admin1@admin1-HP-280-Pro-G6-Microtower-PC:~/T1/3318/assignment12$ mongo

> db.employees.insertOne({ name: "Ram", department: "IT", salary: 500db.employees.insertOne({ name: "Ram", department: "IT", salary: 500db.employees.insertOne({ name: "Ram", department: "IT", salary: 50000, address: "st1", dependents: [ "d1", "d2" ] })

{

    "acknowledged" : true,

    "insertedId" : ObjectId("653222af3c11c33b64961eda")

}

> db.employees.find()

{ "\_id" : ObjectId("653222af3c11c33b64961eda"), "name" : "Ram", "department" : "IT", "salary" : 50000, "address" : "st1", "dependents" : [ "d1", "d2" ] }

> db.employees.insertMany([ { name: "Sham", department: "HR", salary: 60000, address: "st2", dependents: [ "d1", "d2" ] }, { name: "Ramesh", department: "Finance", salary: 70000, address: "st3", dependents: [ "d1" ] } ])

{

    "acknowledged" : true,

    "insertedIds" : [

        ObjectId("653222cb3c11c33b64961edb"),

        ObjectId("653222cb3c11c33b64961edc")

    ]

}

> db.employees.updateOne( { department: "IT" }, { $inc: { salary: 0.1 } } )

{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }

> db.students.find()

{ "\_id" : ObjectId("6528e1c40dc885cf83fc6798"), "rollno" : 1, "name" : "Ram", "prn" : "P1", "Branch" : "COMP", "TOC" : 99, "SPOS" : 99, "SPM" : 99, "DBMS" : 99 }

{ "\_id" : ObjectId("6528e1c40dc885cf83fc6799"), "rollno" : 2, "name" : "Sham", "prn" : "P2", "Branch" : "IT", "TOC" : 10, "SPOS" : 23, "SPM" : 89, "DBMS" : 10 }

{ "\_id" : ObjectId("6528e1c40dc885cf83fc679a"), "rollno" : 3, "name" : "Ramesh", "prn" : "P3", "Branch" : "IT", "TOC" : 56, "SPOS" : 23, "SPM" : 89, "DBMS" : 10 }

{ "\_id" : ObjectId("6528e1c40dc885cf83fc679b"), "rollno" : 4, "name" : "Suresh", "prn" : "P4", "Branch" : "COMP", "TOC" : 56, "SPOS" : 56, "SPM" : 47, "DBMS" : 78 }

{ "\_id" : ObjectId("6528e1c40dc885cf83fc679c"), "rollno" : 5, "name" : "Sita", "prn" : "P5", "Branch" : "ENTC", "TOC" : 56, "SPOS" : 3, "SPM" : 89, "DBMS" : 12 }

{ "\_id" : ObjectId("6528e1c40dc885cf83fc679d"), "rollno" : 6, "name" : "Gita", "prn" : "P6", "Branch" : "ENTC", "TOC" : 56, "SPOS" : 23, "SPM" : 66, "DBMS" : 43 }

{ "\_id" : ObjectId("6528e1c40dc885cf83fc679e"), "rollno" : 7, "name" : "Rajesh", "prn" : "P7", "Branch" : "COMP", "TOC" : 56, "SPOS" : 23, "SPM" : 79, "DBMS" : 41 }

{ "\_id" : ObjectId("6528e1c40dc885cf83fc679f"), "rollno" : 8, "name" : "Raghu", "prn" : "P8", "Branch" : "IT", "TOC" : 56, "SPOS" : 23, "SPM" : 62, "DBMS" : 78 }

> db.employees.updateOne( { name: "Ram" }, { $set: { address: "st321" } } )

{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }

> db.employees.find()

{ "\_id" : ObjectId("653222af3c11c33b64961eda"), "name" : "Ram", "department" : "IT", "salary" : 50000.1, "address" : "st321", "dependents" : [ "d1", "d2" ] }

{ "\_id" : ObjectId("653222cb3c11c33b64961edb"), "name" : "Sham", "department" : "HR", "salary" : 60000, "address" : "st2", "dependents" : [ "d1", "d2" ] }

{ "\_id" : ObjectId("653222cb3c11c33b64961edc"), "name" : "Ramesh", "department" : "Finance", "salary" : 70000, "address" : "st3", "dependents" : [ "d1" ] }

> db.employees.deleteOne({ name: "Ram" })

{ "acknowledged" : true, "deletedCount" : 1 }

> db.employees.find();

{ "\_id" : ObjectId("653222cb3c11c33b64961edb"), "name" : "Sham", "department" : "HR", "salary" : 60000, "address" : "st2", "dependents" : [ "d1", "d2" ] }

{ "\_id" : ObjectId("653222cb3c11c33b64961edc"), "name" : "Ramesh", "department" : "Finance", "salary" : 70000, "address" : "st3", "dependents" : [ "d1" ] }

> db.employees.drop()

true

> db.employees.insertMany([ { name: "Sham", department: "Mechanical", salary: 60000, address: "st2", dependents: [ "d1", "d2" ] }, { name: "Ramesh", department: "Mechanical", salary: 70000, address: "st3", dependents: [ "d1" ] } ])

{

    "acknowledged" : true,

    "insertedIds" : [

        ObjectId("653224743c11c33b64961edd"),

        ObjectId("653224743c11c33b64961ede")

    ]

}

>

> db.employees.mapReduce(

...    function() {

...       var max = this.arrayField[0];

...       for (var i = 1; i < this.arrayField.length; i++) {

...          if (this.arrayField[i] > max) {

...             max = this.arrayField[i];

...          }

...       }

...       emit(this.\_id, max);

...    },

...    function(key, values) {

...       return Math.max.apply(null, values);

...    },

...    {

...       query: { field: "aries" },

...       out: "output\_collection"

...    }

... )

{

    "result" : "output\_collection",

    "timeMillis" : 404,

    "counts" : {

        "input" : 0,

        "emit" : 0,

        "reduce" : 0,

        "output" : 0

    },

    "ok" : 1

}

> db.max\_salaries.find({\_id: "Mechanical"});

>

> db.max\_salaries.find({\_id: "Mechanical"});

> var mapFunction = function() {

...     emit(this.department, this.salary);

... };

> db.employees.mapReduce(

...    function() {

...       var max = this.arrayField[0];

...       for (var i = 1; i < this.arrayField.length; i++) {

...          if (this.arrayField[i] > max) {

...             max = this.arrayField[i];

...          }

...       }

...       emit(this.\_id, max);

...    },

...    function(key, values) {

...       return Math.max.apply(null, values);

...    },

...    {

...       query: { field: "aries" },

...       out: "output\_collection"

...    }

... )

> db.output\_collection.find({\_id: "Mechanical"});

> var mapFunction = function() {

...   if (this.department === "Mechanical") {

...     emit(this.department, this.salary);

...   }

... };

> var reduceFunction = function(key, values) {

...   return Math.max(...values); // Using spread operator to find the maximum value in the array

... };

> db.employees.mapReduce(

...   mapFunction,

...   reduceFunction,

...   { out: "highest\_salary\_mechanical" }

... )

{

    "result" : "highest\_salary\_mechanical",

    "timeMillis" : 426,

    "counts" : {

        "input" : 2,

        "emit" : 2,

        "reduce" : 1,

        "output" : 1

    },

    "ok" : 1

}

> db.highest\_salary\_mechanical.findOne().value

70000

> var mapFunction = function() {

...     emit(this.name, this.dependents.length);

... };

> var reduceFunction = function(key, values) {

...     return Array.sum(values);

... };

> db.employees.mapReduce(

...     mapFunction,

...     reduceFunction,

...     { out: "employee\_dependent\_count" }

... )

{

    "result" : "employee\_dependent\_count",

    "timeMillis" : 423,

    "counts" : {

        "input" : 2,

        "emit" : 2,

        "reduce" : 0,

        "output" : 2

    },

    "ok" : 1

}

> db.employee\_dependent\_count.find()

{ "\_id" : "Ramesh", "value" : 1 }

{ "\_id" : "Sham", "value" : 2 }

> var map = function() {

...   emit("count", 1);

... };

> var reduce = function(key, values) {

...   return Array.sum(values);

... };

> db.employees.mapReduce(

...   map,

...   reduce,

...   { out: "employee\_count" }

... )

{

    "result" : "employee\_count",

    "timeMillis" : 363,

    "counts" : {

        "input" : 2,

        "emit" : 2,

        "reduce" : 1,

        "output" : 1

    },

    "ok" : 1

}

> db.employee\_count.findOne().\_id

count

> db.employee\_count.findOne().find()

2023-10-20T12:42:16.958+0530 E QUERY    [thread1] TypeError: db.employee\_count.findOne(...).find is not a function :

@(shell):1:1

> db.employee\_count.findOne()

{ "\_id" : "count", "value" : 2 }

> db.employee\_count.findOne().value

2

> db.employees.insertMany([ { name: "Sham", department: "Mechanical", salary: 60000, address: "st2", dependents: [ "d1", "d2" ] }, { name: "Ramesh", department: "Mechanical", salary: 70000, address: "st3", dependents: [ "d1" ] } ])

{

    "acknowledged" : true,

    "insertedIds" : [

        ObjectId("653228973c11c33b64961edf"),

        ObjectId("653228973c11c33b64961ee0")

    ]

}

> db.employee\_count.findOne().value

2

> db.employees.find()

{ "\_id" : ObjectId("653224743c11c33b64961edd"), "name" : "Sham", "department" : "Mechanical", "salary" : 60000, "address" : "st2", "dependents" : [ "d1", "d2" ] }

{ "\_id" : ObjectId("653224743c11c33b64961ede"), "name" : "Ramesh", "department" : "Mechanical", "salary" : 70000, "address" : "st3", "dependents" : [ "d1" ] }

{ "\_id" : ObjectId("653228973c11c33b64961edf"), "name" : "Sham", "department" : "Mechanical", "salary" : 60000, "address" : "st2", "dependents" : [ "d1", "d2" ] }

{ "\_id" : ObjectId("653228973c11c33b64961ee0"), "name" : "Ramesh", "department" : "Mechanical", "salary" : 70000, "address" : "st3", "dependents" : [ "d1" ] }

> db.employees.drop()

true

> db.employees.insertMany([ { name: "Sham", department: "Mechanical", salary: 60000, address: "st2", dependents: [ "d1", "d2" ] }, { name: "Ramesh", department: "Mechanical", salary: 70000, address: "st3", dependents: [ "d1" ] } ,

... { name: "Suresh", department: "Mechanical", salary: 70000, address: "st3", dependents: [ "d1" ] } ,

... { name: "Ghansham", department: "Mechanical", salary: 70000, address: "st3", dependents: [ "d1","d2","d3"] }

...

... ])

{

    "acknowledged" : true,

    "insertedIds" : [

        ObjectId("653229183c11c33b64961ee1"),

        ObjectId("653229183c11c33b64961ee2"),

        ObjectId("653229183c11c33b64961ee3"),

        ObjectId("653229183c11c33b64961ee4")

    ]

}

> db.employees.mapReduce(

...   map,

...   reduce,

...   { out: "employee\_count" }

... )

{

    "result" : "employee\_count",

    "timeMillis" : 384,

    "counts" : {

        "input" : 4,

        "emit" : 4,

        "reduce" : 1,

        "output" : 1

    },

    "ok" : 1

}

> db.employee\_count.findOne().value

4

> db.employees.mapReduce(

...     mapFunction,

...     reduceFunction,

...     { out: "employee\_dependent\_count" }

... )

{

    "result" : "employee\_dependent\_count",

    "timeMillis" : 392,

    "counts" : {

        "input" : 4,

        "emit" : 4,

        "reduce" : 0,

        "output" : 4

    },

    "ok" : 1

}

> db.employee\_dependent\_count.find()

{ "\_id" : "Ghansham", "value" : 3 }

{ "\_id" : "Ramesh", "value" : 1 }

{ "\_id" : "Sham", "value" : 2 }

{ "\_id" : "Suresh", "value" : 1 }

>