**Session 4**

**21/04/2021**

**Problem Statement:**

**Create a Collection called as “Staff” under the database “Institute”, Add at least 10 records into the collection. Assume the following to be the details to be incorporated in the document:**

Document Format:

1. Name of the staff
2. Unique Staff ID – example – 086, (create it using \_id)
3. Department – [“Human resource” , “Development Team” , “Administration”, “Maintenance”, .. ]
4. Salary – int
5. Age – int
6. Designation - string
7. Increment – Yes/No
8. Performance status = [Good, Average, Excellent, Not Satisfactory]

Execute the following queries and record the output in the observation:

1. Group all the records by the department type and calculate the average sum of the salary from each department.
2. Group all the records by the staffid, find out the average salary by the age group.
3. Apply the map-reduce aggregation to project the name and amount owned by each staff by doing multiple jobs in a different department.
4. Match all the records having the performance status as “Good”, group them by their Name and compute the salary for each of them.
5. Demonstrate the usage of $match, $group, aggregate pipelines. Demonstrate the usage of $min, $last, $first, $sum, $max query operators with the $group operator.
6. Demonstrate the updateOne, UpdateMany, and replace operations with suitable examples

**Queries:**

>show dbs

BookDB 0.000GB

admin 0.000GB

config 0.000GB

local 0.000GB

> use Institute

switched to db Institute

> show collections

Staff

1. > db.Staff.aggregate([{ $group: {\_id: "$dept", total: { $avg: "$salary" } } } ]);

{ "\_id" : "Corporate", "total" : 700000 }

{ "\_id" : "Developer", "total" : 483333.3333333333 }

{ "\_id" : "HR", "total" : 126666.66666666667 }

{ "\_id" : "Helper Staff", "total" : 10000 }

{ "\_id" : "Programmer", "total" : 100000 }

{ "\_id" : "Management", "total" : 600000 }

1. > db.Staff.aggregate({$group:{\_id:"$age", Avg\_Sal : {$avg: "$salary"}}});

{ "\_id" : 29, "Avg\_Sal" : 900550 }

{ "\_id" : 43, "Avg\_Sal" : 10000 }

{ "\_id" : 25, "Avg\_Sal" : 30000 }

{ "\_id" : 37, "Avg\_Sal" : 700000 }

{ "\_id" : 36, "Avg\_Sal" : 500000 }

{ "\_id" : 57, "Avg\_Sal" : 100000 }

{ "\_id" : 45, "Avg\_Sal" : 600000 }

{ "\_id" : 59, "Avg\_Sal" : 50000 }

{ "\_id" : 30, "Avg\_Sal" : 300000 }

{ "\_id" : 23, "Avg\_Sal" : 50000 }

1. > var mapper = function() {emit(this.name,this.salary)}

> var reduce = function(Name, Sal) {return Array.sum(Sal)}

> db.Staff.mapReduce(mapper, reduce, {out: "Salary"})

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

1. > db.Staff.aggregate([{$match: {performance: "Good"}}, { $group: {\_id: "$name", Salary: {$max: "$salary"}}}]);

{ "\_id" : "Rajesh", "Salary" : 30000 }

1. > db.Staff.aggregate([{$match: {performance: "Good"}}, { $group: {\_id: "$name", Salary: {$max: "$salary"}}}]);

{ "\_id" : "Rajesh", "Salary" : 30000 }

> db.Staff.aggregate( { $group: {\_id: "$dept", Salary: {$sum: "$salary"}}});

{ "\_id" : "Management", "Salary" : 600000 }

{ "\_id" : "Helper Staff", "Salary" : 10000 }

{ "\_id" : "HR", "Salary" : 380000 }

{ "\_id" : "Programmer", "Salary" : 100000 }

{ "\_id" : "Developer", "Salary" : 2351100 }

{ "\_id" : "Corporate", "Salary" : 700000 }

> db.Staff.aggregate( { $group: {\_id: "$dept", Salary: {$min: "$salary"}}});

{ "\_id" : "Management", "Salary" : 600000 }

{ "\_id" : "Helper Staff", "Salary" : 10000 }

{ "\_id" : "HR", "Salary" : 30000 }

{ "\_id" : "Programmer", "Salary" : 100000 }

{ "\_id" : "Developer", "Salary" : 50000 }

{ "\_id" : "Corporate", "Salary" : 700000 }

> db.Staff.aggregate( { $group: {\_id: "$dept", Salary: {$last: "$salary"}}});

{ "\_id" : "Management", "Salary" : 600000 }

{ "\_id" : "Helper Staff", "Salary" : 10000 }

{ "\_id" : "HR", "Salary" : 50000 }

{ "\_id" : "Programmer", "Salary" : 100000 }

{ "\_id" : "Developer", "Salary" : 901100 }

{ "\_id" : "Corporate", "Salary" : 700000 }

> db.Staff.aggregate( { $group: {\_id: "$dept", Salary: {$first: "$salary"}}});

{ "\_id" : "Management", "Salary" : 600000 }

{ "\_id" : "Helper Staff", "Salary" : 10000 }

{ "\_id" : "HR", "Salary" : 30000 }

{ "\_id" : "Programmer", "Salary" : 100000 }

{ "\_id" : "Developer", "Salary" : 50000 }

{ "\_id" : "Corporate", "Salary" : 700000 }

> db.Staff.aggregate( { $group: {\_id: "$dept", Salary: {$max: "$salary"}}});

{ "\_id" : "Management", "Salary" : 600000 }

{ "\_id" : "Helper Staff", "Salary" : 10000 }

{ "\_id" : "HR", "Salary" : 300000 }

{ "\_id" : "Programmer", "Salary" : 100000 }

{ "\_id" : "Developer", "Salary" : 901100 }

{ "\_id" : "Corporate", "Salary" : 700000}

1. > db.Staff.updateOne({\_id: "S001"},{$set : {salary: 61900}});

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

> db.Staff.updateMany({ salary: 30000}, { $set: { salary: 32000 }});

{ "acknowledged" : true, "matchedCount" : 0, "modifiedCount" : 0 }

> db.Staff.replaceOne({\_id: "S001"}, { salary: 43000 });

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