Create a collection by name Customer

customerdb | localhost:27017 | test

```
db.createCollection("customer");
```

0.006 sec.

Key	Value	Type
> ▣ (1)	{ 4 fields }	Object

Insert appropriate values

```
db.customer.find();
db.customer.insert({cust_id:1,acc_bal:14000,acc_type:"Z"});
db.customer.insert({cust_id:2,acc_bal:13000,acc_type:"x"});
db.customer.insert({cust_id:3,acc_bal:16000,acc_type:"y"});
db.customer.insert({cust_id:4,acc_bal:15000,acc_type:"Z"});
db.customer.insert({cust_id:5,acc_bal:10000,acc_type:"y"});
```

customer 0.132 sec.

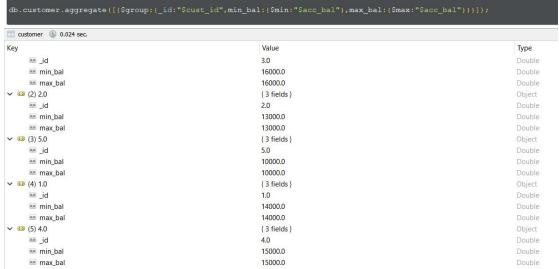| Key	Value	Type
> ▣ (1) ObjectId("5f843263d62ad17b02b40daa")	{ 4 fields }	Object
> ▣ (2) ObjectId("5f843263d62ad17b02b40dab")	{ 4 fields }	Object
> ▣ (3) ObjectId("5f843263d62ad17b02b40dac")	{ 4 fields }	Object
> ▣ (4) ObjectId("5f843263d62ad17b02b40dad")	{ 4 fields }	Object
> ▣ (5) ObjectId("5f843263d62ad17b02b40dae")	{ 4 fields }	Object

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

```
db.customer.aggregate([{$match:{acc_type:"Z"}},{$group:{_id:"$cust_id",Totalbal:{$sum:"$acc_bal"}}},{$match:{Totalbal:{$gt:12000}}}]);
```

customer 0.095 sec.

Key	Value	Type
∨ ▣ (1) 4.0	{ 2 fields }	Object
▦ _id	4.0	Double
▦ Totalbal	45000.0	Double
∨ ▣ (2) 1.0	{ 2 fields }	Object
▦ _id	1.0	Double
▦ Totalbal	42000.0	Double

Determine Minimum and Maximum account balance for each customer_id.

```
db.customer.aggregate([{$group:{_id:"$cust_id",min_bal:{$min:"$acc_bal"},max_bal:{$max:"$acc_bal"}}}]);
```

customer 0.024 sec.

Key	Value	Type
▦ _id	3.0	Double
▦ min_bal	16000.0	Double
▦ max_bal	16000.0	Double
∨ ▣ (2) 2.0	{ 3 fields }	Object
▦ _id	2.0	Double
▦ min_bal	13000.0	Double
▦ max_bal	13000.0	Double
∨ ▣ (3) 5.0	{ 3 fields }	Object
▦ _id	5.0	Double
▦ min_bal	10000.0	Double
▦ max_bal	10000.0	Double
∨ ▣ (4) 1.0	{ 3 fields }	Object
▦ _id	1.0	Double
▦ min_bal	14000.0	Double
▦ max_bal	14000.0	Double
∨ ▣ (5) 4.0	{ 3 fields }	Object
▦ _id	4.0	Double
▦ min_bal	15000.0	Double
▦ max_bal	15000.0	Double

Export the created collection into local file system

```
C:\Users\Laptop\Desktop\mongodb-database-tools-windows-x86_64-100.1.1\bin>mongoexport --db customerdb --collection customer --type=csv --fields cust_id,acc_bal,acc_type --out D:\c1.csv
2020-10-12T17:46:40.532+0530    connected to: mongodb://localhost/
2020-10-12T17:46:40.538+0530    exported 0 records
```

Drop the table

```
db.customer.drop();
```

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

```
C:\Users\Laptop\Desktop\mongodb-database-tools-windows-x86_64-100.1.1\bin>mongoimport --db customerdb --collection customer --type=csv --headerline --file D:\c1.csv
```