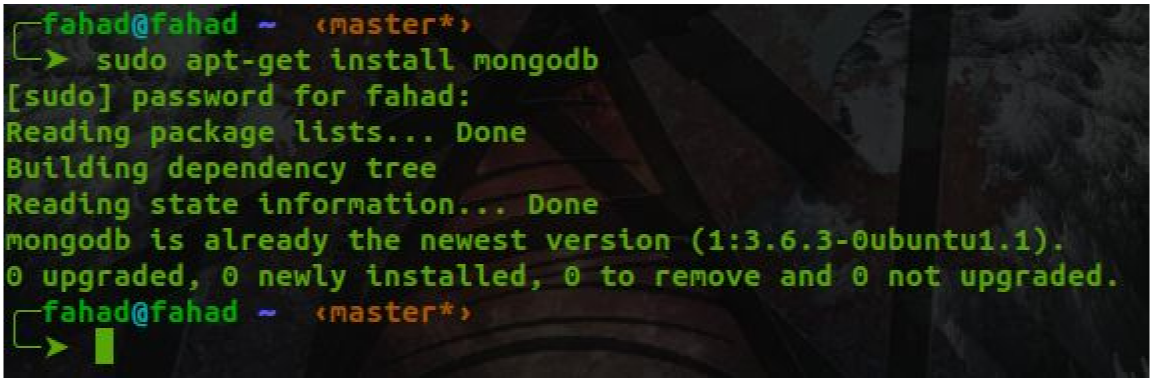
MONGO-DB

Ques 1:- Install latest version of MongoDB from apt-get repository

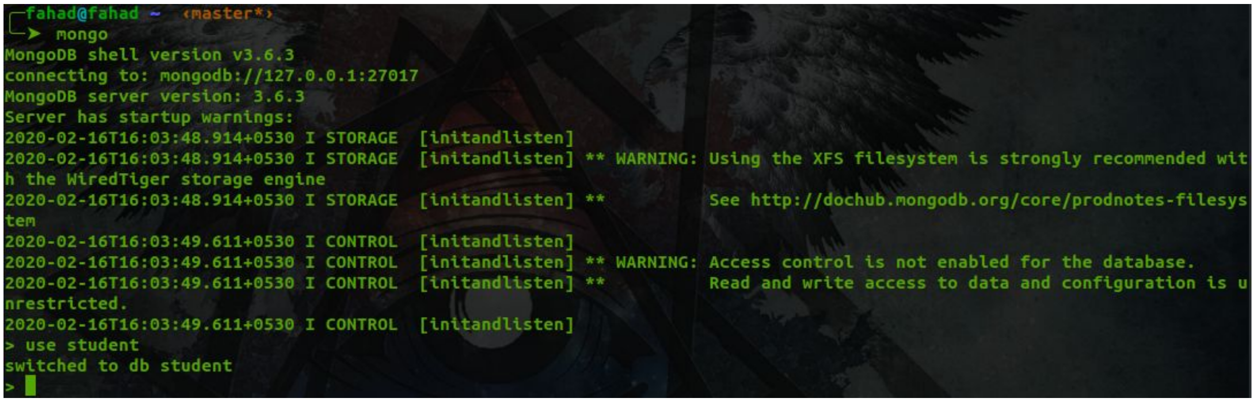
Ans 1:-



```
┌─fahad@fahad ~  ‹master*›
└─➤ sudo apt-get install mongodb
[sudo] password for fahad:
Reading package lists... Done
Building dependency tree
Reading state information... Done
mongodb is already the newest version (1:3.6.3-0ubuntu1.1).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
┌─fahad@fahad ~  ‹master*›
└─➤
```

Ques 2:- Create a database student

Ans 2:- With the help of **use** command we can make and get into that database

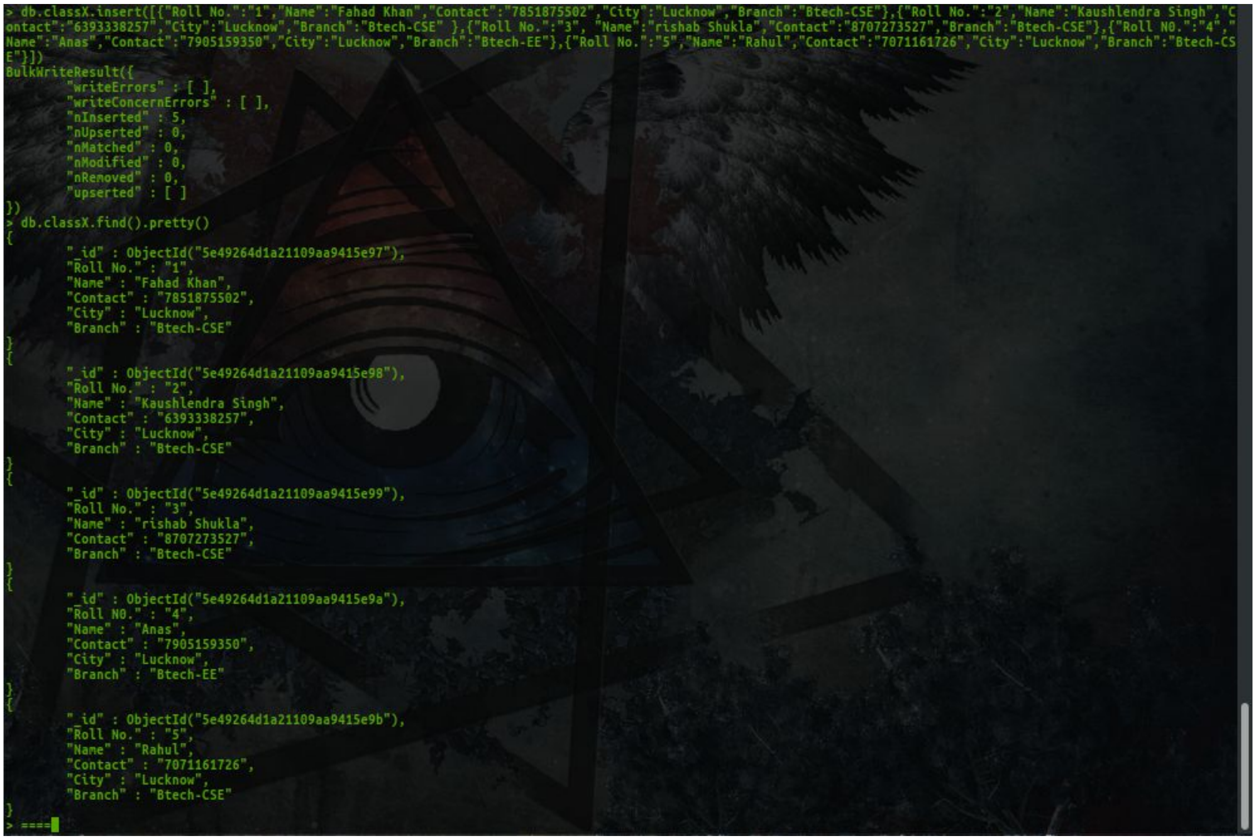


```
┌─fahad@fahad ~  ‹master*›
└─➤ mongo
MongoDB shell version v3.6.3
connecting to: mongodb://127.0.0.1:27017
MongoDB server version: 3.6.3
Server has startup warnings:
2020-02-16T16:03:48.914+0530 I STORAGE  [initandlisten]
2020-02-16T16:03:48.914+0530 I STORAGE  [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended wit
h the WiredTiger storage engine
2020-02-16T16:03:48.914+0530 I STORAGE  [initandlisten] **          See http://dochub.mongodb.org/core/prodnotes-filesys
tem
2020-02-16T16:03:49.611+0530 I CONTROL  [initandlisten]
2020-02-16T16:03:49.611+0530 I CONTROL  [initandlisten] ** WARNING: Access control is not enabled for the database.
2020-02-16T16:03:49.611+0530 I CONTROL  [initandlisten] **          Read and write access to data and configuration is u
nrestricted.
2020-02-16T16:03:49.611+0530 I CONTROL  [initandlisten]
> use student
switched to db student
>
```

Ques 3:- Insert operation : 5 students data (Name, Contact, City, Roll No, Branch)

Ans 3:- db.classX.insert(
{“data”}
)

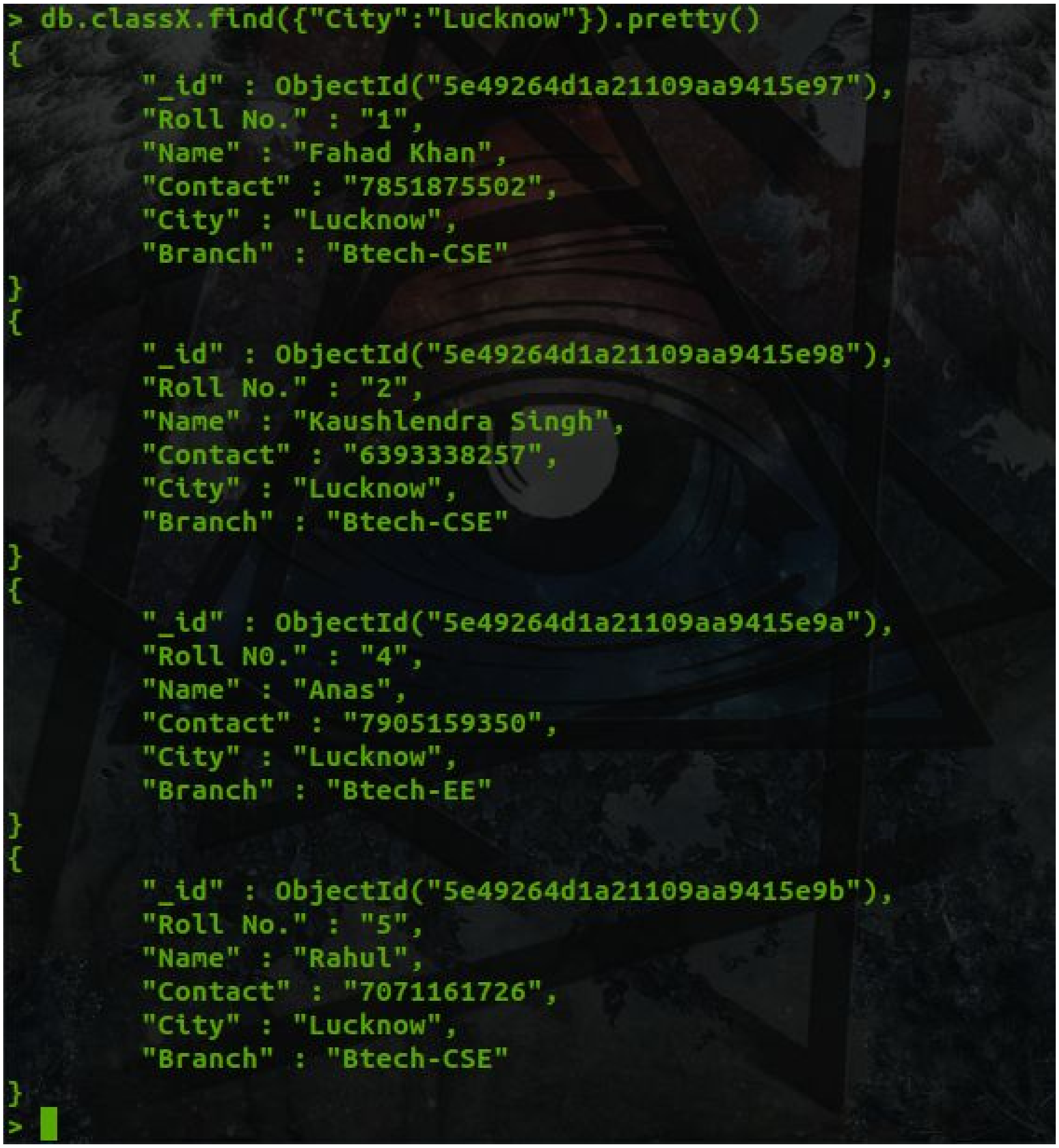
In this format the data is entered



```
> db.classX.insert([{"Roll No.":"1","Name":"Fahad Khan","Contact":"7851875502","City":"Lucknow","Branch":"Btech-CSE"},{"Roll No.":"2","Name":"Kaushlendra Singh","C
ontact":"6393338257","City":"Lucknow","Branch":"Btech-CSE" },{"Roll No.":"3", "Name":"rishab Shukla","Contact":"8707273527","Branch":"Btech-CSE"},{"Roll N0.":"4","
Name":"Anas","Contact":"7905159350","City":"Lucknow","Branch":"Btech-EE"},{"Roll No.":"5","Name":"Rahul","Contact":"7071161726","City":"Lucknow","Branch":"Btech-CS
E"}])
BulkWriteResult({
        "writeErrors" : [ ],
        "writeConcernErrors" : [ ],
        "nInserted" : 5,
        "nUpserted" : 0,
        "nMatched" : 0,
        "nModified" : 0,
        "nRemoved" : 0,
        "upserted" : [ ]
})
> db.classX.find().pretty()
{

        "_id" : ObjectId("5e49264d1a21109aa9415e97"),
        "Roll No." : "1",
        "Name" : "Fahad Khan",
        "Contact" : "7851875502",
        "City" : "Lucknow",
        "Branch" : "Btech-CSE"

}
{

        "_id" : ObjectId("5e49264d1a21109aa9415e98"),
        "Roll No." : "2",
        "Name" : "Kaushlendra Singh",
        "Contact" : "6393338257",
        "City" : "Lucknow",
        "Branch" : "Btech-CSE"

}
{

        "_id" : ObjectId("5e49264d1a21109aa9415e99"),
        "Roll No." : "3",
        "Name" : "rishab Shukla",
        "Contact" : "8707273527",
        "Branch" : "Btech-CSE"

}
{

        "_id" : ObjectId("5e49264d1a21109aa9415e9a"),
        "Roll N0." : "4",
        "Name" : "Anas",
        "Contact" : "7905159350",
        "City" : "Lucknow",
        "Branch" : "Btech-EE"

}
{

        "_id" : ObjectId("5e49264d1a21109aa9415e9b"),
        "Roll No." : "5",
        "Name" : "Rahul",
        "Contact" : "7071161726",
        "City" : "Lucknow",
        "Branch" : "Btech-CSE"

}
> ====
```

Ques 4:- Read operation : All the students belong to a particular city

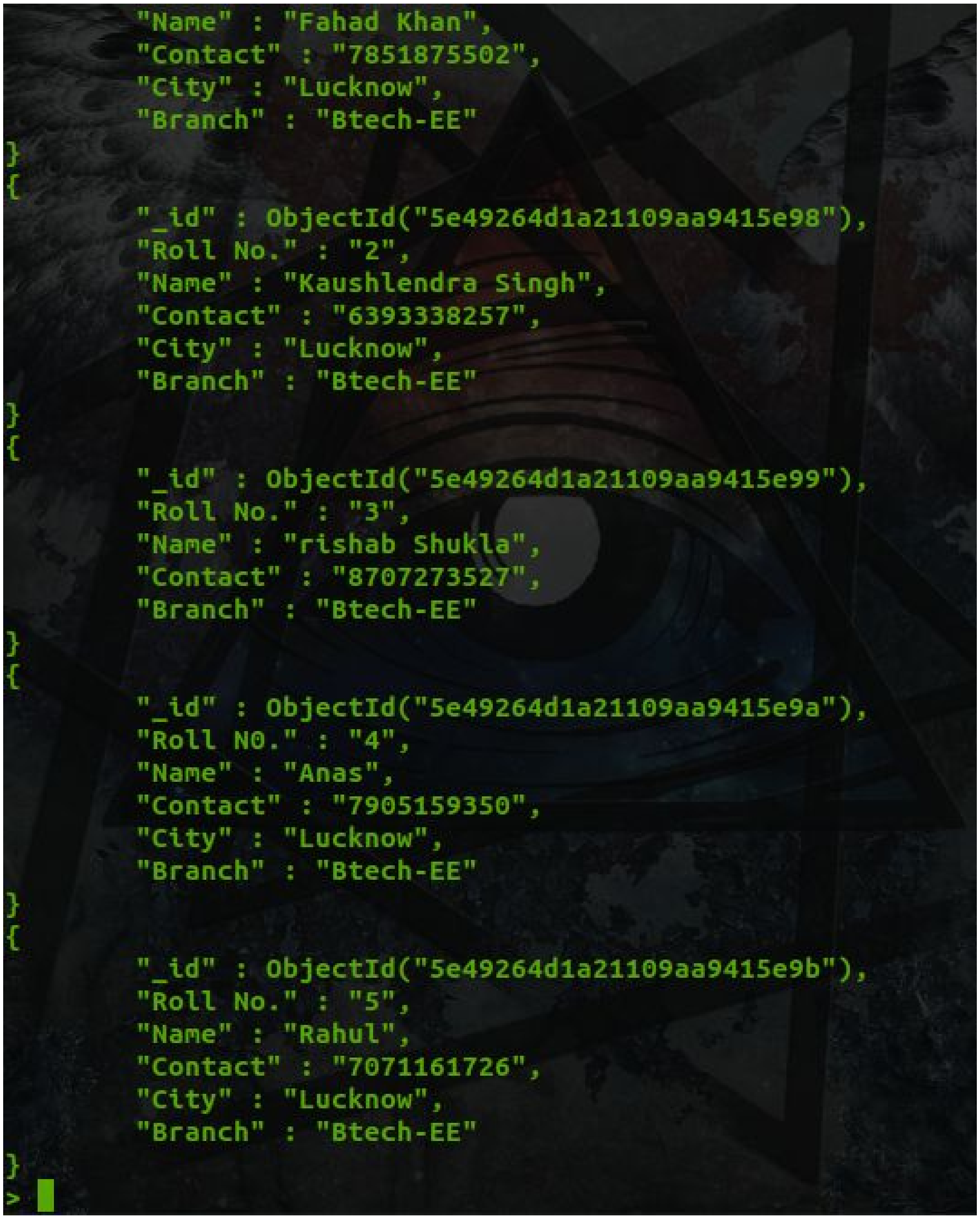
Ans 4:- db.classX.find({“City”:”Lucknow”}) it will list all records whose city is lucknow



```
> db.classX.find({"City":"Lucknow"}).pretty()
{
        "_id" : ObjectId("5e49264d1a21109aa9415e97"),
        "Roll No." : "1",
        "Name" : "Fahad Khan",
        "Contact" : "7851875502",
        "City" : "Lucknow",
        "Branch" : "Btech-CSE"
}
{
        "_id" : ObjectId("5e49264d1a21109aa9415e98"),
        "Roll No." : "2",
        "Name" : "Kaushlendra Singh",
        "Contact" : "6393338257",
        "City" : "Lucknow",
        "Branch" : "Btech-CSE"
}
{
        "_id" : ObjectId("5e49264d1a21109aa9415e9a"),
        "Roll N0." : "4",
        "Name" : "Anas",
        "Contact" : "7905159350",
        "City" : "Lucknow",
        "Branch" : "Btech-EE"
}
{
        "_id" : ObjectId("5e49264d1a21109aa9415e9b"),
        "Roll No." : "5",
        "Name" : "Rahul",
        "Contact" : "7071161726",
        "City" : "Lucknow",
        "Branch" : "Btech-CSE"
}
>
```

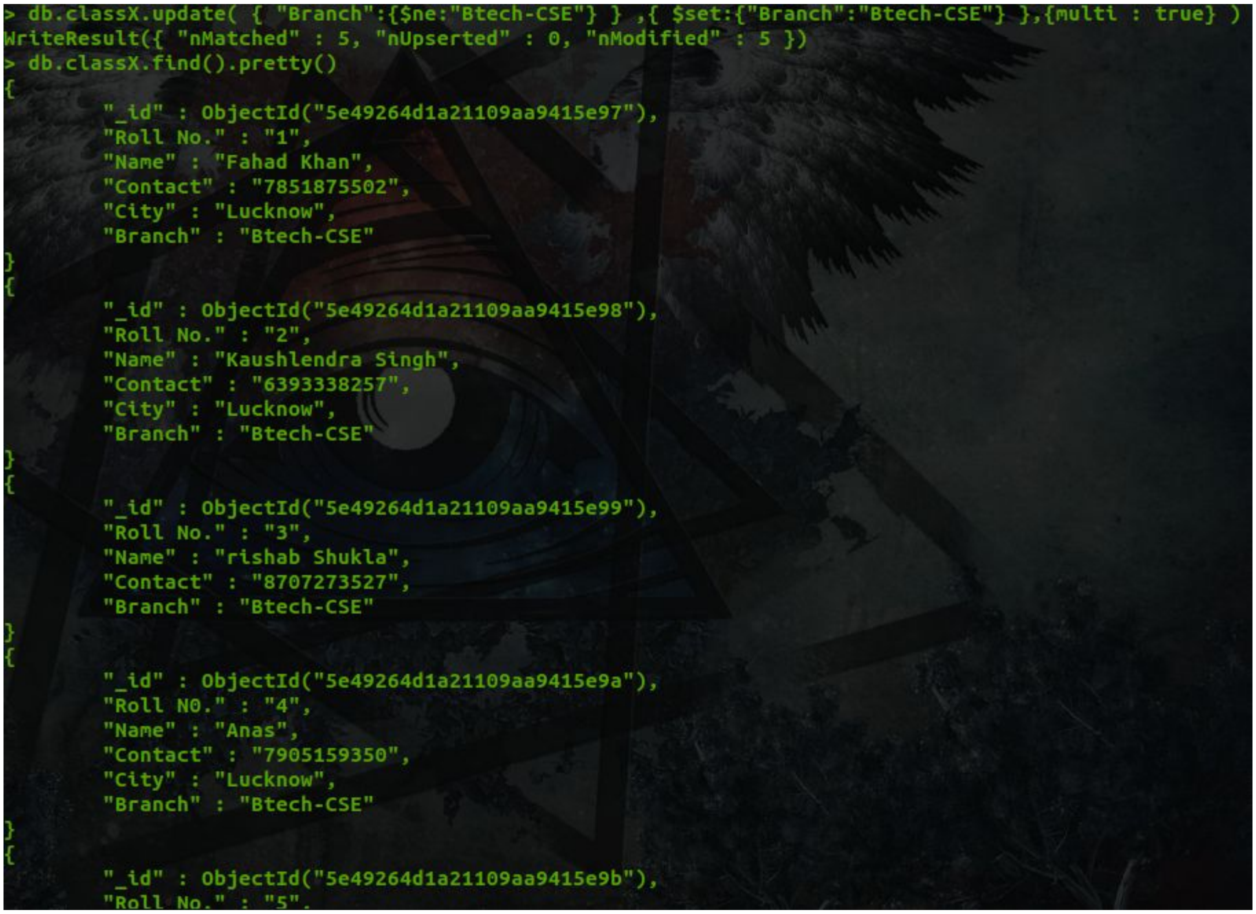
Ques 5:- Update operation : Update the branch of all the students to CSE

Ans 5:- Before modification



```
        "Name" : "Fahad Khan",
        "Contact" : "7851875502",
        "City" : "Lucknow",
        "Branch" : "Btech-EE"
}
{

        "_id" : ObjectId("5e49264d1a21109aa9415e98"),
        "Roll No." : "2",
        "Name" : "Kaushlendra Singh",
        "Contact" : "6393338257",
        "City" : "Lucknow",
        "Branch" : "Btech-EE"
}
{

        "_id" : ObjectId("5e49264d1a21109aa9415e99"),
        "Roll No." : "3",
        "Name" : "rishab Shukla",
        "Contact" : "8707273527",
        "Branch" : "Btech-EE"
}
{

        "_id" : ObjectId("5e49264d1a21109aa9415e9a"),
        "Roll NO." : "4",
        "Name" : "Anas",
        "Contact" : "7905159350",
        "City" : "Lucknow",
        "Branch" : "Btech-EE"
}
{

        "_id" : ObjectId("5e49264d1a21109aa9415e9b"),
        "Roll No." : "5",
        "Name" : "Rahul",
        "Contact" : "7071161726",
        "City" : "Lucknow",
        "Branch" : "Btech-EE"
}
> 
```

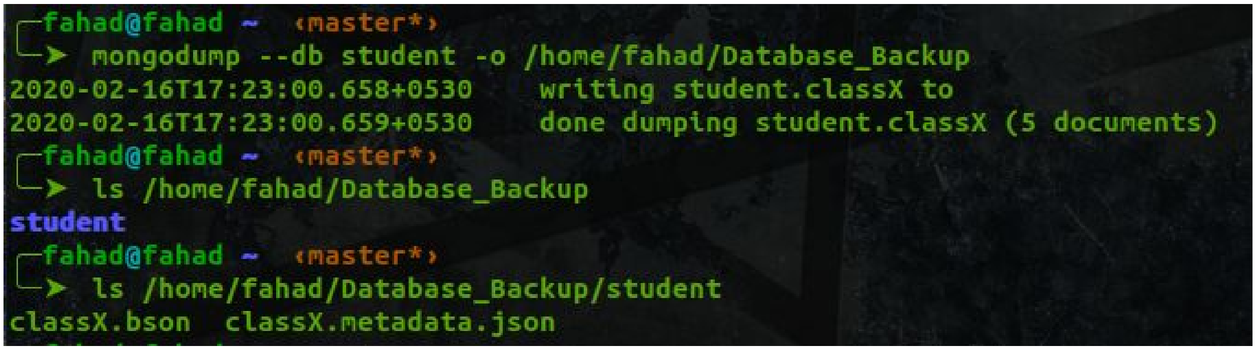
After modification



```
> db.classX.update( { "Branch":{$ne:"Btech-CSE"} } ,{ $set:{"Branch":"Btech-CSE"} },{multi : true} )
WriteResult({ "nMatched" : 5, "nUpserted" : 0, "nModified" : 5 })
> db.classX.find().pretty()
{
        "_id" : ObjectId("5e49264d1a21109aa9415e97"),
        "Roll No." : "1",
        "Name" : "Fahad Khan",
        "Contact" : "7851875502",
        "City" : "Lucknow",
        "Branch" : "Btech-CSE"
}
{
        "_id" : ObjectId("5e49264d1a21109aa9415e98"),
        "Roll No." : "2",
        "Name" : "Kaushlendra Singh",
        "Contact" : "6393338257",
        "City" : "Lucknow",
        "Branch" : "Btech-CSE"
}
{
        "_id" : ObjectId("5e49264d1a21109aa9415e99"),
        "Roll No." : "3",
        "Name" : "rishab Shukla",
        "Contact" : "8707273527",
        "Branch" : "Btech-CSE"
}
{
        "_id" : ObjectId("5e49264d1a21109aa9415e9a"),
        "Roll NO." : "4",
        "Name" : "Anas",
        "Contact" : "7905159350",
        "City" : "Lucknow",
        "Branch" : "Btech-CSE"
}
{
        "_id" : ObjectId("5e49264d1a21109aa9415e9b"),
        "Roll No." : "5",
```

Ques 6:- Take dump of the database

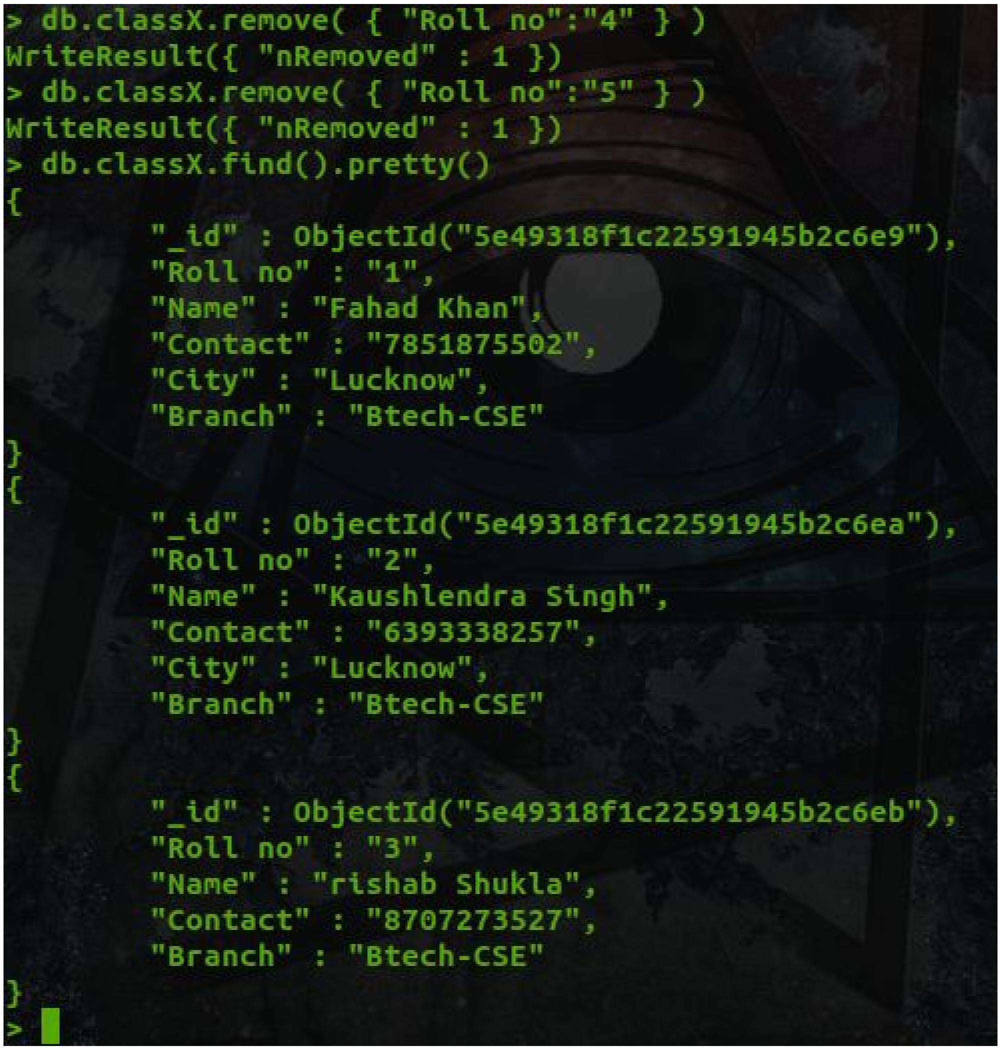
Ans 6:- by **mongodump --db student -o /directory-to-store-the-backup**



```
┌─fahad@fahad ~  ‹master*›
└─➤ mongodump --db student -o /home/fahad/Database_Backup
2020-02-16T17:23:00.658+0530    writing student.classX to
2020-02-16T17:23:00.659+0530    done dumping student.classX (5 documents)
┌─fahad@fahad ~  ‹master*›
└─➤ ls /home/fahad/Database_Backup
student
┌─fahad@fahad ~  ‹master*›
└─➤ ls /home/fahad/Database_Backup/student
classX.bson  classX.metadata.json
```

Ques 7:- Delete operation : Delete the record of last 2 students according to the roll number

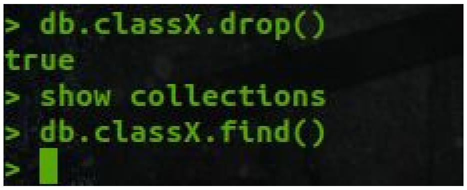
Ans 7:-db.classX.remove({“Roll no”: “4”}) will delete the documents



```
> db.classX.remove( { "Roll no":"4" } )
WriteResult({ "nRemoved" : 1 })
> db.classX.remove( { "Roll no":"5" } )
WriteResult({ "nRemoved" : 1 })
> db.classX.find().pretty()
{
        "_id" : ObjectId("5e49318f1c22591945b2c6e9"),
        "Roll no" : "1",
        "Name" : "Fahad Khan",
        "Contact" : "7851875502",
        "City" : "Lucknow",
        "Branch" : "Btech-CSE"
}
{
        "_id" : ObjectId("5e49318f1c22591945b2c6ea"),
        "Roll no" : "2",
        "Name" : "Kaushlendra Singh",
        "Contact" : "6393338257",
        "City" : "Lucknow",
        "Branch" : "Btech-CSE"
}
{
        "_id" : ObjectId("5e49318f1c22591945b2c6eb"),
        "Roll no" : "3",
        "Name" : "rishab Shukla",
        "Contact" : "8707273527",
        "Branch" : "Btech-CSE"
}
> 
```

Ques 8:- Drop the database

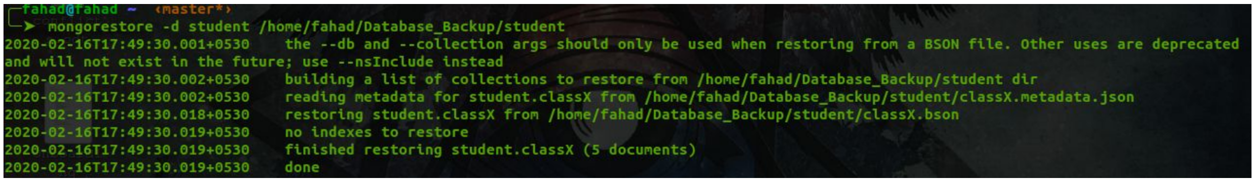
Ans 8:- db.classX.drop() will delete the whole database



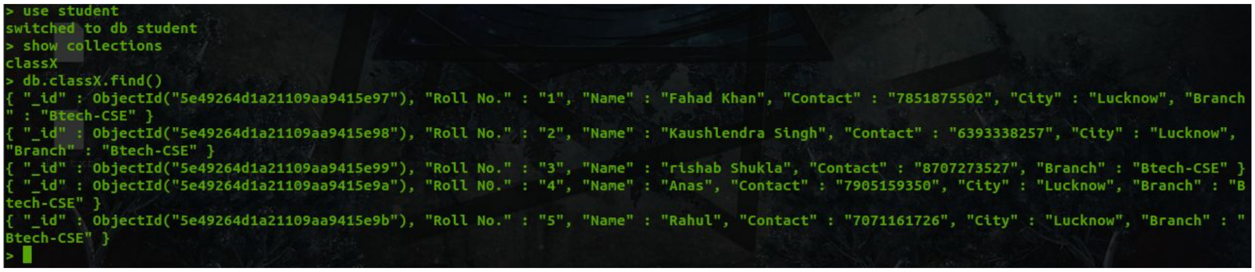
```
> db.classX.drop()
true
> show collections
> db.classX.find()
> 
```

Ques 9:- Restore the database again to have the full data

Ans 9:-



```
┌─fahad@fahad ~  ‹master*›
└─➤ mongorestore -d student /home/fahad/Database_Backup/student
2020-02-16T17:49:30.001+0530    the --db and --collection args should only be used when restoring from a BSON file. Other uses are deprecated
and will not exist in the future; use --nsInclude instead
2020-02-16T17:49:30.002+0530    building a list of collections to restore from /home/fahad/Database_Backup/student dir
2020-02-16T17:49:30.002+0530    reading metadata for student.classX from /home/fahad/Database_Backup/student/classX.metadata.json
2020-02-16T17:49:30.018+0530    restoring student.classX from /home/fahad/Database_Backup/student/classX.bson
2020-02-16T17:49:30.019+0530    no indexes to restore
2020-02-16T17:49:30.019+0530    finished restoring student.classX (5 documents)
2020-02-16T17:49:30.019+0530    done
```

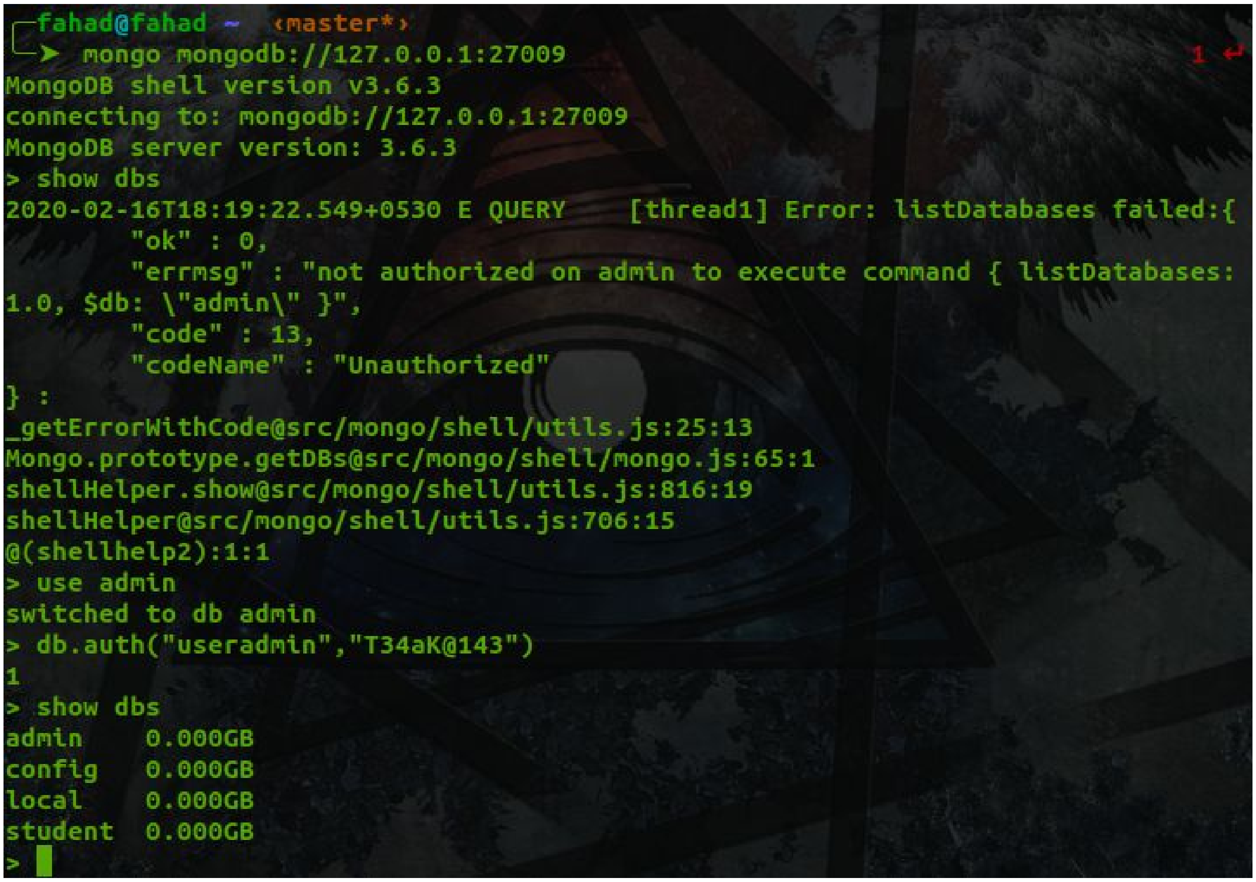


```
> use student
switched to db student
> show collections
classX
> db.classX.find()
{ "_id" : ObjectId("5e49264d1a21109aa9415e97"), "Roll No." : "1", "Name" : "Fahad Khan", "Contact" : "7851875502", "City" : "Lucknow", "Branch
" : "Btech-CSE" }
{ "_id" : ObjectId("5e49264d1a21109aa9415e98"), "Roll No." : "2", "Name" : "Kaushlendra Singh", "Contact" : "6393338257", "City" : "Lucknow",
"Branch" : "Btech-CSE" }
{ "_id" : ObjectId("5e49264d1a21109aa9415e99"), "Roll No." : "3", "Name" : "rishab Shukla", "Contact" : "8707273527", "Branch" : "Btech-CSE" }
{ "_id" : ObjectId("5e49264d1a21109aa9415e9a"), "Roll N0." : "4", "Name" : "Anas", "Contact" : "7905159350", "City" : "Lucknow", "Branch" : "B
tech-CSE" }
{ "_id" : ObjectId("5e49264d1a21109aa9415e9b"), "Roll No." : "5", "Name" : "Rahul", "Contact" : "7071161726", "City" : "Lucknow", "Branch" : "
Btech-CSE" }
> 
```

Ques 10:- Enable authentication on the Mongo

Ans 12:- By switching to admin then by giving the command listed below we can create a user for authentication

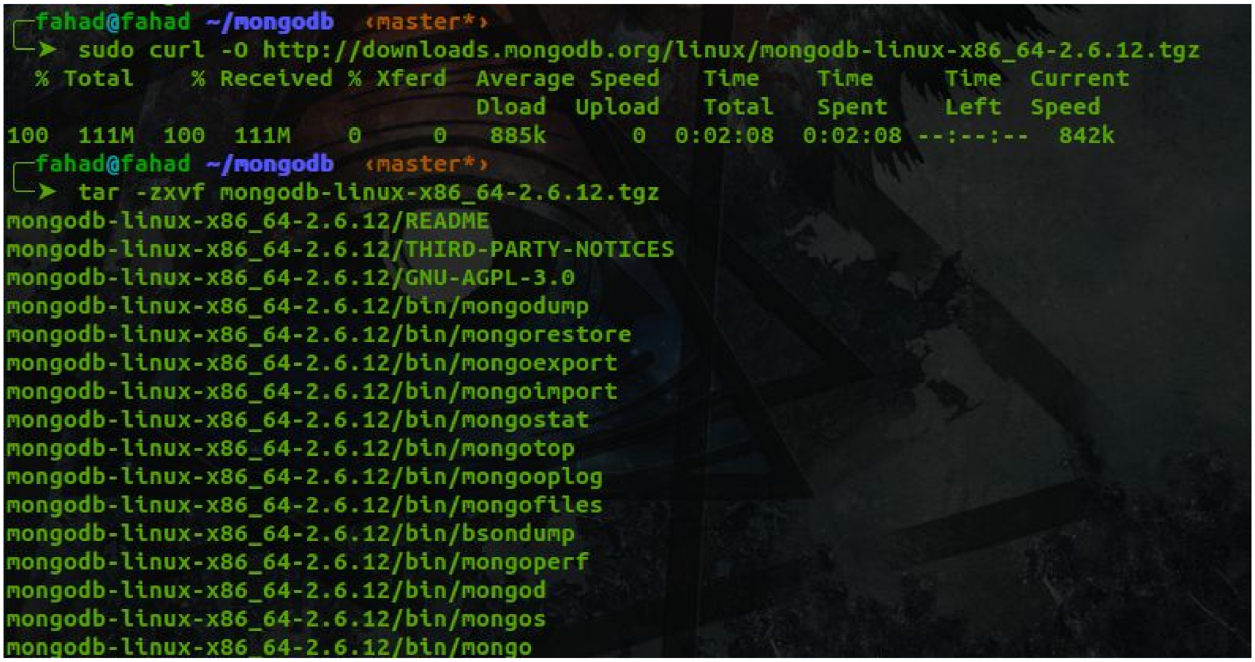
```
db.createUser(
 {
   user: "name_of_admin",
   pwd: "password",
   roles: [ { role: "userAdminAnyDatabase", db: "admin" } ]
 }
)
```



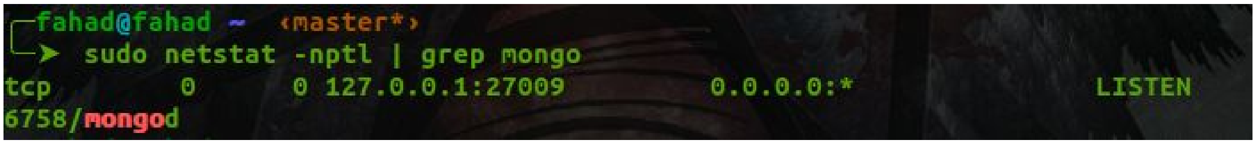
```
╭─fahad@fahad ~  ‹master*›
╰─➤ mongo mongodb://127.0.0.1:27009                                                    1 ↵
MongoDB shell version v3.6.3
connecting to: mongodb://127.0.0.1:27009
MongoDB server version: 3.6.3
> show dbs
2020-02-16T18:19:22.549+0530 E QUERY    [thread1] Error: listDatabases failed:{
        "ok" : 0,
        "errmsg" : "not authorized on admin to execute command { listDatabases:
1.0, $db: \"admin\" }",
        "code" : 13,
        "codeName" : "Unauthorized"
} :
_getErrorWithCode@src/mongo/shell/utils.js:25:13
Mongo.prototype.getDBs@src/mongo/shell/mongo.js:65:1
shellHelper.show@src/mongo/shell/utils.js:816:19
shellHelper@src/mongo/shell/utils.js:706:15
@(shellhelp2):1:1
> use admin
switched to db admin
> db.auth("useradmin","T34aK@143")
1
> show dbs
admin    0.000GB
config   0.000GB
local    0.000GB
student  0.000GB
> 
```

Ques 11:- Install another version of MongoDB from source (Version 2.6.3) and run it on port 27009

Ans 11:-



```
╭─fahad@fahad ~/mongodb  ‹master*›
╰─➤ sudo curl -O http://downloads.mongodb.org/linux/mongodb-linux-x86_64-2.6.12.tgz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  111M  100  111M    0     0   885k      0  0:02:08  0:02:08 --:--:--  842k
╭─fahad@fahad ~/mongodb  ‹master*›
╰─➤ tar -zxvf mongodb-linux-x86_64-2.6.12.tgz
mongodb-linux-x86_64-2.6.12/README
mongodb-linux-x86_64-2.6.12/THIRD-PARTY-NOTICES
mongodb-linux-x86_64-2.6.12/GNU-AGPL-3.0
mongodb-linux-x86_64-2.6.12/bin/mongodump
mongodb-linux-x86_64-2.6.12/bin/mongorestore
mongodb-linux-x86_64-2.6.12/bin/mongoexport
mongodb-linux-x86_64-2.6.12/bin/mongoimport
mongodb-linux-x86_64-2.6.12/bin/mongostat
mongodb-linux-x86_64-2.6.12/bin/mongotop
mongodb-linux-x86_64-2.6.12/bin/mongooplog
mongodb-linux-x86_64-2.6.12/bin/mongofiles
mongodb-linux-x86_64-2.6.12/bin/bsondump
mongodb-linux-x86_64-2.6.12/bin/mongoperf
mongodb-linux-x86_64-2.6.12/bin/mongod
mongodb-linux-x86_64-2.6.12/bin/mongos
mongodb-linux-x86_64-2.6.12/bin/mongo
```



```
┌─fahad@fahad ~  ‹master*›
└─➤ sudo netstat -nptl | grep mongo
tcp        0      0 127.0.0.1:27009         0.0.0.0:*               LISTEN
6758/mongod
```

Ques 12:- Create init service of Mongo installed later*

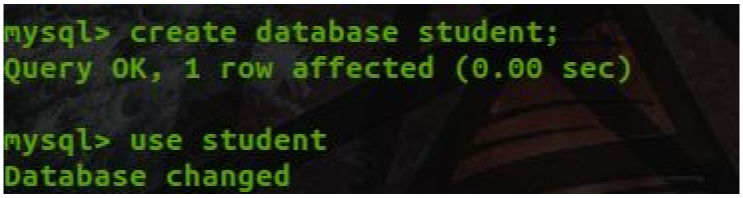
MYSQL

Ques 1. Install latest version of MYSQL from apt-get repository

Ans 1. sudo apt-get install mysql-server

Ques 2. Create a database student

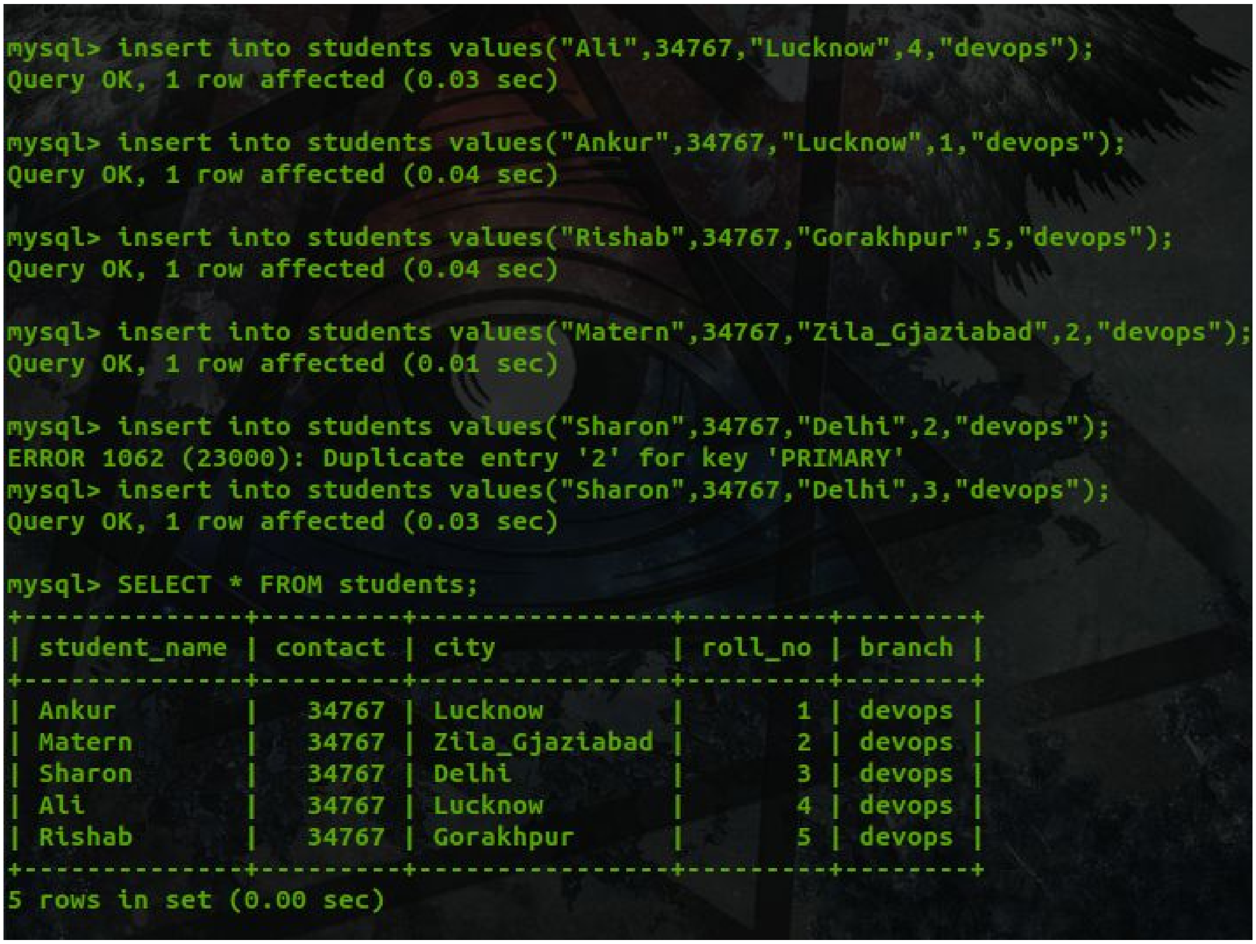
Ans 2.



```
mysql> create database student;
Query OK, 1 row affected (0.00 sec)

mysql> use student
Database changed
```

Ques 3. Insert operation : 5 students data (Name, Contact, City, Roll No, Branch)

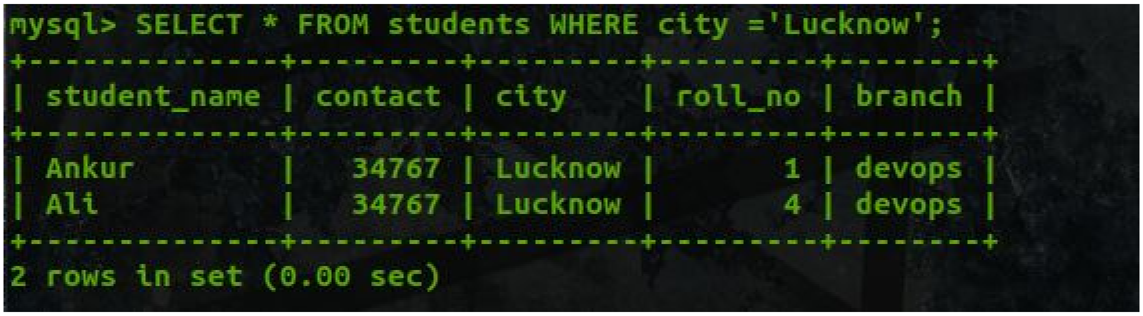
Ans 3.



```
mysql> insert into students values("Ali",34767,"Lucknow",4,"devops");
Query OK, 1 row affected (0.03 sec)

mysql> insert into students values("Ankur",34767,"Lucknow",1,"devops");
Query OK, 1 row affected (0.04 sec)

mysql> insert into students values("Rishab",34767,"Gorakhpur",5,"devops");
Query OK, 1 row affected (0.04 sec)

mysql> insert into students values("Matern",34767,"Zila_Gjaziabad",2,"devops");
Query OK, 1 row affected (0.01 sec)

mysql> insert into students values("Sharon",34767,"Delhi",2,"devops");
ERROR 1062 (23000): Duplicate entry '2' for key 'PRIMARY'
mysql> insert into students values("Sharon",34767,"Delhi",3,"devops");
Query OK, 1 row affected (0.03 sec)

mysql> SELECT * FROM students;
+--------------+---------+----------------+---------+--------+
| student_name | contact | city           | roll_no | branch |
+--------------+---------+----------------+---------+--------+
| Ankur        |   34767 | Lucknow        |       1 | devops |
| Matern       |   34767 | Zila_Gjaziabad |       2 | devops |
| Sharon       |   34767 | Delhi          |       3 | devops |
| Ali          |   34767 | Lucknow        |       4 | devops |
| Rishab       |   34767 | Gorakhpur      |       5 | devops |
+--------------+---------+----------------+---------+--------+
5 rows in set (0.00 sec)
```

Ques 4.Read operation : All the students belong to a particular city

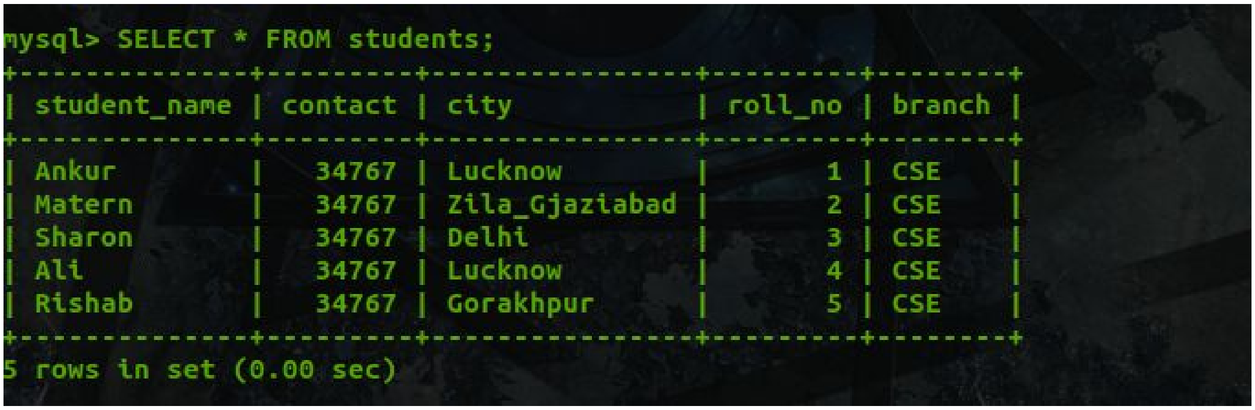
Ans 4.



```
mysql> SELECT * FROM students WHERE city ='Lucknow';
+--------------+---------+---------+---------+--------+
| student_name | contact | city    | roll_no | branch |
+--------------+---------+---------+---------+--------+
| Ankur        |   34767 | Lucknow |       1 | devops |
| Ali          |   34767 | Lucknow |       4 | devops |
+--------------+---------+---------+---------+--------+
2 rows in set (0.00 sec)
```

Ques 5. Update operation : Update the branch of all the students to CSE

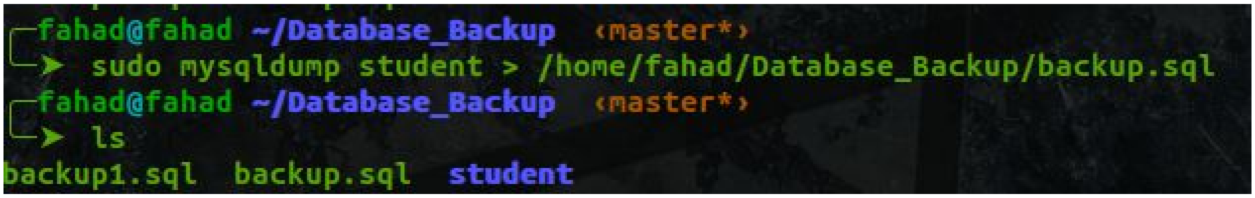
Ans 5.



```
mysql> SELECT * FROM students;
+--------------+---------+----------------+---------+--------+
| student_name | contact | city           | roll_no | branch |
+--------------+---------+----------------+---------+--------+
| Ankur        |   34767 | Lucknow        |       1 | CSE    |
| Matern       |   34767 | Zila_Gjaziabad |       2 | CSE    |
| Sharon       |   34767 | Delhi          |       3 | CSE    |
| Ali          |   34767 | Lucknow        |       4 | CSE    |
| Rishab       |   34767 | Gorakhpur      |       5 | CSE    |
+--------------+---------+----------------+---------+--------+
5 rows in set (0.00 sec)
```

Ques 6**. Take dump of the database**

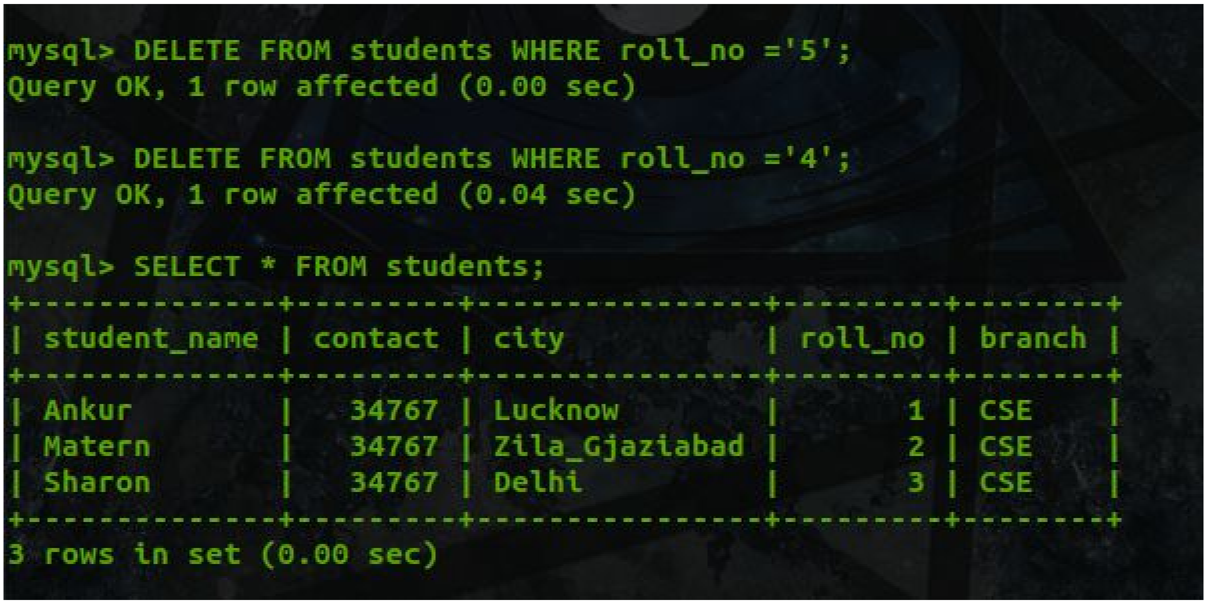
Ans 6.



```
┌─fahad@fahad ~/Database_Backup  ‹master*›
└─➤ sudo mysqldump student > /home/fahad/Database_Backup/backup.sql
┌─fahad@fahad ~/Database_Backup  ‹master*›
└─➤ ls
backup1.sql  backup.sql  student
```

Ques 7. Delete operation : Delete the record of last 2 students according to the roll number

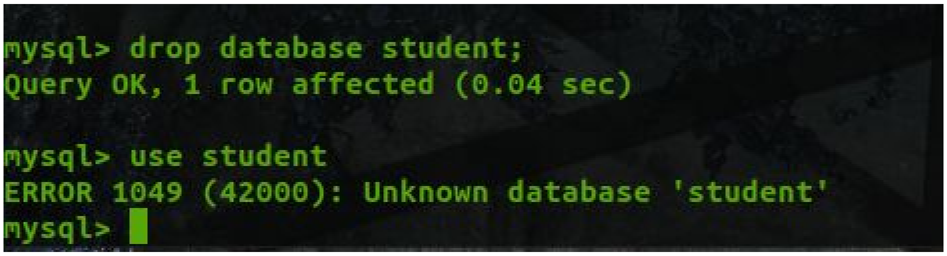
Ans 7.



```
mysql> DELETE FROM students WHERE roll_no ='5';
Query OK, 1 row affected (0.00 sec)

mysql> DELETE FROM students WHERE roll_no ='4';
Query OK, 1 row affected (0.04 sec)

mysql> SELECT * FROM students;
+--------------+---------+----------------+---------+--------+
| student_name | contact | city           | roll_no | branch |
+--------------+---------+----------------+---------+--------+
| Ankur        |   34767 | Lucknow        |       1 | CSE    |
| Matern       |   34767 | Zila_Gjaziabad |       2 | CSE    |
| Sharon       |   34767 | Delhi          |       3 | CSE    |
+--------------+---------+----------------+---------+--------+
3 rows in set (0.00 sec)
```

Ques 8. Drop the database

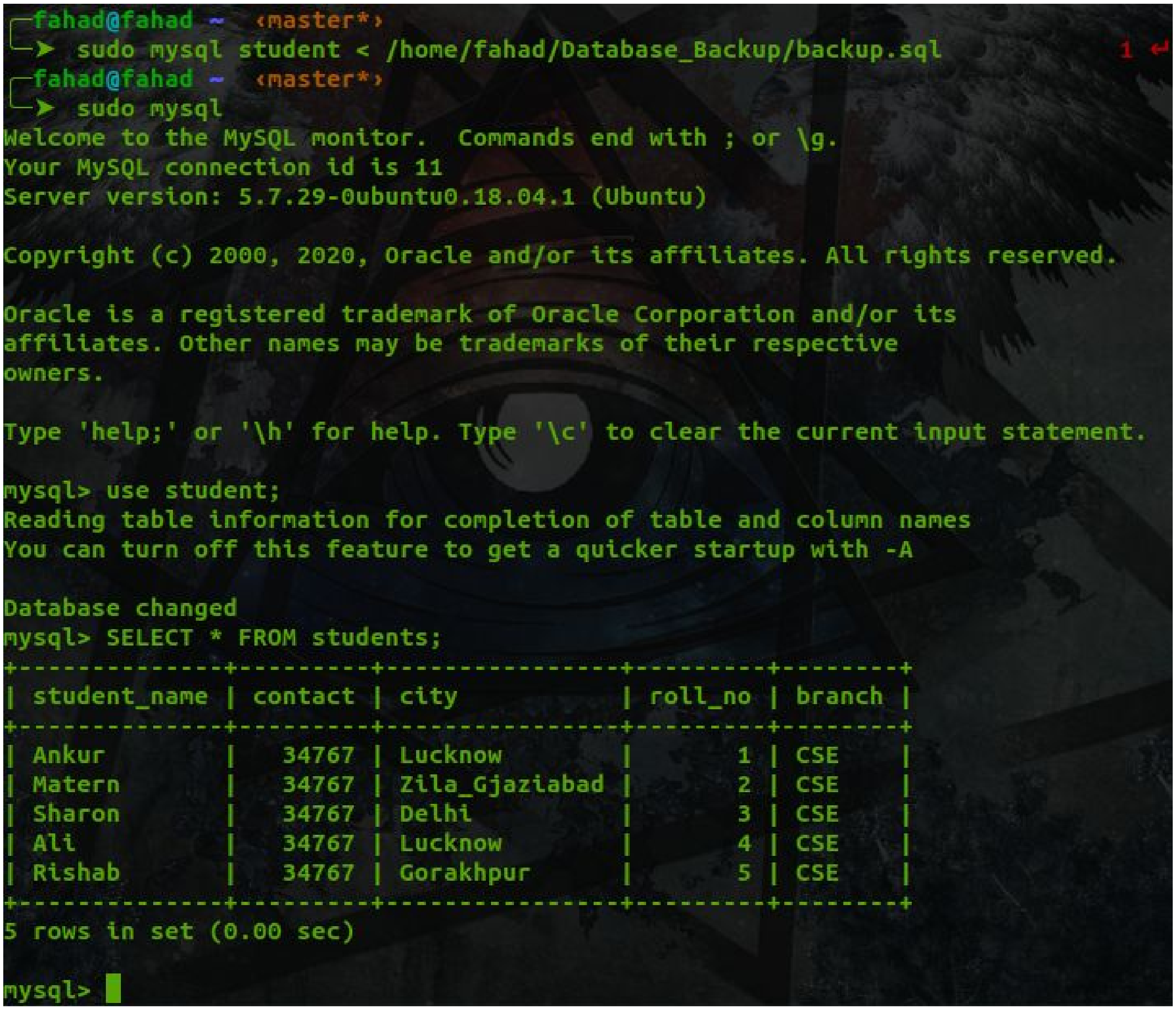
Ans 8 .



```
mysql> drop database student;
Query OK, 1 row affected (0.04 sec)

mysql> use student
ERROR 1049 (42000): Unknown database 'student'
mysql> 
```

Ques 9**. Restore the database again to have the full data**

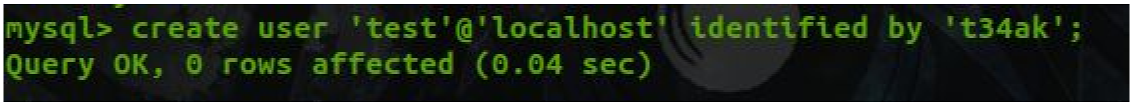
Ans 9.



```
┌─fahad@fahad ~  ‹master*›
└─➤ sudo mysql student < /home/fahad/Database_Backup/backup.sql                     1 ↵
┌─fahad@fahad ~  ‹master*›
└─➤ sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 11
Server version: 5.7.29-0ubuntu0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use student;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> SELECT * FROM students;
+--------------+---------+----------------+---------+--------+
| student_name | contact | city           | roll_no | branch |
+--------------+---------+----------------+---------+--------+
| Ankur        |   34767 | Lucknow        |       1 | CSE    |
| Matern       |   34767 | Zila_Gjaziabad |       2 | CSE    |
| Sharon       |   34767 | Delhi          |       3 | CSE    |
| Ali          |   34767 | Lucknow        |       4 | CSE    |
| Rishab       |   34767 | Gorakhpur      |       5 | CSE    |
+--------------+---------+----------------+---------+--------+
5 rows in set (0.00 sec)

mysql> 
```

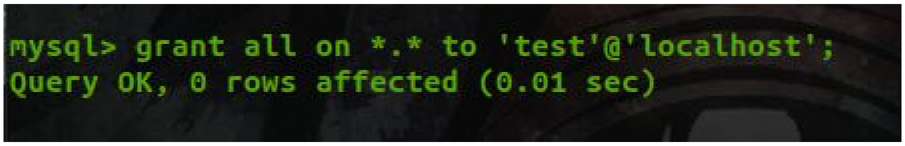
Ques 10:-Enable authentication on the Mongo

Ans 10:-First we have to create a user



```
mysql> create user 'test'@'localhost' identified by 't34ak';
Query OK, 0 rows affected (0.04 sec)
```

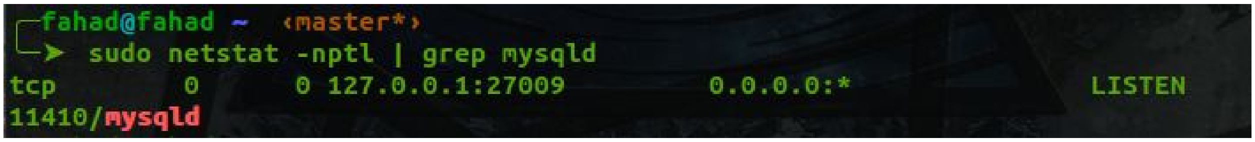
Now we have to grant the permission to the user to view and edit the content of the database



```
mysql> grant all on *.* to 'test'@'localhost';
Query OK, 0 rows affected (0.01 sec)
```

Granting permissions

Ques 11:- Change the port of the mysql to 27009
Ans 11:-



```
┌─fahad@fahad ~  ‹master*›
└─➤ sudo netstat -nptl | grep mysqld
tcp        0      0 127.0.0.1:27009         0.0.0.0:*               LISTEN
11410/mysqld
```