Lab-2

1. Create a collection by name Customers with the following attributes. Cust_id, Acc_Bal, Acc_Type
db.createCollection("Customers");

2. Insert at least 5 values into the table
db.Customers.insert({cust_id:1,Balance:200, Type:"S"});
db.Customer.insert({cust_id:1, Balance:1000, Type:"Z"});
db.Customer.insert({cust_id:2, Balance:100, Type:"Z"});
db.Customer.insert({cust_id:2, Balance:1000, Type:"C"});
db.Customer.insert({cust_id:3, Balance:500, Type:"Z"});
db.Customer.insert({cust_id:2, Balance:50, Type:"S"});

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.aggregate ({$match:{Type:"Z"}}, {$group : { _id : "$cust_id", TotAccBal :{$sum:"$Balance"} } }, {$match:{TotAccBal:{$gt:1200}}});

```
Atlas atlas-pn06xx-shard-0 [primary] test> db.Customer.find();
[
  {
    _id: ObjectId("66029ecf284165729835f95c"),
    cust_id: 1,
    Balance: 1000,
    Type: 'Z'
  },
  {
    _id: ObjectId("66029ee2284165729835f95d"),
    cust_id: 2,
    Balance: 100,
    Type: 'Z'
  },
  {
    _id: ObjectId("66029ef6284165729835f95e"),
    cust_id: 2,
    Balance: 1000,
    Type: 'C'
  },
  {
    _id: ObjectId("66029f04284165729835f95f"),
    cust_id: 3,
    Balance: 500,
    Type: 'Z'
  },
  {
    _id: ObjectId("66029f0e284165729835f960"),
    cust_id: 2,
    Balance: 50,
    Type: 'S'
  }
]
Atlas atlas-pn06xx-shard-0 [primary] test> |
```

4. Determine Minimum and Maximum account balance for each customer_id
db.Customers.aggregate ({$group : { _id : "$cust_id", minAccBal :{$min:"$Balance"},
maxAccBal :{$max:"$Balance"} }});

```
Atlas atlas-pn06xx-shard-0 [primary] test> db.Customers.aggregate ( {$group : { _id : "$cust_id", minAccBal :{$min:"$Balance"}, maxAccBal :{$max:"$Balance"} }});
[ { _id: 1, minAccBal: 200, maxAccBal: 200 } ]
```

5. Sort the documents based on Customer ID in ascending order and Account Balance in
descending order
db.Customers.find().sort({"cust_id":1, "Balance":-1})

```
Atlas atlas-pn06xx-shard-0 [primary] test> db.Customers.find().sort({"cust_id":1, "Balance":-1})
[
  {
    _id: ObjectId("66029da8284165729835f95b"),
    cust_id: 1,
    Balance: 200,
    Type: 'S'
  }
]
Atlas atlas-pn06xx-shard-0 [primary] test> |
```

6. Display only 2 nd and 3 rd records from the collection
db.Customer.find().skip(1).limit(2);

```
Atlas atlas-pn06xx-shard-0 [primary] test> db.Customer.find().skip(1).limit(2);
[
  {
    _id: ObjectId("66029ee2284165729835f95d"),
    cust_id: 2,
    Balance: 100,
    Type: 'Z'
  },
  {
    _id: ObjectId("66029ef6284165729835f95e"),
    cust_id: 2,
    Balance: 1000,
    Type: 'C'
  }
]
```