B.M.S. COLLEGE OF ENGINEERING BENGALURU
Autonomous Institute, Affiliated to VTU

Lab Record

Big-Data Analytics

Submitted in partial fulfillment for the 6th Semester Laboratory

Bachelor of Technology
in
Computer Science and Engineering

Submitted by:

Nitish N Banakar

1BM18CS065

Department of Computer Science and Engineering
B.M.S. College of Engineering
Bull Temple Road, Basavanagudi, Bangalore 560 019
Mar-June 2021

B.M.S. COLLEGE OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Big-Data Analytics (20CS6PEBDA) laboratory has been carried out by Nitish N Banakar(1BM18CS065) during the 6th Semester Mar-June-2021.

Signature of the Faculty Incharge:

NAME OF THE FACULTY:

Bhoomika A P
Associate Professor
Department of Computer Science and Engineering
B.M.S. College of Engineering, Bangalore

Table of Contents

Program – 1

Date – 29/03/2021

1. Create a keyspace by name Employee

```
cqlsh> create keyspace employee with replication = {'class': 'SimpleStrategy', 'replication_factor':
1};

cqlsh> use employee;
```

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

```
cqlsh:employee> create table employeeinfo(emp_id int primary key, emp_name text, designation
text, doj timestamp, salary double, dept_name text);
```

3. Insert the values into the table in batch

```
cqlsh:employee> begin batch
        ... insert into employeeinfo(emp_id, emp_name, designation, doj, salary, dept_name) values
        …(1, 'Ajay', 'Data analyst', '2018-04-16', 20000, 'Corporate');
        ... insert into employeeinfo(emp_id, emp_name, designation, doj, salary, dept_name) values
        …(121, 'Chaitra', 'web design', '2019-08-06', 15000, 'web_designer');
        ... apply batch;

cqlsh:employee> select * from employeeinfo;
```

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

```
cqlsh:employee> update employeeinfo set emp_name = 'Joy', dept_name = 'Management' where
emp_id = 121;

cqlsh:employee> select * from employeeinfo;
```

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

```
cqlsh:employee> alter table employeeinfo add projects set<text>;
```

6. Update the altered table to add project names.

```
cqlsh:employee> update employeeinfo set projects = {'project1', 'project2'} where emp_id in(1,121);

cqlsh:employee> select * from employeeinfo;
```

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

```
cqlsh:employee> begin batch
        ... insert into employeeinfo(emp_id, emp_name, designation, doj, salary, dept_name) values
        …(121, 'Boris', 'MTO', '2001-08-05', 12212, 'Corporate') using ttl 15;
        ... apply batch;

cqlsh:employee> select ttl(designation) from employeeinfo where emp_id = 121;
```

Output :

```
Terminal
Your Interactive Bash Terminal.
$ cqlsh
Connected to Test Cluster at 127.0.0.1:9042.
[cqlsh 5.0.1 | Cassandra 4.0-beta2 | CQL spec 3.4.5 | Native protocol v4]
Use HELP for help.
cqlsh>
cqlsh>
cqlsh> create keyspace employee with replication = {'class': 'SimpleStrategy', 'replication_factor': 1};
cqlsh> use employee;
cqlsh:employee> create table employeeinfo(emp_id int primary key, emp_name text, designation text, doj timestamp, salary double, dept_name text);
cqlsh:employee> begin batch
           ... insert into employeeinfo(emp_id, emp_name, designation, doj, salary, dept_name) values (1, 'Ajay', 'Data analyst', '2018-04-16', 2000
0, 'Corporate');
           ... insert into employeeinfo(emp_id, emp_name, designation, doj, salary, dept_name) values (121, 'Chaitra', 'web design', '2019-08-06', 1
5000, 'web_designer');
           ... apply batch;
cqlsh:employee> select * from employeeinfo;

 emp_id | dept_name     | designation  | doj                             | emp_name | salary
--------+---------------+--------------+---------------------------------+----------+--------
      1 |     Corporate | Data analyst | 2018-04-16 00:00:00.000000+0000 |     Ajay |  20000
    121 | web_designer  |   web design | 2019-08-06 00:00:00.000000+0000 |  Chaitra |  15000

(2 rows)
cqlsh:employee> update employeeinfo set emp_name = 'Joy', dept_name = 'Management' where emp_id = 121;
cqlsh:employee> select * from employeeinfo;

 emp_id | dept_name  | designation  | doj                             | emp_name | salary
--------+------------+--------------+---------------------------------+----------+--------
      1 |  Corporate | Data analyst | 2018-04-16 00:00:00.000000+0000 |     Ajay |  20000
    121 | Management |   web design | 2019-08-06 00:00:00.000000+0000 |      Joy |  15000

(2 rows)
cqlsh:employee> alter table employeeinfo add projects set<text>;
cqlsh:employee> update employeeinfo set projects = {'project1', 'project2'} where emp_id in(1,121);
cqlsh:employee> select * from employeeinfo;

 emp_id | dept_name  | designation  | doj                             | emp_name | projects                  | salary
--------+------------+--------------+---------------------------------+----------+---------------------------+--------
      1 |  Corporate | Data analyst | 2018-04-16 00:00:00.000000+0000 |     Ajay | {'project1', 'project2'} |  20000
    121 | Management |   web design | 2019-08-06 00:00:00.000000+0000 |      Joy | {'project1', 'project2'} |  15000

(2 rows)
```

```
Terminal
--------+------------+--------------+---------------------------------+----------+--------
      1 |  Corporate | Data analyst | 2018-04-16 00:00:00.000000+0000 |     Ajay |  20000
    121 | Management |   web design | 2019-08-06 00:00:00.000000+0000 |      Joy |  15000

(2 rows)
cqlsh:employee> alter table employeeinfo add projects set<text>;
cqlsh:employee> update employeeinfo set projects = {'project1', 'project2'} where emp_id in(1,121);
cqlsh:employee> select * from employeeinfo;

 emp_id | dept_name  | designation  | doj                             | emp_name | projects                  | salary
--------+------------+--------------+---------------------------------+----------+---------------------------+--------
      1 |  Corporate | Data analyst | 2018-04-16 00:00:00.000000+0000 |     Ajay | {'project1', 'project2'} |  20000
    121 | Management |   web design | 2019-08-06 00:00:00.000000+0000 |      Joy | {'project1', 'project2'} |  15000

(2 rows)
cqlsh:employee> begin batch
           ... insert into employeeinfo(emp_id, emp_name, designation, doj, salary, dept_name) values (121, 'Boris', 'MTO', '2001-08-05', 12212, 'Co
rporate') using ttl 15;
           ... apply batch;
cqlsh:employee> select ttl(designation) from employeeinfo where emp_id = 121;

 ttl(designation)
------------------
             null

(1 rows)
cqlsh:employee> select * from employeeinfo;

 emp_id | dept_name | designation  | doj                             | emp_name | projects                  | salary
--------+-----------+--------------+---------------------------------+----------+---------------------------+--------
      1 | Corporate | Data analyst | 2018-04-16 00:00:00.000000+0000 |     Ajay | {'project1', 'project2'} |  20000
    121 |      null |         null |                            null |     null | {'project1', 'project2'} |   null

(2 rows)
cqlsh:employee> begin batch insert into employeeinfo(emp_id, emp_name, designation, doj, salary, dept_name) values (121, 'Boris', 'MTO', '2001-08-05'
, 12212, 'Corporate') using ttl 120; apply batch;
cqlsh:employee> select ttl(designation) from employeeinfo where emp_id = 121;
                                                 ttl(designation)
------------------
              109

(1 rows)
cqlsh:employee>
```

Program – 2

Perform the following DB operations using Cassandra.

1.Create a keyspace by name Library

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

2. Create a column family by name Library-Info with attributes
Stud_Id Primary Key,
Counter_value of type Counter,
Stud_Name, Book-Name, Book-Id, Date_of_issue

cqlsh:library> create table library_info(id int, counter_val counter, stud_name text, book_name text, book_id int, issue_date timestamp,primary key(id,stud_name,book_name,book_id,issue_date));

3. Insert the values into the table in batch

cqlsh:library> update library_info SET counter_val = counter_val +1 where id = 1 and stud_name = 'Anand' and book_name = 'CNS' and book_id = 121 and issue_date='2020-12-31';

cqlsh:library> update library_info SET counter_val = counter_val +1 where id = 3 and stud_name = 'Arjun' and book_name = 'ML' and book_id = 112 and issue_date='2021-02-01';

cqlsh:library> update library_info SET counter_val = counter_val +1 where id = 5 and stud_name = 'Chaitra' and book_name = 'Python' and book_id = 114 and issue_date='2009-08-27';
cqlsh:library> select * from library_info;

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

cqlsh:library> update library_info SET counter_val = counter_val +1 where id = 3 and stud_name = 'Arjun' and book_name = 'ML' and book_id = 112 and issue_date='2021-02-01';

4. Write a query to show that a student with id 112 has taken a book “BDA” 2 times.

cqlsh:library> select * from library_info where counter_val = 2 allow filtering;

5. Export the created column to a csv file

```
cqlsh:library> copy library_info(id,counter_val,stud_name,book_name,book_id,issue_date) to
'Desktop/library_data.csv';
```

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

```
cqlsh:library> copy library_info(id,counter_val,stud_name,book_name,book_id,issue_date) from
'Desktop/library_data.csv';
```

Output :

```
Terminal
Your Interactive Bash Terminal.
$ cqlsh
Connected to Test Cluster at 127.0.0.1:9042.
[cqlsh 5.0.1 | Cassandra 4.0-beta2 | CQL spec 3.4.5 | Native protocol v4]
Use HELP for help.
cqlsh>
cqlsh>
cqlsh> create keyspace library  with replication = { 'class' : 'SimpleStrategy','replication_factor':1};
cqlsh> use library;
cqlsh:library> create table library_info( id int, counter_val counter, stud_name text, book_name text, book_id int, issue_date timestamp,primary key(
id,stud_name,book_name,book_id,issue_date));
cqlsh:library> update library_info SET counter_val = counter_val +1 where id = 1 and stud_name = 'Anand' and book_name = 'CNS' and book_id = 121 and
issue_date='2020-12-31';
cqlsh:library> update library_info SET counter_val = counter_val +1 where id = 3 and stud_name = 'Arjun' and book_name = 'ML' and book_id = 112 and i
ssue_date='2021-02-01';
cqlsh:library> update library_info SET counter_val = counter_val +1 where id = 5 and stud_name = 'Chaitra' and book_name = 'Python' and book_id = 114
 and issue_date='2009-08-27';
cqlsh:library> select * from library_info;

 id | stud_name | book_name | book_id | issue_date                      | counter_val
----+-----------+-----------+---------+---------------------------------+-------------
  5 |   Chaitra |    Python |     114 | 2009-08-27 00:00:00.000000+0000 |           1
  1 |     Anand |       CNS |     121 | 2020-12-31 00:00:00.000000+0000 |           1
  3 |     Arjun |        ML |     112 | 2021-02-01 00:00:00.000000+0000 |           1

(3 rows)
cqlsh:library> update library_info SET counter_val = counter_val +1 where id = 3 and stud_name = 'Arjun' and book_name = 'BDA' and book_id = 112 and
issue_date='2011-12-20';
cqlsh:library> select * from library_info where counter_val = 2 allow filtering;

 id | stud_name | book_name | book_id | issue_date | counter_val
----+-----------+-----------+---------+------------+-------------

(0 rows)
cqlsh:library> update library_info SET counter_val = counter_val +1 where id = 3 and stud_name = 'Arjun' and book_name = 'ML' and book_id = 112 and i
ssue_date='2011-12-20';
cqlsh:library> select * from library_info where counter_val = 2 allow filtering;
 id | stud_name | book_name | book_id | issue_date | counter_val
----+-----------+-----------+---------+------------+-------------

(0 rows)
cqlsh:library> update library_info SET counter_val = counter_val +1 where id = 3 and stud_name = 'Arjun' and book_name = 'ML' and book_id = 112 and i
ssue_date='2021-02-01';
```

```
Terminal

(3 rows)
cqlsh:library> update library_info SET counter_val = counter_val +1 where id = 3 and stud_name = 'Arjun' and book_name = 'BDA' and book_id = 112 and
issue_date='2011-12-20';
cqlsh:library> select * from library_info where counter_val = 2 allow filtering;

 id | stud_name | book_name | book_id | issue_date | counter_val
----+-----------+-----------+---------+------------+-------------

(0 rows)
cqlsh:library> update library_info SET counter_val = counter_val +1 where id = 3 and stud_name = 'Arjun' and book_name = 'ML' and book_id = 112 and i
ssue_date='2011-12-20';
cqlsh:library> select * from library_info where counter_val = 2 allow filtering;
 id | stud_name | book_name | book_id | issue_date | counter_val
----+-----------+-----------+---------+------------+-------------

(0 rows)
cqlsh:library> update library_info SET counter_val = counter_val +1 where id = 3 and stud_name = 'Arjun' and book_name = 'ML' and book_id = 112 and i
ssue_date='2021-02-01';
cqlsh:library> select * from library_info where counter_val = 2 allow filtering;
 id | stud_name | book_name | book_id | issue_date                      | counter_val
----+-----------+-----------+---------+---------------------------------+-------------
  3 |     Arjun |        ML |     112 | 2021-02-01 00:00:00.000000+0000 |           2

(1 rows)
cqlsh:library> copy library_info(id,counter_val,stud_name,book_name,book_id,issue_date) to 'Desktop/library_data.csv';
Using 1 child processes

Starting copy of library.library_info with columns [id, counter_val, stud_name, book_name, book_id, issue_date].
cqlshlib.copyutil.ExportProcess.write_rows_to_csv(): writing row
cqlshlib.copyutil.ExportProcess.write_rows_to_csv(): writing row
cqlshlib.copyutil.ExportProcess.write_rows_to_csv(): writing row
cqlshlib.copyutil.ExportProcess.write_rows_to_csv(): writing row
cqlshlib.copyutil.ExportProcess.write_rows_to_csv(): writing row
Processed: 5 rows; Rate:       8 rows/s; Avg. rate:       9 rows/s
5 rows exported to 1 files in 0.555 seconds.
cqlsh:library> copy library_info(id,counter_val,stud_name,book_name,book_id,issue_date) from 'Desktop/library_data.csv';
Using 1 child processes

Starting copy of library.library_info with columns [id, counter_val, stud_name, book_name, book_id, issue_date].
Processed: 5 rows; Rate:       9 rows/s; Avg. rate:      14 rows/s
5 rows imported from 1 files in 0.365 seconds (0 skipped).
cqlsh:library>
```

Program – 3

Perform the following DB operations using MongoDB.

1. Create a database “Student” with the following attributes Rollno, Age, ContactNo, Email-Id.
use student

2. Insert appropriate values

db.student.insert({Roll: 10, Name: "suma", age: 21, contact: "7723112389", email: "suma@gmail.com"})
db.student.insert({Roll: 11, Name: "ABC", age: 20, contact: "9263532389", email: "abc@gmail.com"})
db.student.insert({Roll: 12, Name: "shek", age: 21, contact: "7788996655", email: "shek@gmail.com"})
db.student.insert({Roll: 13, Name: "raj", age: 20, contact: "1234123412", email: "raj@gmail.com"})

3. Write a query to update Email-Id of a student with rollno 10.

db.student.update({Roll:10}, {$set: {email: "suma123@gmail.com"}})

4. Replace the student name from “ABC” to “FEM” of rollno 11.

db.student.update({Roll:11}, {$set: {Name: "FEM"}})

5. Export the created table into local file system

mongoexport --db student --collection student --type csv --out D:\export.csv --fields “Roll,Name,age,contact,email”

6. Drop the table

db.student.drop()

7. Import a given csv dataset from the local file system into mongodb collection.

mongoimport --db student --collection student --type csv --file D:\export.csv --headerline

Output :

BDA_Lab3.js | * show dbs | db.getCollection("Student... | * mongoexport --db=ne... | * student

New Connection | localhost:27017 | new_bda

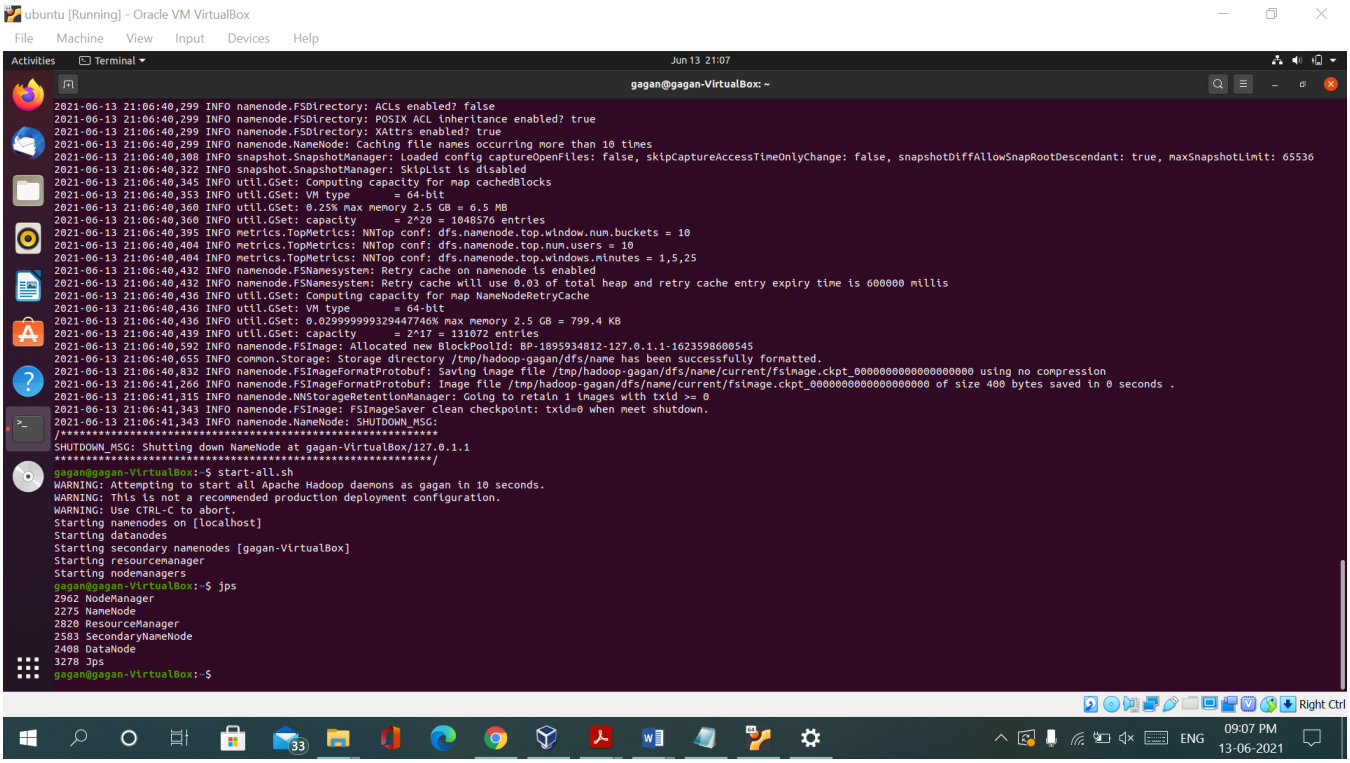
```
db.getCollection('Student').find({})
```

Student | 0.003 sec. | 0 | 50

Key	Value	Type
ContactNo	9380692117	String
Emailid	nitishb.cs18@bmsce.ac.in	String
Name	Nitish	String
(2) ObjectId("606abb6b9dbcbf27f3d98af7")	{ 6 fields }	Object
_id	ObjectId("606abb6b9dbcbf27f3d98af7")	ObjectId
Rollno	061	String
Age	20.0	Double
ContactNo	9898989898	String
Emailid	nikhil.cs18@bmsce.ac.in	String
Name	Nikhil	String
(3) ObjectId("606abb6b9dbcbf27f3d98af8")	{ 6 fields }	Object
_id	ObjectId("606abb6b9dbcbf27f3d98af8")	ObjectId
Rollno	071	String
Age	20.0	Double
ContactNo	9898979778	String
Emailid	pranav.cs18@bmsce.ac.in	String
Name	Pranav	String
(4) ObjectId("606abbad9dbcbf27f3d98af9")	{ 6 fields }	Object
_id	ObjectId("606abbad9dbcbf27f3d98af9")	ObjectId
Rollno	010	String
Age	20.0	Double
ContactNo	9380696544	String
Emailid	akash@gmail.com	String
Name	Akshay	String
(5) ObjectId("606ad80d9dbcbf27f3d98afa")	{ 6 fields }	Object
_id	ObjectId("606ad80d9dbcbf27f3d98afa")	ObjectId
Name	FEM	String
Rollno	011	String
Age	20.0	Double
ContactNo	9198989898	String
Emailid	abc.cs18@bmsce.ac.in	String
(6) ObjectId("606ad80d9dbcbf27f3d98afb")	{ 6 fields }	Object
_id	ObjectId("606ad80d9dbcbf27f3d98afb")	ObjectId
Name	XYX	String
Rollno	100	String
Age	20.0	Double
ContactNo	9198979778	String
Emailid	xyx.cs18@bmsce.ac.in	String

Program – 4

Screenshot of Hadoop installation :



Program – 5

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

nitsh@Nitish:/usr/local/hadoop/bin$ sudo su hduser

[sudo] password for nitsh:

hduser@Nitish:/usr/local/hadoop/bin$ hadoop version

Hadoop 2.10.1

Subversion https://github.com/apache/hadoop -r 1827467c9a56f133025f28557bfc2c562d78e816

Compiled by centos on 2020-09-14T13:17Z

Compiled with protoc 2.5.0

From source with checksum 3114edef868f1f3824e7d0f68be03650

This command was run using /usr/local/hadoop/share/hadoop/common/hadoop-common-2.10.1.jar

hduser@Nitish:/usr/local/hadoop/bin$ cd ~

hduser@Nitish:~$ start-all.sh

hduser@Nitish:~$ jps

19521 Jps

17825 DataNode

18275 ResourceManager

18085 SecondaryNameNode

17607 NameNode

18446 NodeManager

1)

```
hduser@Nitish:~$ hadoop fs -mkdir /newDataFlair
```

2)

```
hduser@Nitish:~$ hadoop fs -ls /
Found 1 items
drwxr-xr-x   - hduser supergroup          0 2021-04-20 15:24 /newDataFlair
```

3)

```
hduser@Nitish:~$ hdfs dfs -copyFromLocal ~/temp.txt /newDataFlair
hduser@Nitish:~$ hdfs dfs -ls /newDataFlair
Found 1 items
-rw-r--r--   1 hduser supergroup         18 2021-04-20 20:59 /newDataFlair/temp.txt
```

4)

```
hduser@Nitish:~$ hadoop fs -count -q /newDataFlair
        none             inf            none             inf            1            1                18 /newDataFlair
```

5)

```
hduser@Nitish:~$ hdfs dfs -cat /newDataFlair/temp.txt
Nitish N Banakar
```

6)

```
hduser@Nitish:~$ hadoop fs -appendToFile ~/nitish.txt /newDataFlair/temp.txt
hduser@Nitish:~$ hdfs dfs -cat /newDataFlair/temp.txt
Nitish N Banakar
```

1BM18CS065

7)

```
hduser@Nitish:~$ hdfs dfs -mkdir /sample
hduser@Nitish:~$ hdfs dfs -cp /newDataFlair/temp.txt /sample/copyfile
hduser@Nitish:~$ hdfs dfs -cat /sample/copyfile
Nitish N Banakar

1BM18CS065
```

8)

```
hduser@Nitish:~$ hadoop fs -du -h -x /sample/copyfile
33  /sample/copyfile
```

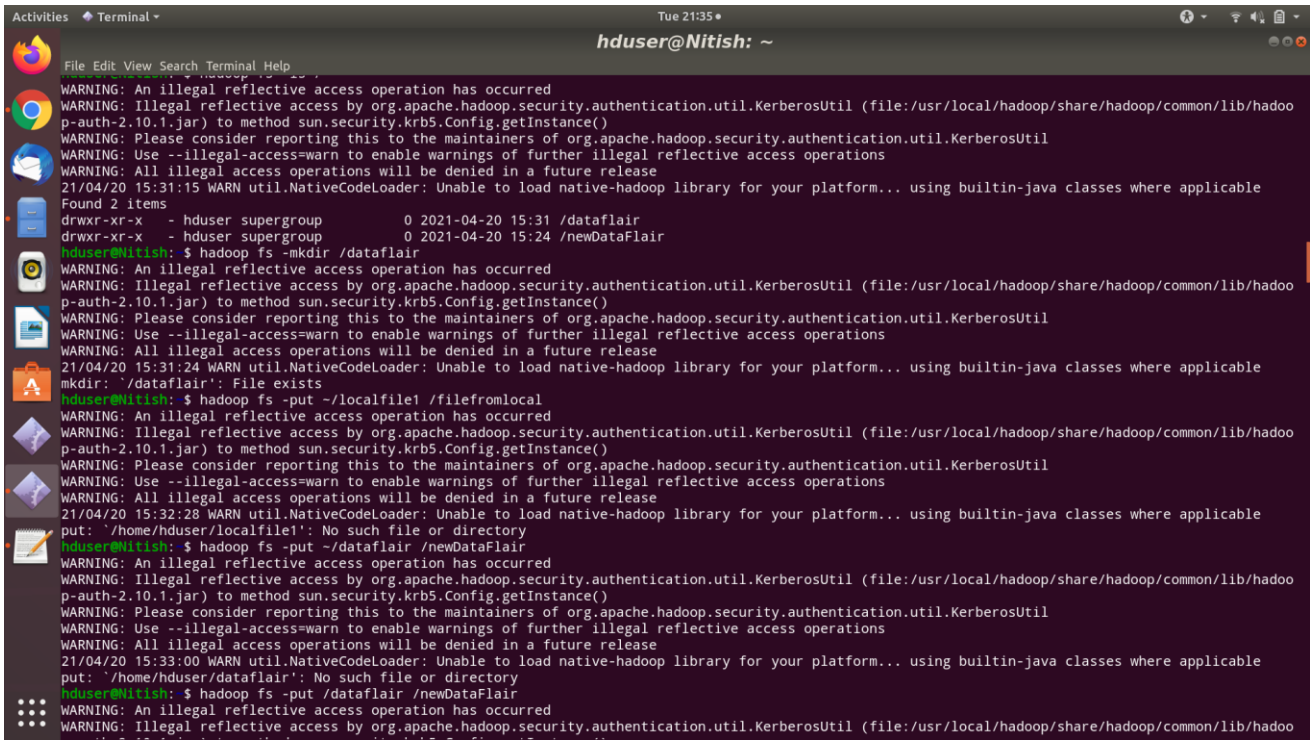
9)

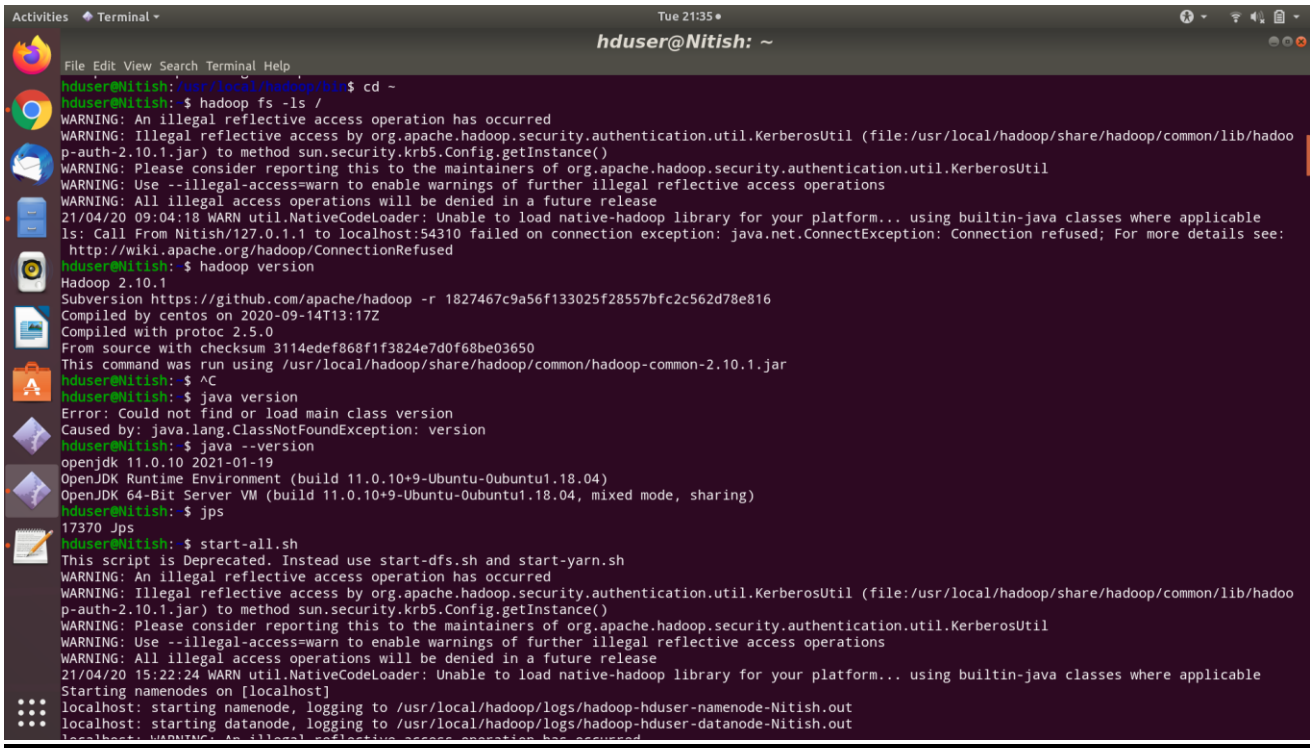
```
hduser@Nitish:~$ hadoop fs -mkdir /dataflair
hduser@Nitish:~$ hadoop fs -mv /newDataFlair/temp.txt /dataflair
hduser@Nitish:~$ hadoop fs -ls /dataflair
Found 1 items
-rw-r--r--   1 hduser supergroup         33 2021-04-20 21:13 /dataflair/temp.txt
```

10)

```
hduser@Nitish:~$ hadoop fs -rm /sample/copyfile
Deleted /sample/copyfile
hduser@Nitish:~$ hadoop fs -rm -R /newDataFlair
Deleted /newDataFlair
hduser@Nitish:~$ hadoop fs -ls /
Found 2 items
drwxr-xr-x   - hduser supergroup          0 2021-04-20 21:25 /dataflair
drwxr-xr-x   - hduser supergroup          0 2021-04-20 21:27 /sample
```

Screenshot


Activities
Terminal
Tue 21:35
hduser@Nitish: ~
File Edit View Search Terminal Help
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/04/20 15:31:15 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 2 items
drwxr-xr-x - hduser supergroup 0 2021-04-20 15:31 /dataflair
drwxr-xr-x - hduser supergroup 0 2021-04-20 15:24 /newDataFlair
hduser@Nitish: ~$ hadoop fs -mkdir /dataflair
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/04/20 15:31:24 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
mkdir: `/dataflair': File exists
hduser@Nitish: ~$ hadoop fs -put ~/localfile1 /filefromlocal
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/04/20 15:32:28 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
put: `/home/hduser/localfile1': No such file or directory
hduser@Nitish: ~$ hadoop fs -put ~/dataflair /newDataFlair
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/04/20 15:33:00 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
put: `/home/hduser/dataflair': No such file or directory
hduser@Nitish: ~$ hadoop fs -put /dataflair /newDataFlair
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop


Activities
Terminal
Tue 21:35
hduser@Nitish: ~
File Edit View Search Terminal Help
hduser@Nitish:/usr/local/hadoop/bin$ cd ~
hduser@Nitish: ~$ hadoop fs -ls /
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/04/20 09:04:18 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
ls: Call From Nitish/127.0.1.1 to localhost:54310 failed on connection exception: java.net.ConnectException: Connection refused; For more details see:
http://wiki.apache.org/hadoop/ConnectionRefused
hduser@Nitish: ~$ hadoop version
Hadoop 2.10.1
Subversion https://github.com/apache/hadoop -r 1827467c9a56f133025f28557bfc2c562d78e816
Compiled by centos on 2020-09-14T13:17Z
Compiled with protoc 2.5.0
From source with checksum 3114edef868f1f3824e7d0f68be03650
This command was run using /usr/local/hadoop/share/hadoop/common/hadoop-common-2.10.1.jar
hduser@Nitish: ~$ ^C
hduser@Nitish: ~$ java version
Error: Could not find or load main class version
Caused by: java.lang.ClassNotFoundException: version
hduser@Nitish: ~$ java --version
openjdk 11.0.10 2021-01-19
OpenJDK Runtime Environment (build 11.0.10+9-Ubuntu-0ubuntu1.18.04)
OpenJDK 64-Bit Server VM (build 11.0.10+9-Ubuntu-0ubuntu1.18.04, mixed mode, sharing)
hduser@Nitish: ~$ jps
17370 Jps
hduser@Nitish: ~$ start-all.sh
This script is Deprecated. Instead use start-dfs.sh and start-yarn.sh
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/04/20 15:22:24 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Starting namenodes on [localhost]
localhost: starting namenode, logging to /usr/local/hadoop/logs/hadoop-hduser-namenode-Nitish.out
localhost: starting datanode, logging to /usr/local/hadoop/logs/hadoop-hduser-datanode-Nitish.out

```
Activities  Terminal                    Tue 21:36
                    hduser@Nitish: ~
File  Edit  View  Search  Terminal  Help
p-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/04/20 21:08:44 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
        none             inf            none             inf            1            1                 18 /newDataFlair
hduser@Nitish:~$ hdfs dfs -cat /newDataFlair/temp.txt
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoo
p-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/04/20 21:11:14 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

Nitish N Banakar
hduser@Nitish:~$ vi nitish.txt
hduser@Nitish:~$ hadoop fs -appendToFile ~/nitish.txt /newDataFlair/temp.txt
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoo
p-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/04/20 21:13:20 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hduser@Nitish:~$ hdfs dfs -cat /newDataFlair/temp.txt
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoo
p-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/04/20 21:14:05 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

Nitish N Banakar



1BM18CS065
hduser@Nitish:~$ ^C
```

Program – 6

Create a Map Reduce program to

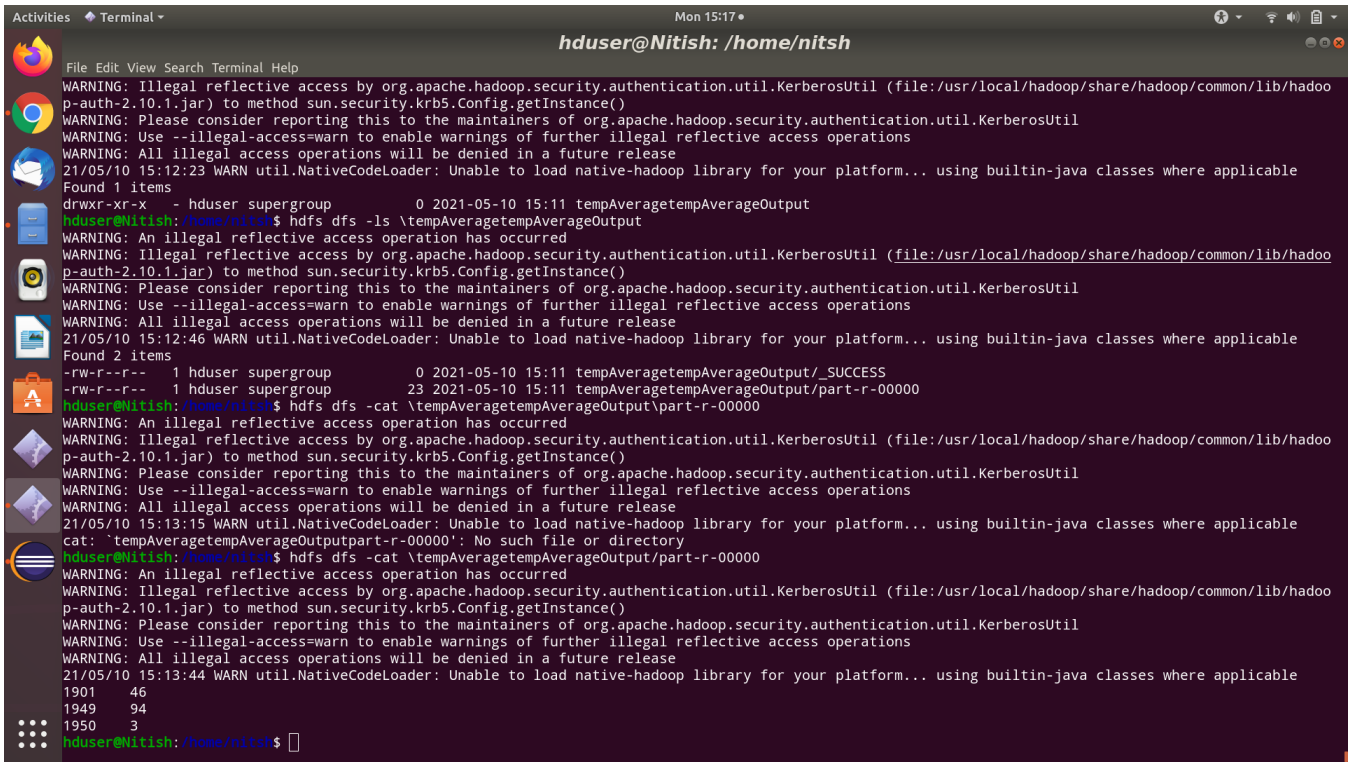
a) Find the average temperature for each year from the NCDC data set.

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

```
        }
}
//AverageMapper.java
package temperature;
import org.apache.hadoop.io.*;
import org.apache.hadoop.mapreduce.*;
import java.io.IOException;
public class AverageMapper extends Mapper <LongWritable, Text, Text, IntWritable>
{
    public static final int MISSING = 9999;
    public void map(LongWritable key, Text value, Context context) throws IOException,
    InterruptedException
    {
        String line = value.toString();
        String year = line.substring(15,19);
        int temperature;
        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 != MISSING && quality.matches("[01459]"))
        context.write(new Text(year),new IntWritable(temperature));
    }
}
```

```
//AverageReducer.java
package temperature;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.*;
import java.io.IOException;
public class AverageReducer extends Reducer <Text, IntWritable,Text, IntWritable>
{
        public void reduce(Text key, Iterable<IntWritable> values, Context context) throws
        IOException,InterruptedException
        {
            int max_temp = 0;
            int count = 0;
            for (IntWritable value : values)
            {
                max_temp += value.get();
                count+=1;
            }
            context.write(key, new IntWritable(max_temp/count));
        }
}
```

Screenshot:



```
Activities  Terminal                Mon 15:17
hduser@Nitish: /home/nitsh
File  Edit  View  Search  Terminal  Help
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoo
p-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/05/10 15:12:23 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 1 items
drwxr-xr-x   - hduser supergroup          0 2021-05-10 15:11 tempAveragetempAverageOutput
hduser@Nitish:/home/nitsh$ hdfs dfs -ls \tempAveragetempAverageOutput
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoo
p-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/05/10 15:12:46 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 2 items
-rw-r--r--   1 hduser supergroup          0 2021-05-10 15:11 tempAveragetempAverageOutput/_SUCCESS
-rw-r--r--   1 hduser supergroup         23 2021-05-10 15:11 tempAveragetempAverageOutput/part-r-00000
hduser@Nitish:/home/nitsh$ hdfs dfs -cat \tempAveragetempAverageOutput\part-r-00000
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoo
p-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/05/10 15:13:15 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
cat: `tempAveragetempAverageOutputpart-r-00000': No such file or directory
hduser@Nitish:/home/nitsh$ hdfs dfs -cat \tempAveragetempAverageOutput/part-r-00000
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoo
p-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/05/10 15:13:44 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
1901    46
1949    94
1950    3
hduser@Nitish:/home/nitsh$
```

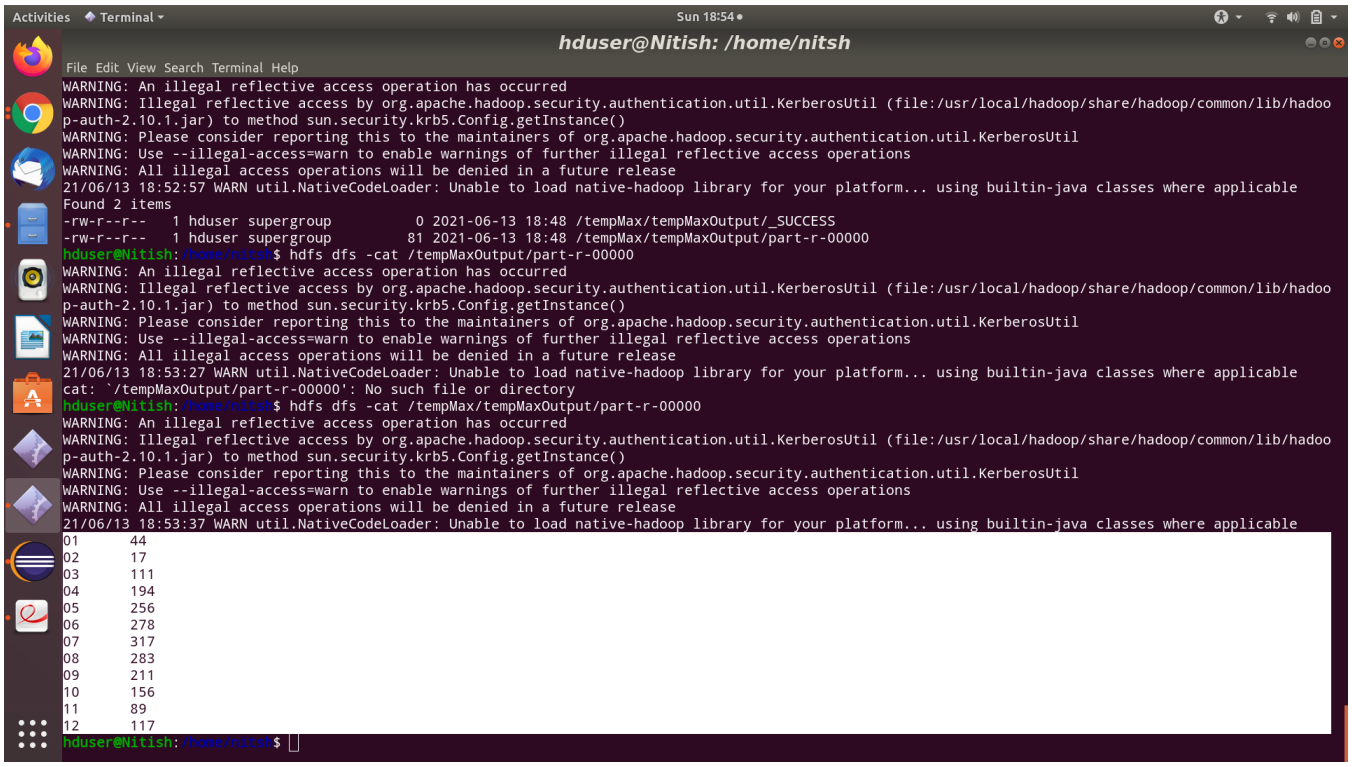
b) Find the mean max temperature for every month.

```
//TempDriver.java
package temperatureMax;
import org.apache.hadoop.io.*;
import org.apache.hadoop.fs.*;
import org.apache.hadoop.mapreduce.*;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
public class TempDriver
{
        public static void main (String[] args) throws Exception
        {
            if (args.length != 2)
            {
            System.err.println("Please Enter the input and output parameters");
            System.exit(-1);
            }
            Job job = new Job();
            job.setJarByClass(TempDriver.class);
            job.setJobName("Max temperature");
            FileInputFormat.addInputPath(job,new Path(args[0]));
            FileOutputFormat.setOutputPath(job,new Path (args[1]));
            job.setMapperClass(TempMapper.class);
            job.setReducerClass(TempReducer.class);
            job.setOutputKeyClass(Text.class);
            job.setOutputValueClass(IntWritable.class);
```

```
            System.exit(job.waitForCompletion(true)?0:1);
        }
}
//TempMapper.java
package temperatureMax;
import org.apache.hadoop.io.*;
import org.apache.hadoop.mapreduce.*;
import java.io.IOException;
public class TempMapper extends Mapper <LongWritable, Text, Text, IntWritable>
{
    public static final int MISSING = 9999;
    public void map(LongWritable key, Text value, Context context) throws
    IOException,
    InterruptedException
    {
        String line = value.toString();
        String month = line.substring(19,21);
        int temperature;
        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 != MISSING && quality.matches("[01459]"))
        context.write(new Text(month),new IntWritable(temperature));
    }
```

```
}
//TempReducer.java
package temperatureMax;
import org.apache.hadoop.io.*;
import org.apache.hadoop.mapreduce.*;
import java.io.IOException;
public class TempMapper extends Mapper <LongWritable, Text, Text, IntWritable>
{
        public static final int MISSING = 9999;
        public void map(LongWritable key, Text value, Context context) throws
        IOException,
        InterruptedException
        {
            String line = value.toString();
            String month = line.substring(19,21);
            int temperature;
            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 != MISSING && quality.matches("[01459]"))
            context.write(new Text(month),new IntWritable(temperature));
        }
}
```

Screenshot:


Activities
Terminal
Sun 18:54
hduser@Nitish: /home/nitsh
File Edit View Search Terminal Help
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/06/13 18:52:57 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 2 items
-rw-r--r-- 1 hduser supergroup 0 2021-06-13 18:48 /tempMax/tempMaxOutput/_SUCCESS
-rw-r--r-- 1 hduser supergroup 81 2021-06-13 18:48 /tempMax/tempMaxOutput/part-r-00000
hduser@Nitish:/home/nitsh$ hdfs dfs -cat /tempMaxOutput/part-r-00000
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/06/13 18:53:27 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
cat: `/tempMaxOutput/part-r-00000': No such file or directory
hduser@Nitish:/home/nitsh$ hdfs dfs -cat /tempMax/tempMaxOutput/part-r-00000
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.10.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
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
21/06/13 18:53:37 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
01 44
02 17
03 111
04 194
05 256
06 278
07 317
08 283
09 211
10 156
11 89
12 117
hduser@Nitish:/home/nitsh$

Program – 7

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

```
// TopN.java

package sortWords;

import org.apache.hadoop.conf.Configuration;

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.io.IntWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapreduce.Job;

import org.apache.hadoop.mapreduce.Mapper;

import org.apache.hadoop.mapreduce.Reducer;

import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;

import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

import org.apache.hadoop.util.GenericOptionsParser;

import utils.MiscUtils;

import java.io.IOException;

import java.util.*;

public class TopN {

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

            Configuration conf = new Configuration();

            String[] otherArgs = new GenericOptionsParser(conf,
            args).getRemainingArgs();

            if (otherArgs.length != 2) {

                System.err.println("Usage: TopN <in> <out>");

                System.exit(2);
```

```
        }
    Job job = Job.getInstance(conf);
    job.setJobName("Top N");
    job.setJarByClass(TopN.class);
    job.setMapperClass(TopNMapper.class);
    //job.setCombinerClass(TopNReducer.class);
    job.setReducerClass(TopNReducer.class);
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(IntWritable.class);
    FileInputFormat.addInputPath(job, new Path(otherArgs[0]));
    FileOutputFormat.setOutputPath(job, new Path(otherArgs[1]));
    System.exit(job.waitForCompletion(true) ? 0 : 1);
}
/**
* The mapper reads one line at the time, splits it into an array of single words and emits every
* word to the reducers with the value of 1.
*/
public static class TopNMapper extends Mapper<Object, Text, Text, IntWritable> {
private final static IntWritable one = new IntWritable(1);
private Text word = new Text();
private String tokens = "[_|$#<>\\^=\\[\\]\\*/\\\\,;,.\\-:()?!\"']";
@Override
public void map(Object key, Text value, Context context) throws IOException,
InterruptedException {
        String cleanLine = value.toString().toLowerCase().replaceAll(tokens, " ");
```

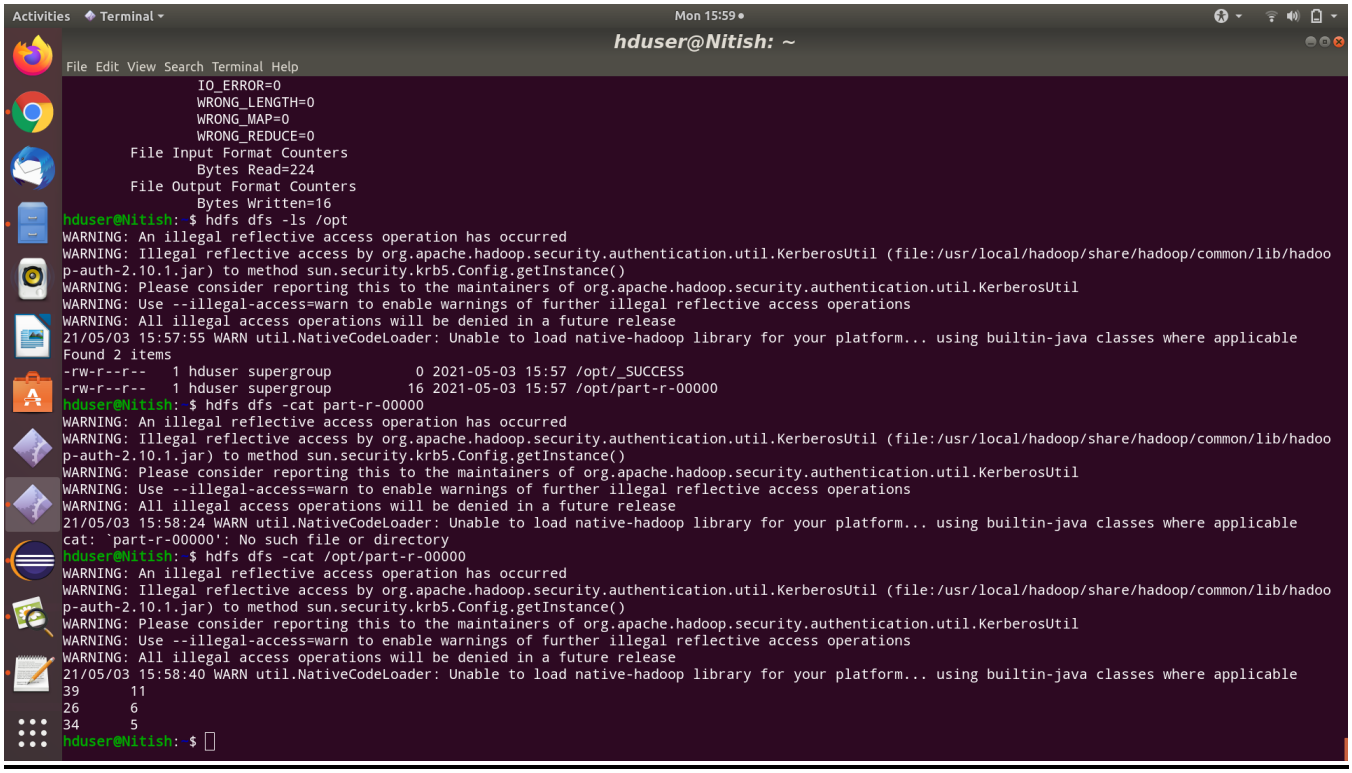
```
            StringTokenizer itr = new StringTokenizer(cleanLine);
            while (itr.hasMoreTokens()) {
            word.set(itr.nextToken().trim());
            context.write(word, one);
            }
            }
}
/**
* The reducer retrieves every word and puts it into a Map: if the word already exists in the
* map, increments its value, otherwise sets it to 1.
*/
public static class TopNReducer extends Reducer<Text, IntWritable, Text, IntWritable> {
private Map<Text, IntWritable> countMap = new HashMap<>();
@Override
public void reduce(Text key, Iterable<IntWritable> values, Context context) throws
IOException, InterruptedException {
        // computes the number of occurrences of a single word
        int sum = 0;
        for (IntWritable val : values) {
        sum += val.get();
}
// puts the number of occurrences of this word into the map.
// We need to create another Text object because the Text instance
// we receive is the same for all the words
```

```
            countMap.put(new Text(key), new IntWritable(sum));
}
@Override
protected void cleanup(Context context) throws IOException, InterruptedException {
Map<Text, IntWritable> sortedMap = MiscUtils.sortByValues(countMap);
int counter = 0;
            for (Text key : sortedMap.keySet()) {
            if (counter++ == 3) {
                break;
            }
            context.write(key, sortedMap.get(key));
            }
}
}
/**
* The combiner retrieves every word and puts it into a Map: if the word already exists in the
* map, increments its value, otherwise sets it to 1.
*/
public static class TopNCombiner extends Reducer<Text, IntWritable, Text, IntWritable> {
@Override
        public void reduce(Text key, Iterable<IntWritable> values, Context context) throws
        IOException, InterruptedException {
        // computes the number of occurrences of a single word
        int sum = 0;
```

```
        for (IntWritable val : values) {
            sum += val.get();
        }
        context.write(key, new IntWritable(sum));
        }
}
}
// MiscUtils.java
package utils;
import java.util.*;
public class MiscUtils {
/**
* sorts the map by values. Taken from:
* http://javarevisited.blogspot.it/2012/12/how-to-sort-hashmap-java-by-key-and-value.html
*/
public static <K extends Comparable, V extends Comparable> Map<K, V>
sortByValues(Map<K, V> map) {
List<Map.Entry<K, V>> entries = new LinkedList<Map.Entry<K, V>>(map.entrySet());
Collections.sort(entries, new Comparator<Map.Entry<K, V>>() {
            @Override
            public int compare(Map.Entry<K, V> o1, Map.Entry<K, V> o2) {
            return o2.getValue().compareTo(o1.getValue());
            }
});
//LinkedHashMap will keep the keys in the order they are inserted
```

```
            //which is currently sorted on natural ordering
            Map<K, V> sortedMap = new LinkedHashMap<K, V>();
            for (Map.Entry<K, V> entry : entries) {
            sortedMap.put(entry.getKey(), entry.getValue());
        }
        return sortedMap;
}
}
```

Screenshot:



Program – 8

Create a Map Reduce program to demonstrating join operation :

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



public class JoinDriver extends Configured implements Tool {

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

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

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

            if (args.length != 3) {
                    System.out.println("Usage: <Department Emp Strength input> <Department Name
input> <output>");
                    return -1;
            }

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

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

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

        MultipleInputs.addInputPath(conf, AInputPath, TextInputFormat.class, Posts.class);
        MultipleInputs.addInputPath(conf, BInputPath, TextInputFormat.class, User.class);

        FileOutputFormat.setOutputPath(conf, outputPath);

        conf.setPartitionerClass(KeyPartitioner.class);
        conf.setOutputValueGroupingComparator(TextPair.FirstComparator.class);

        conf.setMapOutputKeyClass(TextPair.class);

        conf.setReducerClass(JoinReducer.class);

        conf.setOutputKeyClass(Text.class);

        JobClient.runJob(conf);

        return 0;
    }
    public static void main(String[] args) throws Exception {

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

```
// JoinReducer.java
import java.io.IOException;
import java.util.Iterator;

import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.*;

public class JoinReducer extends MapReduceBase implements Reducer<TextPair, Text, Text, Text> {

        public void reduce (TextPair key, Iterator<Text> values, OutputCollector<Text, Text>
output, Reporter reporter)
            throws IOException
        {

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

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

import org.apache.hadoop.io.IntWritable;

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

        public void map(LongWritable key, Text value, OutputCollector<TextPair, Text> output,
Reporter reporter)
                throws IOException
        {

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

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

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

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

        public void map(LongWritable key, Text value, OutputCollector<TextPair, Text> output,
Reporter reporter)
                throws IOException
        {
            String valueString = value.toString();
            String[] SingleNodeData = valueString.split("\t");
            output.collect(new TextPair(SingleNodeData[3], "0"), new Text(SingleNodeData[9]));
        }
}

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

import org.apache.hadoop.io.*;

public class TextPair implements WritableComparable<TextPair> {

  private Text first;
  private Text second;

  public TextPair() {
```

```
  set(new Text(), new Text());
}

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

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

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

public Text getFirst() {
  return first;
}

public Text getSecond() {
  return second;
}

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

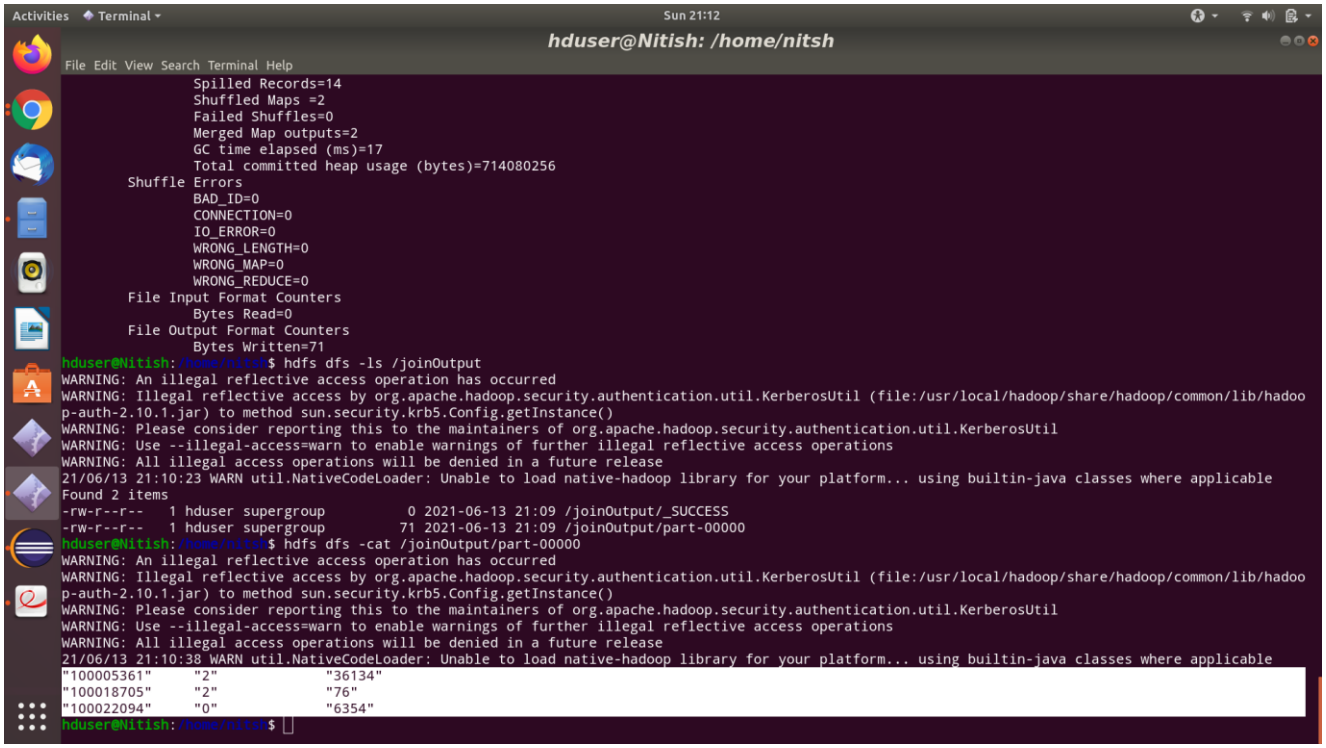
```
public void readFields(DataInput in) throws IOException {
  first.readFields(in);
  second.readFields(in);
}

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

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

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

public int compareTo(TextPair tp) {
  int cmp = first.compareTo(tp.first);
  if (cmp != 0) {
```

```
    return cmp;
  }
  return second.compareTo(tp.second);
}
// ^^ TextPair

// vv TextPairComparator
public static class Comparator extends WritableComparator {

  private static final Text.Comparator TEXT_COMPARATOR = new Text.Comparator();

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

  @Override
  public int compare(byte[] b1, int s1, int l1,
                     byte[] b2, int s2, int l2) {

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

```
    } catch (IOException e) {
      throw new IllegalArgumentException(e);
    }
  }
}

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

// vv TextPairFirstComparator
public static class FirstComparator extends WritableComparator {

  private static final Text.Comparator TEXT_COMPARATOR = new Text.Comparator();

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

  @Override
  public int compare(byte[] b1, int s1, int l1,
                     byte[] b2, int s2, int l2) {

    try {
      int firstL1 = WritableUtils.decodeVIntSize(b1[s1]) + readVInt(b1, s1);
      int firstL2 = WritableUtils.decodeVIntSize(b2[s2]) + readVInt(b2, s2);
      return TEXT_COMPARATOR.compare(b1, s1, firstL1, b2, s2, firstL2);
```

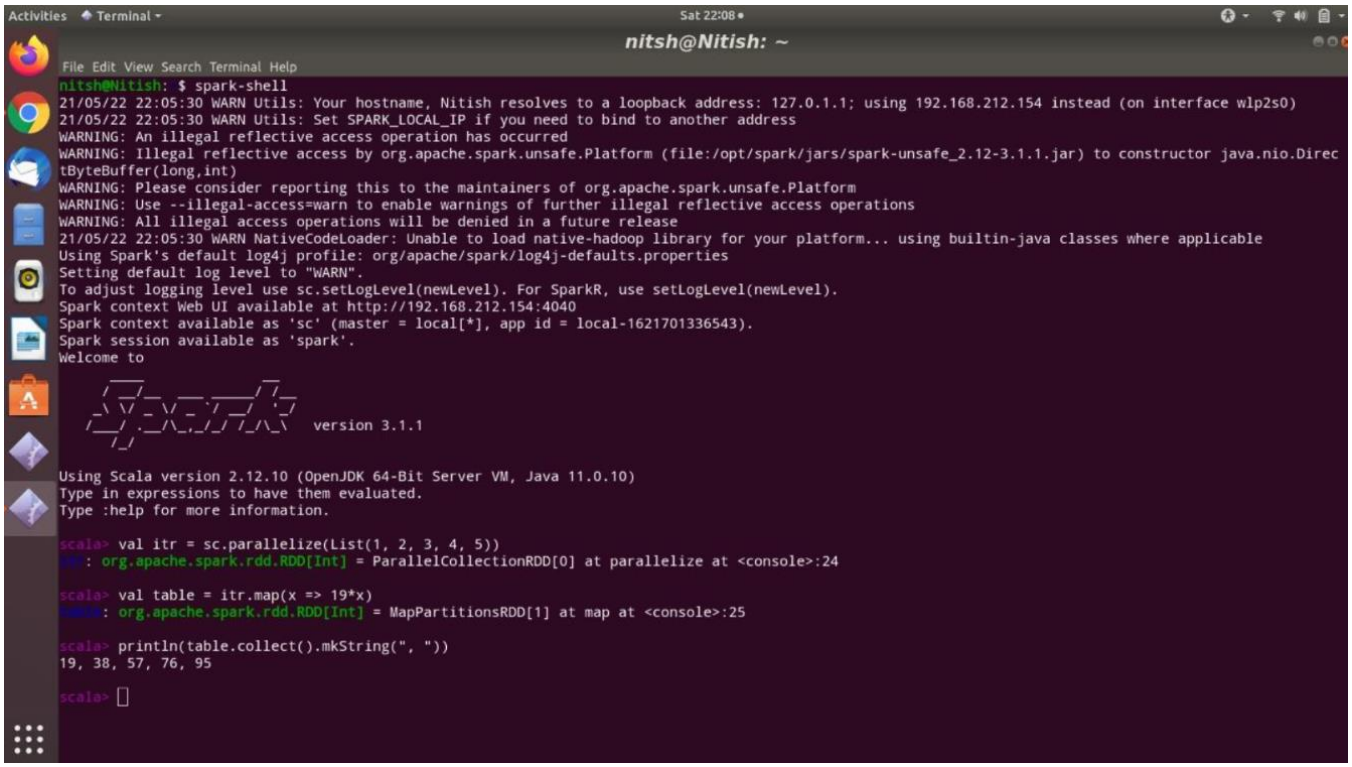
```
    } catch (IOException e) {
      throw new IllegalArgumentException(e);
    }
  }
  @Override
  public int compare(WritableComparable a, WritableComparable b) {
    if (a instanceof TextPair && b instanceof TextPair) {
      return ((TextPair) a).first.compareTo(((TextPair) b).first);
    }
    return super.compare(a, b);
  }
 }
 // ^^ TextPairFirstComparator
}
```

Output :



Program – 9

Screenshot of Spark Installed:



Program – 10

Using RDD and FlaMap 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

Input:

nitsh@Nitish:~$ cat input.txt

car

deer

car

deer

car

deer

car

deer

car

deer

bear

river

bear

river

bear

river

car

car

```
nitsh@Nitish:~$ spark-shell
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 3.1.1
      /_/

Using Scala version 2.12.10 (OpenJDK 64-Bit Server VM, Java 11.0.10)
Type in expressions to have them evaluated.
Type :help for more information.

scala> val textfile = sc.textFile("/home/nitsh/WEEK 10/input.txt")
textfile: org.apache.spark.rdd.RDD[String] = /home/nitsh/WEEK 10/input.txt MapPartitionsRDD[1] at textFile at <console>:24

scala> val counts = textfile.flatMap(line => line.split(" ")).map(word => (word,1)).reduceByKey(_ + _)
counts: org.apache.spark.rdd.RDD[(String, Int)] = ShuffledRDD[4] at reduceByKey at <console>:25

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

scala> val sorted = ListMap(counts.collect.sortWith(_._2 > _._2):_*)
[Stage 0:>                                                          (0 + 2)
sorted: scala.collection.immutable.ListMap[String,Int] = ListMap(car -> 7, deer -> 5, bear -> 3, river -> 3)

scala> println(sorted)
ListMap(car -> 7, deer -> 5, bear -> 3, river -> 3)
```

```
scala> for((k,v)<-sorted)
     | {
     |  if(v>4)
     |  {
     |   println(k+" - "+v)
     |  }
     | }
car - 7
deer - 5
```

Screenshot: