1. Create a database “Student” with the following attributes Rollno, Age, ContactNo, Email-Id.
2. Insert appropriate values

```
> db.createCollection("Student");
{ "ok" : 1 }
> db.Student.insert({"_id":1,"roll":1,name:"prateek","age":21,"email":"prateek
@gmail.com"});
WriteResult({ "nInserted" : 1 })
> db.Student.insert({"_id":2,"roll":2,name:"rahul;","age":21,"email":"rahul@gm
ail.com"});
WriteResult({ "nInserted" : 1 })
> db.Student.insert({"_id":3,"roll":10,name:"saif;","age":21,"email":"saif@gma
il.com"});
WriteResult({ "nInserted" : 1 })
> db.Student.insert({"_id":4,"roll":14,name:"wali;","age":21,"email":"wali@gma
il.com"});
WriteResult({ "nInserted" : 1 })
```

```
> db.Student.find()
{ "_id" : 1, "roll" : 1, "name" : "prateek", "age" : 21, "email" : "prateek@gm
ail.com" }
{ "_id" : 2, "roll" : 2, "name" : "rahul;", "age" : 21, "email" : "rahul@gmail
.com" }
{ "_id" : 3, "roll" : 10, "name" : "saif;", "age" : 21, "email" : "saif@gmail.
com" }
{ "_id" : 4, "roll" : 14, "name" : "wali;", "age" : 21, "email" : "wali@gmail.
com" }
```

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

```
> db.Student.update({"roll":10},{$set:{"email":"saifnew@gmail.com"}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.Student.find({"roll":10})
{ "_id" : 3, "roll" : 10, "name" : "saif;", "age" : 21, "email" : "saifnew@gmail.com" }
>
```

Replace the student name from “ABC” to “FEM” of rollno 1

```
> db.Student.update({"name":"wali;"},{$set:{"name":"walinew"}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
```

5. Export the created table into local file system

```
prateek@ubuntu:~$ mongoexport -d collegedb -c Student -f name,age,contactNO,em
ailId --type=csv -o student.csv
2020-10-08T15:08:43.450+0530    connected to: localhost
2020-10-08T15:08:43.451+0530    exported 4 records
prateek@ubuntu:~$ cat student.csv
name,age,contactNO,emailId
prateek,21,,
rahul;,21,,
saif;,21,,
walinew,21,,
```

6. Drop the table

```
> show dbs
admin      0.000GB
collegedb  0.000GB
company    0.000GB
config     0.000GB
local      0.000GB
mydb       0.000GB
> use collegedb
switched to db collegedb
> db.Student.drop()
true
```

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

```
prateek@ubuntu:~$ mongoimport -d collegedb -c Student --type csv --file studen
t.csv --headerline
2020-10-08T16:01:52.805+0530    connected to: localhost
2020-10-08T16:01:53.047+0530    imported 4 documents
```

2

Perform the following DB operations using MongoDB.

1. Create a collection by name Customers with the following attributes.
 Cust_id, Acc_Bal, Acc_Type

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

2. Insert at least 5 values into the table

```
> db.Customers.insert({"_id":1,"custid":1,"accountbalance":2000,"acctype":"saving"});
WriteResult({ "nInserted" : 1 })
> db.Customers.insert({"_id"21,"custid":2,"accountbalance":400,"acctype":"saving"});
uncaught exception: SyntaxError: missing : after property id :
@(shell):1:26
> db.Customers.insert({"_id":2,"custid":2,"accountbalance":400,"acctype":"saving"});
WriteResult({ "nInserted" : 1 })
> db.Customers.insert({"_id":3,"custid":3,"accountbalance":5000,"acctype":"saving"});
WriteResult({ "nInserted" : 1 })
> db.Customers.insert({"_id":4,"custid":4,"accountbalance":50000,"acctype":"saving"});
WriteResult({ "nInserted" : 1 })
> db.Customers.insert({"_id":5,"custid":5,"accountbalance":500000,"acctype":"saving"});
WriteResult({ "nInserted" : 1 })
```

3. Write a query to display those records whose total account balance is greater than 1200 of account type 'Z' for each customer_id.

```
> db.Customers.find({"accountbalance":{$gte:50000},"acctype":"saving"});
{ "_id" : 4, "custid" : 4, "accountbalance" : 50000, "acctype" : "saving" }
{ "_id" : 5, "custid" : 5, "accountbalance" : 500000, "acctype" : "saving" }
```

4. Determine Minimum and Maximum account balance for each customer_id.

```
> db.Customers.aggregate(
...     [
...       {
...         $group:
...           {
...             "_id": "$custid",
...             "max_bal": { $max: "$accountbalance" },
...             "min_bal": { $min: "$accountbalance" }
...           }
...       }
...     ]
... );
{ "_id" : 1, "max_bal" : 2000, "min_bal" : 2000 }
{ "_id" : 5, "max_bal" : 500000, "min_bal" : 90000 }
{ "_id" : 2, "max_bal" : 400, "min_bal" : 400 }
{ "_id" : 4, "max_bal" : 50000, "min_bal" : 50000 }
{ "_id" : 6, "max_bal" : 50000, "min_bal" : 50000 }
{ "_id" : 3, "max_bal" : 5000, "min_bal" : 5000 }
```

5. Export the created collection into local file system

Done in previous questions.

6. Drop the table

Done in previous questions.

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

Done in previous questions.