MongoDB - Aggregation and Indexing: Design and Develop MongoDB Queries using aggregation and

indexing with suitable example using MongoDB.

Create collection student{ Rollno ,Name, Class, Div, Subject, Marks, Address} and enter 6 entries or

more. And perform the following:

1. Find the maximum marks of student in DS who stay in the same city

2. Calculates the average of given marks.

3. Inserts the any value to an array in the resulting document.

4. Create a simple and compound index on name and class.

5. Delete index of name.

6. Implement aggregation pipeline using pipeline operations(example : $sort, $ match ,$group etc in

single statement)

.ANS :=

Insert data  
db.student.insertMany([{ Rollno: 1, Name: "Riya", Class: "TE", Div: "A", Subject: "DS", Marks: 85, Address: "Pune" }, { Rollno: 2, Name: "Soham", Class: "TE", Div: "B", Subject: "DBMS", Marks: 40, Address: "Mumbai" }, { Rollno: 3, Name: "Meera", Class: "TE", Div: "B", Subject: "DS", Marks: 60, Address: "Pune" }, { Rollno: 4, Name: "Anish", Class: "TE", Div: "C", Subject: "SPOS", Marks: 70, Address: "Pune" }, { Rollno: 5, Name: "Sara", Class: "TE", Div: "A", Subject: "DS", Marks: 90, Address: "Nagpur" }, { Rollno: 6, Name: "Arjun", Class: "TE", Div: "B", Subject: "SPOS", Marks: 50, Address: "Pune" }])

1 db.student.aggregate([{ $match: { Subject: "DS" } }, { $group: { \_id: "$Address", maxMarks: { $max: "$Marks" } } }])

2

db.student.aggregate([{ $group: { \_id: null, avgMarks: { $avg: "$Marks" } } }])

3

db.student.updateMany({}, { $push: { arrayField: "newValue" } })

4

db.student.createIndex({ Name: 1 })

5

db.student.createIndex({ Class: 1, Div: 1 })

6

db.student.dropIndex("Name\_1")

7

db.student.aggregate([{ $match: { Subject: "DS" } }, { $sort: { Marks: -1 } }, { $group: { \_id: "$Div", totalMarks: { $sum: "$Marks" } } }])