**CO-5**

**1.In database Employee.**

> db.empdetails.update({Name:"Mohan"},{$set:{dept:"Designer"}});

WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })

> db.empdetails.update({Name:"Raju"},{$set:{dept:"Tester"}});

WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })

1. find the average salary of each dept.

> db.empdetails.aggregate([{$group:{\_id:"$dept","Avg\_salary":{$avg:"$salary"}}}]);

{ "\_id" : "Developer", "Avg\_salary" : 79000 }

{ "\_id" : "Designer", "Avg\_salary" : 42000 }

{ "\_id" : "Tester", "Avg\_salary" : 42000 }

1. find the minimum salary of each dept.

> db.empdetails.aggregate([{$group:{\_id:"$dept","Min\_salary":{$min:"$salary"}}}]);

{ "\_id" : "Tester", "Min\_salary" : 25000 }

{ "\_id" : "Developer", "Min\_salary" : 79000 }

{ "\_id" : "Designer", "Min\_salary" : 32000 }

1. find the maximum salary of each dept.

> db.empdetails.aggregate([{$group:{\_id:"$dept","Max\_salary":{$max:"$salary"}}}]);

{ "\_id" : "Tester", "Max\_salary" : 59000 }

{ "\_id" : "Developer", "Max\_salary" : 79000 }

{ "\_id" : "Designer", "Max\_salary" : 52000 }

1. find the no.of employees of each dept.

> db.empdetails.aggregate([{$group:{\_id:"$dept","No of Employee":{$sum:1}}}]);

{ "\_id" : "Developer", "No of Employee" : 1 }

{ "\_id" : "Designer", "No of Employee" : 2 }

{ "\_id" : "Tester", "No of Employee" : 2 }

1. sort the collection empDetails in descending order of name

> db.empdetails.find().sort({Name:-1});

{ "\_id" : ObjectId("629ec0852e56075ffe218ad7"), "Name" : "Ram", "Age" : 23, "Email" : "ram@gmail.com", "Phone" : 7356841225, "salary" : 25000, "dept" : "Tester" }

{ "\_id" : ObjectId("629adaae881d51cafe890ca6"), "Name" : "Raju", "Age" : 27, "Email" : "raju@gmail.com", "Phone" : 9565852585, "salary" : 59000, "dept" : "Tester" }

{ "\_id" : ObjectId("629adafb881d51cafe890ca8"), "Name" : "Mohan", "Age" : 28, "Email" : "abiram@gmail.com", "Phone" : 9395658525, "salary" : 32000, "dept" : "Designer" }

{ "\_id" : ObjectId("629ec0b72e56075ffe218ad8"), "Name" : "Manu", "Age" : 23, "Email" : "manu@gmail.com", "Phone" : 7356841225, "salary" : 52000, "dept" : "Designer" }

{ "\_id" : ObjectId("629adad3881d51cafe890ca7"), "Name" : "Bhuvan", "Age" : 28, "Email" : "bhuvan@gmail.com", "Phone" : 9656582526, "salary" : 79000, "dept" : "Developer" }

1. Create a text index for ‘name’ and search for names mohan and bhuvan

> db.empdetails.find({$text:{$search:"Mohan Bhuvan"}});

{ "\_id" : ObjectId("629adad3881d51cafe890ca7"), "Name" : "Bhuvan", "Age" : 28, "Email" : "bhuvan@gmail.com", "Phone" : 9656582526, "salary" : 79000, "dept" : "Developer" }

{ "\_id" : ObjectId("629adafb881d51cafe890ca8"), "Name" : "Mohan", "Age" : 28, "Email" : "abiram@gmail.com", "Phone" : 9395658525, "salary" : 32000, "dept" : "Designer" }

**2.create a database Inventory and create an orders collection. Apply MapReduce operation for finding the total purchase of each customer.**

> use inventory

switched to db inventory

> db.createCollection("order");

{ "ok" : 1 }

> db.order.insert({custid:200,name:"Maya",item:"rice",price:340});

WriteResult({ "nInserted" : 1 })

> db.order.insert({custid:200,name:"Maya",item:"Wheet",price:240});

WriteResult({ "nInserted" : 1 })

> db.order.insert({custid:201,name:"Meera",item:"Wheet",price:240});

WriteResult({ "nInserted" : 1 })

> db.order.insert({custid:201,name:"Meera",item:"Sugar",price:80});

WriteResult({ "nInserted" : 1 })

> db.order.insert({custid:202,name:"Mohan",item:"Salt",price:50});

WriteResult({ "nInserted" : 1 })

> db.order.find();

{ "\_id" : ObjectId("629ec9312e56075ffe218ad9"), "custid" : 200, "name" : "Maya", "item" : "rice", "price" : 340 }

{ "\_id" : ObjectId("629ec9fd2e56075ffe218adb"), "custid" : 200, "name" : "Maya", "item" : "Wheet", "price" : 240 }

{ "\_id" : ObjectId("629eca3a2e56075ffe218adc"), "custid" : 201, "name" : "Meera", "item" : "Wheet", "price" : 240 }

{ "\_id" : ObjectId("629eca5b2e56075ffe218add"), "custid" : 201, "name" : "Meera", "item" : "Sugar", "price" : 80 }

{ "\_id" : ObjectId("629ecb0b2e56075ffe218ade"), "custid" : 202, "name" : "Mohan", "item" : "Salt", "price" : 50 }

> var mapFunction=function(){emit(this.custid,this.price);};

> var reduceFunction=function(key,values){return Array.sum(values);};

> db.order.mapReduce(mapFunction,reduceFunction,{'out':"map\_example"});

{ "result" : "map\_example", "ok" : 1 }

> db.map\_example.find();

{ "\_id" : 200, "value" : 580 }

{ "\_id" : 202, "value" : 50 }

{ "\_id" : 201, "value" : 320 }

>var mapFunction=function(){emit(this.custid,this.price);};

> var reduceFunction=function(key,values){return Array.avg(values);};

> db.order.mapReduce(mapFunction,reduceFunction,{query:{custid:{$gt:201}},'out':"map\_example"});

{ "result" : "map\_example", "ok" : 1 }

> db.map\_example.find();

{ "\_id" : 202, "value" : 50 }