bytes=108
                Combine input records=0
                Combine output records=0
                Reduce input groups=5
                Reduce shuffle bytes=190
                Reduce input records=15
                Reduce output records=5
                Spilled Records=30
                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=95
        File Output Format Counters
                Bytes Written=40
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -ls /bda_hadoop/toutput
Found 2 items
-rw-r--r--   1 hadoop supergroup          0 2025-05-26 14:59 /bda_hadoop/toutput/_SUCCESS
-rw-r--r--   1 hadoop supergroup         40 2025-05-26 14:59 /bda_hadoop/toutput/part-r-00000
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ hdfs dfs -cat /bda_hadoop/toutput/part-r-00000
banana  5
apple   4
fruit   3
mango   2
kiwi    1
hadoop@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$
```

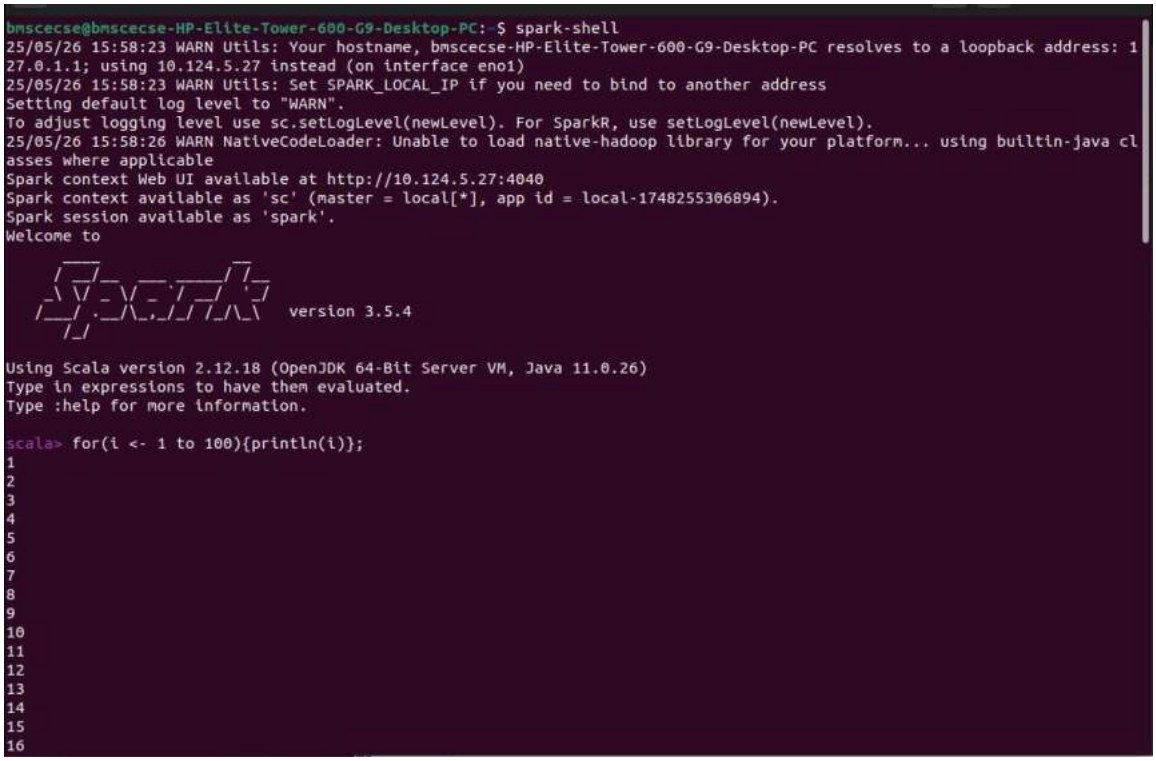
Program 8

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

Observation:



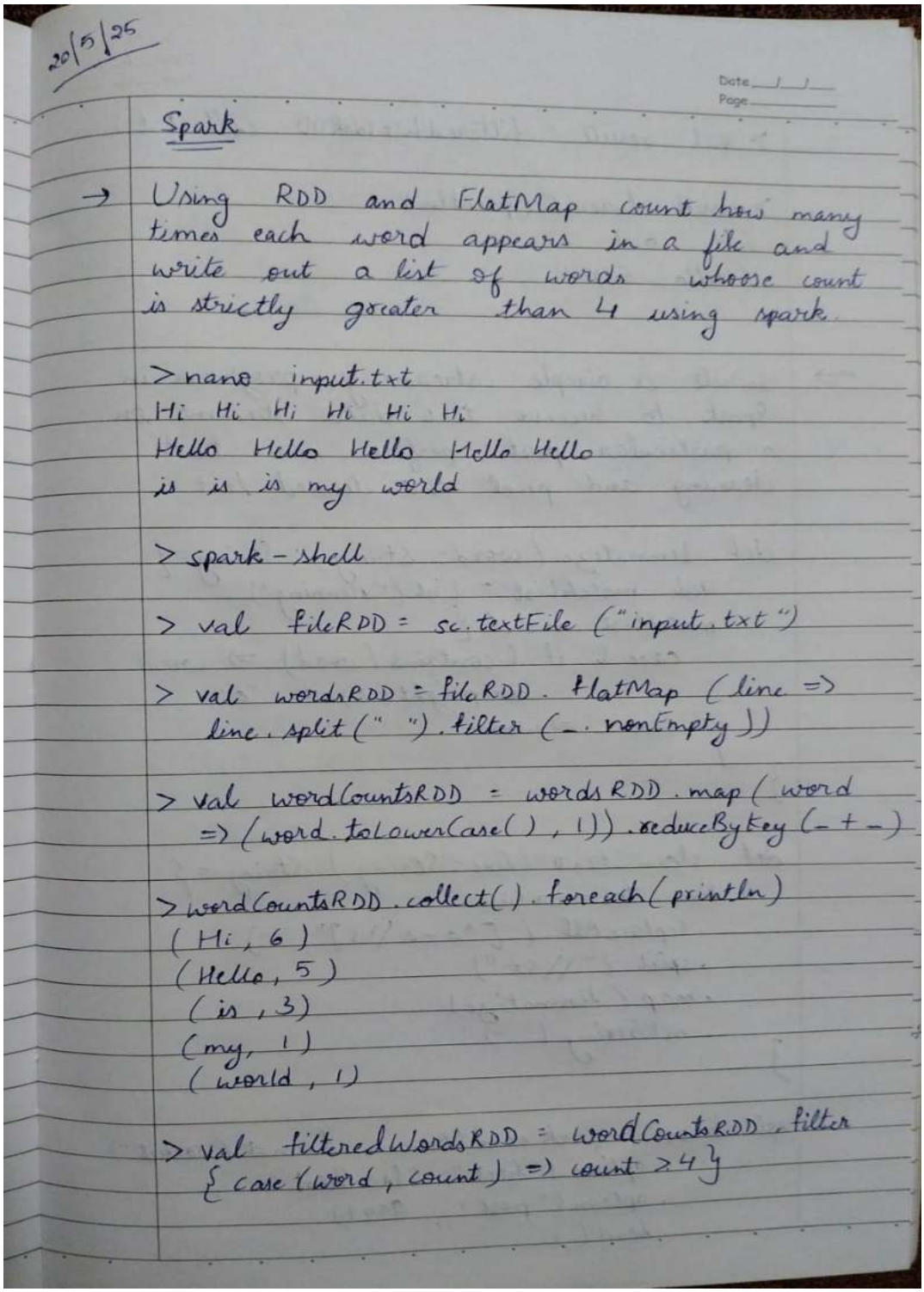
Code with Output:



Program 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

Observation:



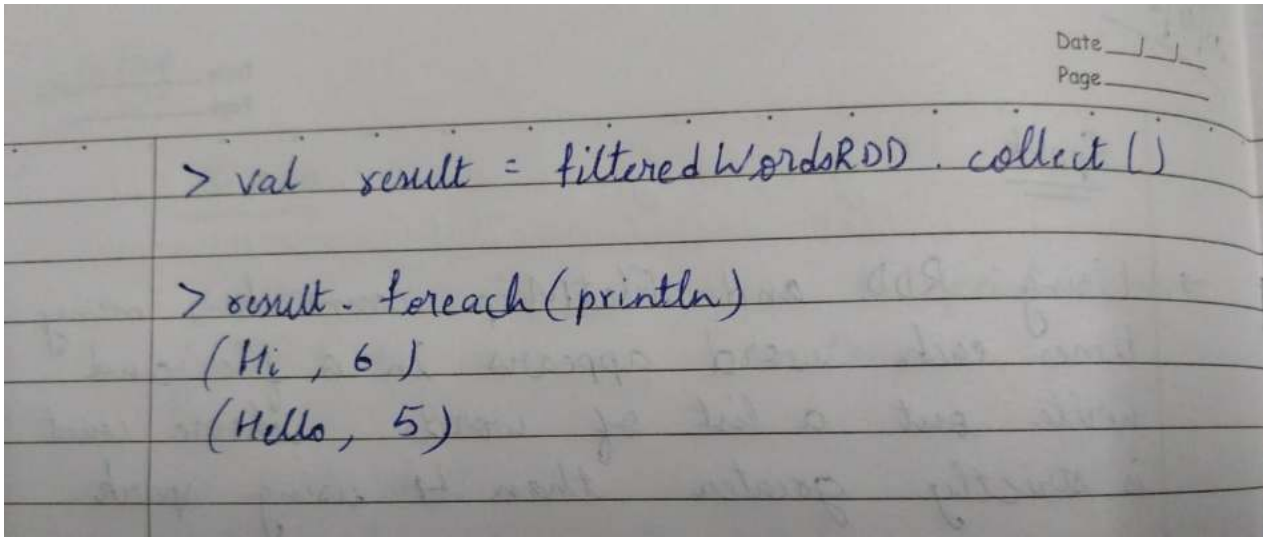
20/5/25

Date / /
Page

Spark

→ 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.

```
> nano input.txt
Hi Hi Hi Hi Hi Hi
Hello Hello Hello Hello Hello
is is is my world

> spark-shell

> val fileRDD = sc.textFile("input.txt")

> val wordsRDD = fileRDD.flatMap(line =>
  line.split(" ").filter(_.nonEmpty))

> val wordCountsRDD = wordsRDD.map(word
  => (word.toLowerCase(), 1)).reduceByKey(_+_)

> wordCountsRDD.collect().foreach(println)
(Hi, 6)
(Hello, 5)
(is, 3)
(my, 1)
(world, 1)

> val filteredWordsRDD = wordCountsRDD.filter
  {case (word, count) => count > 4}
```



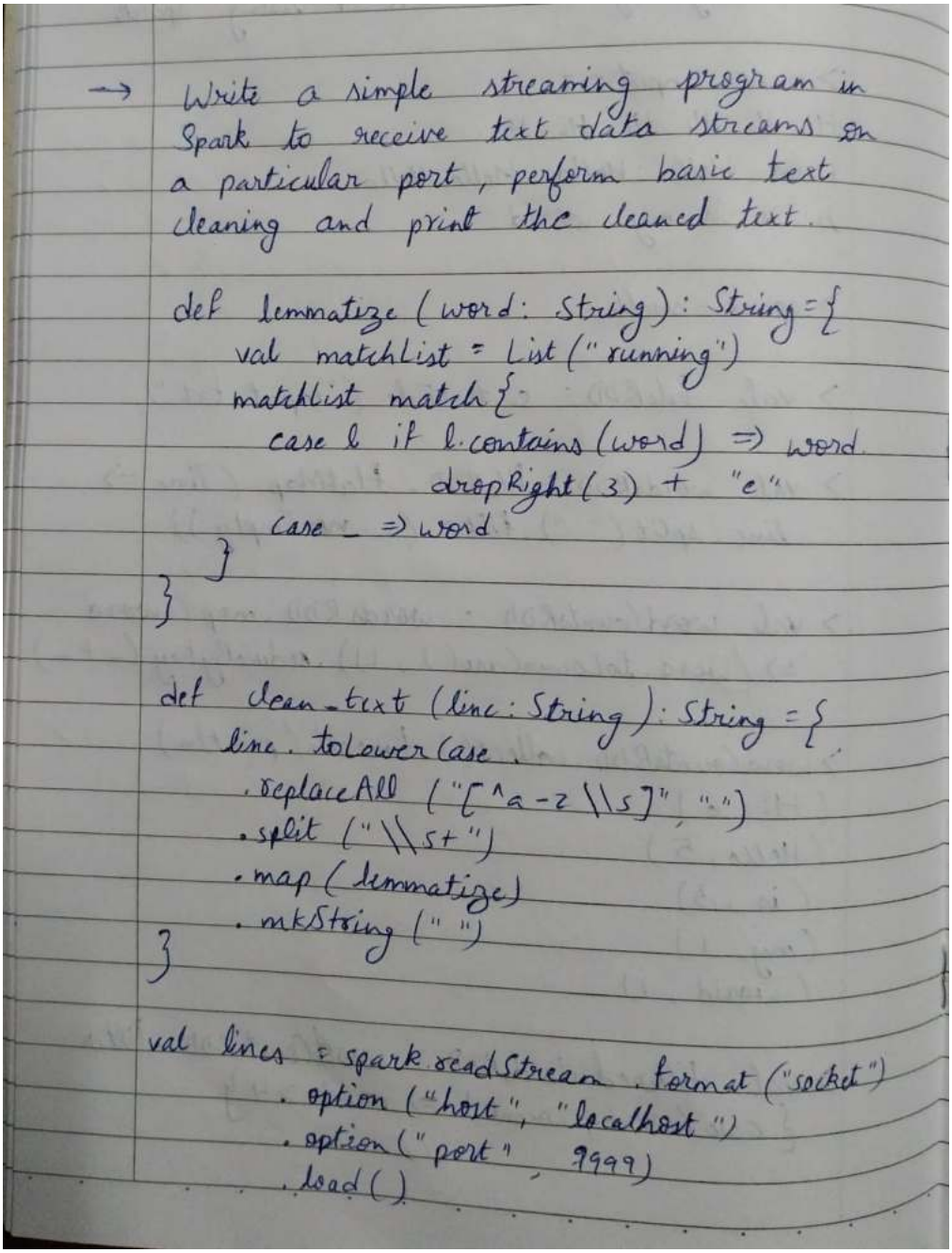
Code with Output:

```
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ echo "code code code code code spark spark spark spark spark hell
o hello hi hi joe ken">input.txt
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ spark-shell
25/05/26 16:01:15 WARN Utils: Your hostname, bmscecse-HP-Elite-Tower-600-G9-Desktop-PC resolves to a loopback address:
127.0.1.1; using 10.124.5.27 instead (on interface eno1)
25/05/26 16:01:15 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
25/05/26 16:01:17 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java c
lasses where applicable
Spark context Web UI available at http://10.124.5.27:4040
Spark context available as 'sc' (master = local[*], app id = local-1748255477930).
Spark session available as 'spark'.
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 3.5.4
      /_/

Using Scala version 2.12.18 (OpenJDK 64-Bit Server VM, Java 11.0.26)
Type in expressions to have them evaluated.
Type :help for more information.

scala> val lines=sc.textFile("input.txt")
lines: org.apache.spark.rdd.RDD[String] = input.txt MapPartitionsRDD[1] at textFile at <console>:23

scala> val words=lines.flatmap(line => line.split(" "))
<console>:23: error: value flatmap is not a member of org.apache.spark.rdd.RDD[String]
       val words=lines.flatmap(line => line.split(" "))
                       ^

scala> val words=lines.flatMap(line => line.split(" "))
words: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[2] at flatMap at <console>:23

scala> val wordParts = words.map(word => (word,1))
wordParts: org.apache.spark.rdd.RDD[(String, Int)] = MapPartitionsRDD[3] at map at <console>:23

scala> val wordcount = wordParts.reduceByKey(_+_)
wordcount: org.apache.spark.rdd.RDD[(String, Int)] = ShuffledRDD[4] at reduceByKey at <console>:23

scala> val freq = wordcount.filter {case (word,count) => count > 4}
freq: org.apache.spark.rdd.RDD[(String, Int)] = MapPartitionsRDD[5] at filter at <console>:23

scala> freq.collect().foreach(println)
(spark,5)
(code,5)

scala>
```

Program 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).

Observation:



→ Write a simple streaming program in Spark to receive text data streams on a particular port, perform basic text cleaning and print the cleaned text.

```
def lemmatize (word: String): String = {
    val matchList = List ("running")
    matchlist match {
      case l if l.contains (word) => word.
          dropRight (3) + "e"
      case _ => word
    }
}

def clean_text (line: String): String = {
    line. toLowerCase .
      .replaceAll ("[^a-z\\s]", "")
      .split ("\\s+")
      .map (lemmatize)
      .mkString (" ")
}

val lines = spark.readStream .format ("socket")
      . option ("host", "localhost")
      . option ("port", 9999)
      .load ()
```

Date __/__/__
Page _____

```
import org.apache.spark.sql.functions.udf

val cleanUDF = udf(clean_text _)

val cleaned = lines.withColumn("cleaned",
      cleanUDF("value")))

val query = cleaned.select("cleaned").
    .writeStream
    .outputMode("append")
    .format("console")
    .start()

query.awaitTermination()
```

- New Terminal

```
nc -lk 9999
Text = Hello! Spark Shell!
```

Output :

value : Hello! Spark Shell!
cleaned: Spark Shell