VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

BIG DATA ANALYTICS (20CS6PEBDA)

Submitted by

ROHAN SIWACH (1BM19CS132)

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

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

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

CERTIFICATE

This is to certify that the Lab work entitled “**BIG DATA ANALYTICS**” carried out by **ROHAN SIWACH (1BM19CS132),** who is bonafide student of **B. M. S. College of Engineering.** It is in partial fulfillment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum during the year 2022. The Lab report has been approved as it satisfies the academic requirements in respect of a**Big Data Analytics - (20CS6PEBDA)**work prescribed for the said degree.

Nameof the Lab-Incharge	**Mrs Rajeshwini**
Designation	Professor
Department of CSE	Department of CSE
BMSCE, Bengaluru	BMSCE, Bengaluru

`

Index Sheet

Course Outcome

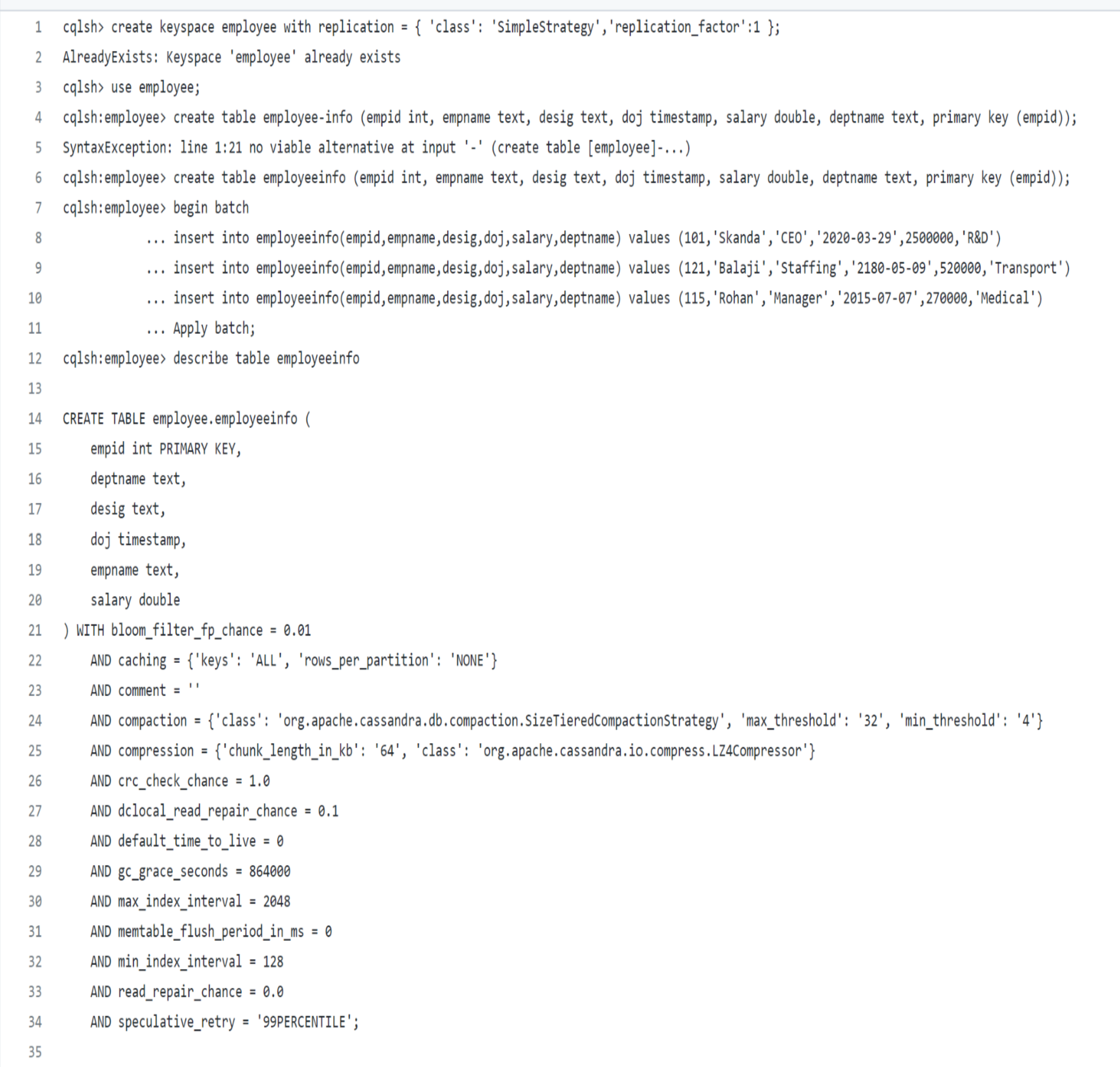
CO1	Apply the concept of NoSQL, Hadoop or Spark for a given task
CO2	Analyze the Big Data and obtain insight using data analytics mechanisms.
CO3	Design and implement Big data applications by applying NoSQL, Hadoop or Spark

Program 1 Mongo DB CRUD Demonstration

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

>db.Student.insert({_id:1,name:"Saffan",grade:9});
WriteResult({ "nInserted" : 1 })

>db.Student.find();
{ "_id" : 1, "name" : "Saffan", "grade" : 9 }
{ "_id" : 2, "name" : "Abc", "grade" : 10 }
{ "_id" : 3, "name" : "Mno", "grade" : 5 }
{ "_id" : 4, "name" : "Pqr", "grade" : 8 }

>db.Student.find().pretty();

> show collections;
Student

#HERE upsert=> update else insert if doesn't exist
>
db.Student.update({_id:6,name:"qwert"},{$set:{grade:4}},{upsert:true});
WriteResult({ "nMatched" : 0, "nUpserted" : 1, "nModified" : 0, "_id" : 6
})

> db.Student.update({_id:2},{$set:{age:21}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })


> db.Student.save({name:"zzz",_id:10,grade:8});
WriteResult({ "nMatched" : 0, "nUpserted" : 1, "nModified" : 0, "_id" : 10
})



> db.Student.update({_id:2},{$unset:{age:21}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
```

```
 > db.Student.update({_id:2},{$unset:{age:21}});
 WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })

 > db.Student.find({},{name:1,grade:1,_id:0});
 { "name" : "Saffan", "grade" : 9 }
 { "name" : "Abc", "grade" : 10 }
 { "name" : "Mno", "grade" : 5 }
 { "name" : "Pqr", "grade" : 8 }
 { "grade" : 4, "name" : "qwert" }


 > db.Student.find({grade:{$lt:5}},{name:1,grade:1,_id:0});
 { "grade" : 2, "name" : "qwert" }

 >
 db.Student.find({name:{$in:["Saffan","Abc","Mno"]}},{name:1,grade:1,_id
 :0});
 { "name" : "Saffan", "grade" : 9 }
 { "name" : "Abc", "grade" : 10 }
 { "name" : "Mno", "grade" : 5 }

 > db.Student.find({name:/^S/},{name:1,grade:1,_id:0});
 { "name" : "Saffan", "grade" : 9 }

 > db.Student.find({name:/.b/},{name:1,grade:1,_id:0});
 { "name" : "Abc", "grade" : 10 }

 > db.Student.count();
 6

 > db.Student.count({grade:9});
 1

 > db.Student.find().sort({name:1});
 { "_id" : 2, "name" : "Abc", "grade" : 10 }
 { "_id" : 3, "name" : "Mno", "grade" : 5 }
 { "_id" : 4, "name" : "Pqr", "grade" : 8 }
 { "_id" : 1, "name" : "Saffan", "grade" : 9 }
 { "_id" : 7, "name" : "kkk", "grade" : 6 }
 { "_id" : 6, "grade" : 2, "name" : "qwert" }


 > db.Student.find().sort({name:1,grade:-1});
 { "_id" : 2, "name" : "Abc", "grade" : 10 }
 { "_id" : 3, "name" : "Mno", "grade" : 5 }
 { "_id" : 4, "name" : "Pqr", "grade" : 8 }
 { "_id" : 1, "name" : "Saffan", "grade" : 9 }
 { "_id" : 7, "name" : "kkk", "grade" : 6 }
 { "_id" : 6, "grade" : 2, "name" : "qwert" }

 > db.Student.find({grade:8}).limit(3);
 { "_id" : 4, "name" : "Pqr", "grade" : 8 }
 { "_id" : 10, "name" : "zzz", "grade" : 8 }


 > db.Student.find().skip(2);
 { "_id" : 3, "name" : "Mno", "grade" : 5 }
 { "_id" : 4, "name" : "Pqr", "grade" : 8 }
 { "_id" : 6, "grade" : 2, "name" : "qwert" }
 { "_id" : 7, "name" : "kkk", "grade" : 6 }
 { "_id" : 10, "name" : "zzz", "grade" : 8 }

 ---------------------------------------------
 > db.food.insert({_id:1,fruits:['apple','mango']})
 WriteResult({ "nInserted" : 1 })


> db.food.update({_id:2},{$set:{'fruits.1':'apple'}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })

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

Program 2 Employee DB - Cassendra



```
cqlsh> create keyspace employee with replication = { 'class': 'SimpleStrategy','replication_factor':1 };
AlreadyExists: Keyspace 'employee' already exists
cqlsh> use employee;
cqlsh:employee> create table employee-info (empid int, empname text, desig text, doj timestamp, salary double, deptname text, primary key (empid));
SyntaxException: line 1:21 no viable alternative at input '-' (create table [employee]-...)
cqlsh:employee> create table employeeinfo (empid int, empname text, desig text, doj timestamp, salary double, deptname text, primary key (empid));
cqlsh:employee> begin batch
            ... insert into employeeinfo(empid,empname,desig,doj,salary,deptname) values (101,'Skanda','CEO','2020-03-29',2500000,'R&D')
            ... insert into employeeinfo(empid,empname,desig,doj,salary,deptname) values (121,'Balaji','Staffing','2180-05-09',520000,'Transport')
            ... insert into employeeinfo(empid,empname,desig,doj,salary,deptname) values (115,'Rohan','Manager','2015-07-07',270000,'Medical')
            ... Apply batch;
cqlsh:employee> describe table employeeinfo

CREATE TABLE employee.employeeinfo (
    empid int PRIMARY KEY,
    deptname text,
    desig text,
    doj timestamp,
    empname text,
    salary double
) WITH bloom_filter_fp_chance = 0.01
    AND caching = {'keys': 'ALL', 'rows_per_partition': 'NONE'}
    AND comment = ''
    AND compaction = {'class': 'org.apache.cassandra.db.compaction.SizeTieredCompactionStrategy', 'max_threshold': '32', 'min_threshold': '4'}
    AND compression = {'chunk_length_in_kb': '64', 'class': 'org.apache.cassandra.io.compress.LZ4Compressor'}
    AND crc_check_chance = 1.0
    AND dclocal_read_repair_chance = 0.1
    AND default_time_to_live = 0
    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_chance = 0.0
    AND speculative_retry = '99PERCENTILE';

```

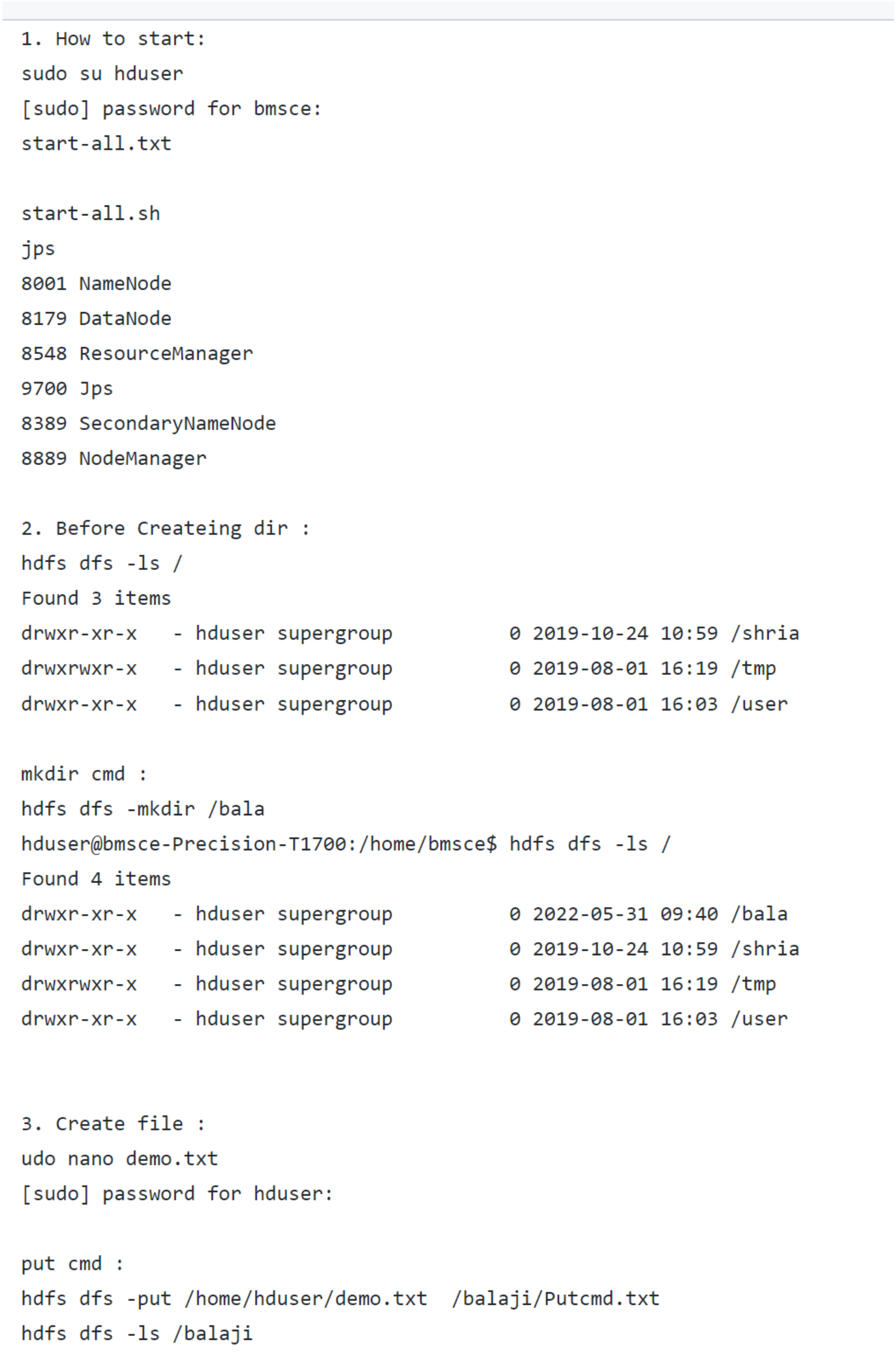
```
cqlsh:employee> select * from employeeinfo
            ... ;

 empid | deptname  | desig    | doj                             | empname | salary
-------+-----------+----------+---------------------------------+---------+---------
   121 | Transport | Staffing | 2180-05-08 18:30:00.000000+0000 |  Balaji | 5.2e+05
   115 |   Medical |  Manager | 2015-07-06 18:30:00.000000+0000 |   Rohan | 2.7e+05
   101 |       R&D |      CEO | 2020-03-28 18:30:00.000000+0000 |  Skanda | 2.5e+06

```

```
cqlsh:employee> alter table employeeinfo add projects text;
cqlsh:employee> select * from employeeinfo ;

 empid | deptname  | desig    | doj                             | empname | projects | salary
-------+-----------+----------+---------------------------------+---------+----------+---------
   121 | Transport | Staffing | 2180-05-08 18:30:00.000000+0000 |  Balaji |     null | 5.2e+05
   115 |   Medical |  Manager | 2015-07-06 18:30:00.000000+0000 |   Rohan |     null | 2.7e+05
   101 |       R&D |      CEO | 2020-03-28 18:30:00.000000+0000 |  Skanda |     null | 2.5e+06

(3 rows)
cqlsh:employee> update employeeinfo set projects='Cassandra' where empid = 101
            ... ;
cqlsh:employee> update employeeinfo set projects='Andriod' where empid = 121 ;
cqlsh:employee> update employeeinfo set projects='DevOps' where empid = 115 ;
cqlsh:employee> select * from employeeinfo ;

 empid | deptname  | desig    | doj                             | empname | projects  | salary
-------+-----------+----------+---------------------------------+---------+-----------+---------
   121 | Transport | Staffing | 2180-05-08 18:30:00.000000+0000 |  Balaji |   Andriod | 5.2e+05
   115 |   Medical |  Manager | 2015-07-06 18:30:00.000000+0000 |   Rohan |    DevOps | 2.7e+05
   101 |       R&D |      CEO | 2020-03-28 18:30:00.000000+0000 |  Skanda | Cassandra | 2.5e+06
```

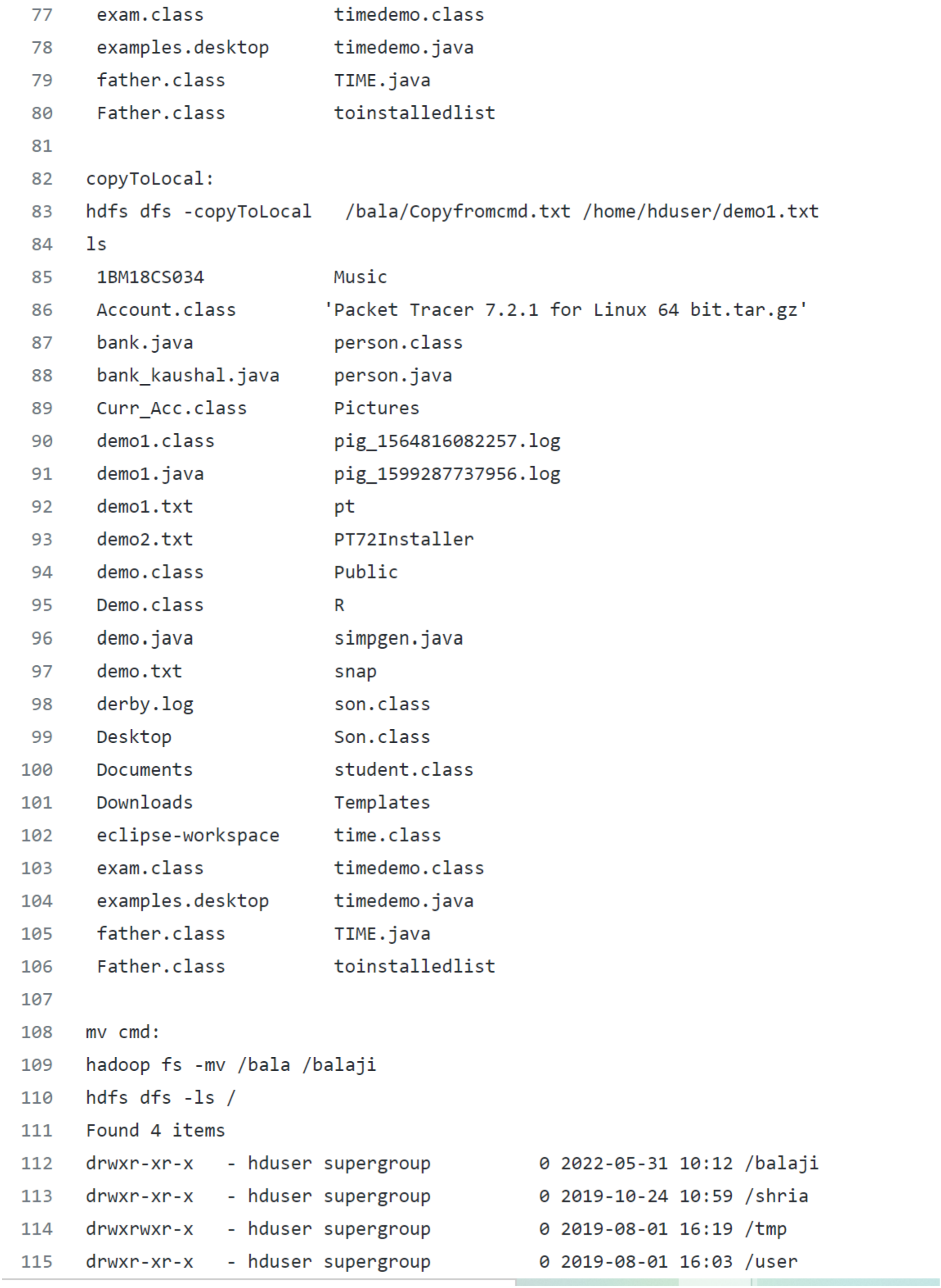
Program 3 : Library DB - Cassandra

```
1) CREATE keyspace library1 with replication={ 'class':'SimpleStrategy', 'replication_factor':1 };

 2) CREATE TABLE lib.libinfo1 ( s_id int, sname text, book text, bid int, doi timestamp, counter_val counter, PRIMARY KEY (s_id, sname, book, bid, doi) );

 3) update libinfo set counter_val=counter_val+1 where s_id=1 and sname='saf' and book='harry potter1' and bid=1 and doi='2022-05-05';

4) cqlsh:lib> update libinfo set counter_val=counter_val+1 where s_id=1 and sname='saf' and book='harry potter1';
cqlsh:lib> select * from libinfo;

 5) cqlsh:lib> select counter_val from libinfo where s_id=1 and sname='saf' and book='harry potter1';

 counter_val

 -------------

2

 6) COPY libinfo(s_id,sname,book,bid,doi,counter_val) TO 'data1.csv' WITH HEADER = TRUE; 7) COPY libinfo(s_id,sname,book,bid,doi) FROM 'libdata.csv' WITH HEADER = TRUE;
```

Program 4 : HADOOP



```
1. How to start:
sudo su hduser
[sudo] password for bmsce:
start-all.txt

start-all.sh
jps
8001 NameNode
8179 DataNode
8548 ResourceManager
9700 Jps
8389 SecondaryNameNode
8889 NodeManager

2. Before Createing dir :
hdfs dfs -ls /
Found 3 items
drwxr-xr-x   - hduser supergroup          0 2019-10-24 10:59 /shria
drwxrwxr-x   - hduser supergroup          0 2019-08-01 16:19 /tmp
drwxr-xr-x   - hduser supergroup          0 2019-08-01 16:03 /user

mkdir cmd :
hdfs dfs -mkdir /bala
hduser@bmsce-Precision-T1700:/home/bmsce$ hdfs dfs -ls /
Found 4 items
drwxr-xr-x   - hduser supergroup          0 2022-05-31 09:40 /bala
drwxr-xr-x   - hduser supergroup          0 2019-10-24 10:59 /shria
drwxrwxr-x   - hduser supergroup          0 2019-08-01 16:19 /tmp
drwxr-xr-x   - hduser supergroup          0 2019-08-01 16:03 /user


3. Create file :
udo nano demo.txt
[sudo] password for hduser:

put cmd :
hdfs dfs -put /home/hduser/demo.txt  /balaji/Putcmd.txt
hdfs dfs -ls /balaji
hdfs dfs -ls /balaji
Found 4 items
-rw-r--r--   1 hduser supergroup         21 2022-05-31 10:11 /balaji/CopyFromcmd.txt
-rw-r--r--   1 hduser supergroup         21 2022-05-31 10:12 /balaji/Copyfromcmd.txt
-rw-r--r--   1 hduser supergroup         21 2022-05-31 10:03 /balaji/Putcmd.txt
-rw-r--r--   1 hduser supergroup          0 2022-05-31 09:47 /balaji/hello.txt


copyFromLocal:
hdfs dfs -copyFromLocal /home/hduser/demo.txt  /balaji/Copyfromcmd.txt
hdfs dfs -ls /balaji
Found 4 items
-rw-r--r--   1 hduser supergroup         21 2022-05-31 10:11 /balaji/CopyFromcmd.txt
-rw-r--r--   1 hduser supergroup         21 2022-05-31 10:12 /balaji/Copyfromcmd.txt
-rw-r--r--   1 hduser supergroup         21 2022-05-31 10:03 /balaji/Putcmd.txt
-rw-r--r--   1 hduser supergroup          0 2022-05-31 09:47 /balaji/hello.txt


get :
hdfs dfs -get   /bala/Copyfromcmd.txt /home/hduser/demo2.txt
ls
 1BM18CS034            Music
 Account.class        'Packet Tracer 7.2.1 for Linux 64 bit.tar.gz'
 bank.java             person.class
 bank_kaushal.java     person.java
 Curr_Acc.class        Pictures
 demo1.class           pig_1564816082257.log
 demo1.java            pig_1599287737956.log
 demo1.txt             pt
 demo2.txt             PT72Installer
 demo.class            Public
 Demo.class            R
 demo.java             simpgen.java
 demo.txt              snap
 derby.log             son.class
 Desktop               Son.class
 Documents             student.class
 Downloads             Templates
 eclipse-workspace     time.class
```



```
 exam.class             timedemo.class
 examples.desktop       timedemo.java
 father.class           TIME.java
 Father.class           toinstalledlist

copyToLocal:
hdfs dfs -copyToLocal   /bala/Copyfromcmd.txt /home/hduser/demo1.txt
ls
 1BM18CS034             Music
 Account.class         'Packet Tracer 7.2.1 for Linux 64 bit.tar.gz'
 bank.java              person.class
 bank_kaushal.java      person.java
 Curr_Acc.class         Pictures
 demo1.class            pig_1564816082257.log
 demo1.java             pig_1599287737956.log
 demo1.txt              pt
 demo2.txt              PT72Installer
 demo.class             Public
 Demo.class             R
 demo.java              simpgen.java
 demo.txt               snap
 derby.log              son.class
 Desktop                Son.class
 Documents              student.class
 Downloads              Templates
 eclipse-workspace      time.class
 exam.class             timedemo.class
 examples.desktop       timedemo.java
 father.class           TIME.java
 Father.class           toinstalledlist

mv cmd:
hadoop fs -mv /bala /balaji
hdfs dfs -ls /
Found 4 items
drwxr-xr-x   - hduser supergroup          0 2022-05-31 10:12 /balaji
drwxr-xr-x   - hduser supergroup          0 2019-10-24 10:59 /shria
drwxrwxr-x   - hduser supergroup          0 2019-08-01 16:19 /tmp
drwxr-xr-x   - hduser supergroup          0 2019-08-01 16:03 /user
```

```
cp cmd:
hadoop fs -cp /balaji /sam
hdfs dfs -ls /
Found 5 items
drwxr-xr-x   - hduser supergroup          0 2022-05-31 10:12 /balaji
drwxr-xr-x   - hduser supergroup          0 2022-05-31 10:34 /sam
drwxr-xr-x   - hduser supergroup          0 2019-10-24 10:59 /shria
drwxrwxr-x   - hduser supergroup          0 2019-08-01 16:19 /tmp
drwxr-xr-x   - hduser supergroup          0 2019-08-01 16:03 /user
```