1.**In database Employee.**

1. **find the average salary of each dept.**

> db.student.aggregate([{$group:{\_id:"$dept","avgsal":{$avg:"$salary"}}}])

{ "\_id" : "Tester", "avgsal" : 13500 }

{ "\_id" : "Designer", "avgsal" : 5000 }

{ "\_id" : "Developer", "avgsal" : 40550 }

1. **find the minimum salary of each dept.**

> db.student.aggregate([{$group:{\_id:"$dept","minimum sal":{$min:"$salary"}}}])

{ "\_id" : "Tester", "minimum sal" : 2000 }

{ "\_id" : "Designer", "minimum sal" : 4000 }

{ "\_id" : "Developer", "minimum sal" : 40550 }

1. **find the maximum salary of each dept.**

> db.student.aggregate([{$group:{\_id:"$dept","maximum sal":{$max:"$salary"}}}])

{ "\_id" : "Designer", "maximum sal" : 6000 }

{ "\_id" : "Developer", "maximum sal" : 40550 }

{ "\_id" : "Tester", "maximum sal" : 25000 }

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

> db.student.aggregate([{$group:{\_id:"$dept","no of emp":{$sum:1}}}])

{ "\_id" : "Designer", "no of emp" : 2 }

{ "\_id" : "Developer", "no of emp" : 1 }

{ "\_id" : "Tester", "no of emp" : 2 }

1. **sort the collection empDetails in descending order of name**

> db.student.find().sort({"name":-1}).pretty()

{

"\_id" : ObjectId("629ad05a7738eba5b7d72ab3"),

"name" : "raju",

"age" : 22,

"email" : "raju123@gmail.com",

"phone" : 1299667890,

"salary" : 4000,

"dept" : "Designer"

}

{

"\_id" : ObjectId("629ad03f7738eba5b7d72ab1"),

"name" : "pappu",

"age" : 21,

"email" : "ac123@gmail.com",

"phone" : 1278667890,

"salary" : 2000,

"dept" : "Tester"

}

{

"\_id" : ObjectId("629ad04d7738eba5b7d72ab2"),

"name" : "mohan",

"age" : 24,

"email" : "aswwwiin3@gmail.com",

"phone" : 1299667890,

"salary" : 6000,

"dept" : "Designer"

}

{

"\_id" : ObjectId("629f0bdf0775691cd5de0ff5"),

"name" : "manu",

"age" : 22,

"email" : "abc123@gmail.com",

"phone" : 1234567890,

"salary" : 25000,

"dept" : "Tester"

}

{

"\_id" : ObjectId("629ad0647738eba5b7d72ab4"),

"name" : "bhuvan",

"age" : 22,

"email" : "bhuvan123@gmail.com",

"phone" : 1289667890,

"salary" : 40550,

"dept" : "Developer"

}

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

> db.student.createIndex({name:"text"})

{

"numIndexesBefore" : 1,

"numIndexesAfter" : 2,

"createdCollectionAutomatically" : false,

"ok" : 1

}

> db.student.find({$text:{$search:"mohan bhuvan"}}).pretty()

{

"\_id" : ObjectId("629ad0647738eba5b7d72ab4"),

"name" : "bhuvan",

"age" : 22,

"email" : "bhuvan123@gmail.com",

"phone" : 1289667890,

"salary" : 40550,

"dept" : "Developer"

}

{

"\_id" : ObjectId("629ad04d7738eba5b7d72ab2"),

"name" : "mohan",

"age" : 24,

"email" : "aswwwiin3@gmail.com",

"phone" : 1299667890,

"salary" : 6000,

"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":"201","name":"Manu","item":"rice","price":340})

WriteResult({ "nInserted" : 1 })

> db.order.insert({"custid":"202","name":"Meera","item":"sugar","price":150})

WriteResult({ "nInserted" : 1 })

> db.order.insert({"custid":"202","name":"Meera","item":"wheat","price":250})

WriteResult({ "nInserted" : 1 })

> db.order.insert({"custid":"200","name":"Maya","item":"wheat","price":250})

WriteResult({ "nInserted" : 1 })

> db.order.find()

{ "\_id" : ObjectId("629f16630775691cd5de0ff6"), "custid" : "200", "name" : "Maya", "item" : "rice", "price" : 340 }

{ "\_id" : ObjectId("629f167c0775691cd5de0ff7"), "custid" : "201", "name" : "Manu", "item" : "rice", "price" : 340 }

{ "\_id" : ObjectId("629f169a0775691cd5de0ff8"), "custid" : "202", "name" : "Meera", "item" : "sugar", "price" : 150 }

{ "\_id" : ObjectId("629f16b90775691cd5de0ff9"), "custid" : "202", "name" : "Meera", "item" : "wheat", "price" : 250 }

{ "\_id" : ObjectId("629f16fd0775691cd5de0ffa"), "custid" : "200", "name" : "Maya", "item" : "wheat", "price" : 250 }

>

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

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

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

{ "result" : "mapexample", "ok" : 1 }

> db.mapexample.find();

{ "\_id" : "202", "value" : 400 }

{ "\_id" : "201", "value" : 340 }

{ "\_id" : "200", "value" : 590 }

>

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"})

> db.map\_example.find();

{ "\_id" : "202", "value" : 400 }