**ASSIGNMENT**

**Question 1: List the names and departments of students who have more than 85% attendance and are skilled in both "MongoDB" and "Python".**

**Solution:**

db.students.find({ //Name: Kushagra Srivastava | Roll No: 1240258242

... attendance: { $gt: 85},

... skills: { $all: ["MongoDB", "Python"]}}

... )

**Output:**

A screenshot of a computer

AI-generated content may be incorrect.

**Question 2: Show all faculty who are teaching more than 2 courses. Display their names and the total number of courses they teach.**

**Solution:**

db.faculty.aggregate( [

... { $project: { \_id: 0, name: 1, totalC: { $size: "$courses" } } },

... { $match: {totalC: {$gt: 2} } }

**A screen shot of a computer

AI-generated content may be incorrect.**... ])

**Output:**

**Question 3: Write a query to show each student’s name along with the course titles they are enrolled in (use $lookup between enrollments, students, and courses).**

**Solution:**

db.enrollments.aggregate([ //Name: Kushagra Srivastava | Roll No- 1240258242

... {

... $lookup: {

... from: "students",

... localField: "student\_id",

... foreignField: "\_id",

... as: "student\_details"

... }

... },

... { $unwind: "$student\_details" },

... {

... $lookup: {

... from: "courses",

... localField: "course\_id",

... foreignField: "\_id",

... as: "course\_details"

... }

... },

... { $unwind: "$course\_details" },

... {

... $project: {

... \_id: 0,

... studentName: "$student\_details.name",

... courseTitle: "$course\_details.title"

... } }

**A screenshot of a computer

AI-generated content may be incorrect.**... ])

**Output:**

**Question 4: For each course, display the course title, number of students enrolled, and average marks (use $group).**

**Solution:**

db.enrollments.aggregate([ {

$group: {

\_id: "$course\_id",

totalStudents: { $sum: 1 },

averageMarks: { $avg: "$marks" }

} },

{

$lookup: {

from: "courses",

localField: "\_id",

foreignField: "\_id",

as: "course\_details"

} },

{ $unwind: "$course\_details" }, {

$project: {

\_id: 0,

courseTitle: "$course\_details.title",

totalStudents: 1,

averageMarks: { $round: ["$averageMarks", 2] }

} }

**A screen shot of a computer

AI-generated content may be incorrect.**] )

**Output:**

**Question 5: Find the top 3 students with the highest average marks across all enrolled courses.**

**Solution:**

db.enrollments.aggregate([ //Name: Kushagra Srivastava

{

$group: {

\_id: "$student\_id",

averageMarks: { $avg: "$marks" }

} }, {

$sort: { averageMarks: -1 }

}, {

$limit: 3

},{

$lookup: {

from: "students",

localField: "\_id",

foreignField: "\_id",

as: "student\_details"

} },

{ $unwind: "$student\_details" }, {

$project: {

\_id: 0,

studentName: "$student\_details.name",

averageMarks: { $round: ["$averageMarks", 2] }

} } ] )

**A screen shot of a computer program

AI-generated content may be incorrect.Output:**

**Question 6: Count how many students are in each department. Display the department with the highest number of students.**

**Solution:**

db.students.aggregate([ //Name: Kushagra Srivastava

{

$group: {

\_id: "$department",

totalStudents: { $sum: 1 }

} },

{

$sort: { totalStudents: -1 }

},

{

$limit: 1

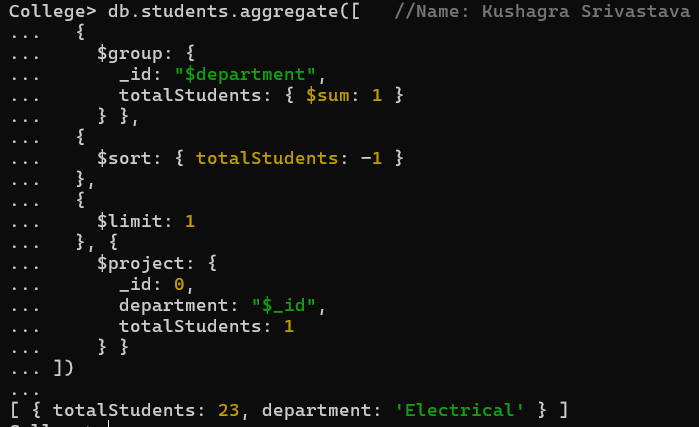
}, {

$project: {

\_id: 0,

department: "$\_id",

totalStudents: 1

**** } }

])

**Output:**

**Question 7: Update attendance to 100% for all students who won any "Hackathon".**

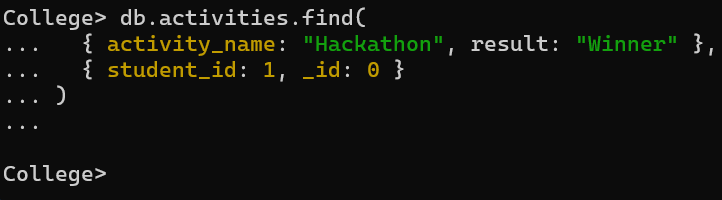
**Solution:**

db.activities.find( //Name: Kushagra Srivastava

{ activity\_name: "Hackathon", result: "Winner" },

{ student\_id: 1, \_id: 0 }

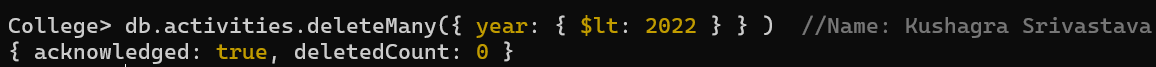
)

**Output:**

**Question 8: Delete all student activity records where the activity year is before 2022.**

**Solution:**

db.activities.deleteMany({ year: { $lt: 2022 } } ) //Name: Kushagra Srivastava

**Output: **

**Question 9: Upsert a course record for "Data Structures" with ID "C150" and credits 4—if it doesn’t exist, insert it; otherwise update its title to "Advanced Data Structures".**

**Solution:**

db.courses.updateOne( //Name: Kushagra Srivastava

{ \_id: "C150" },

{

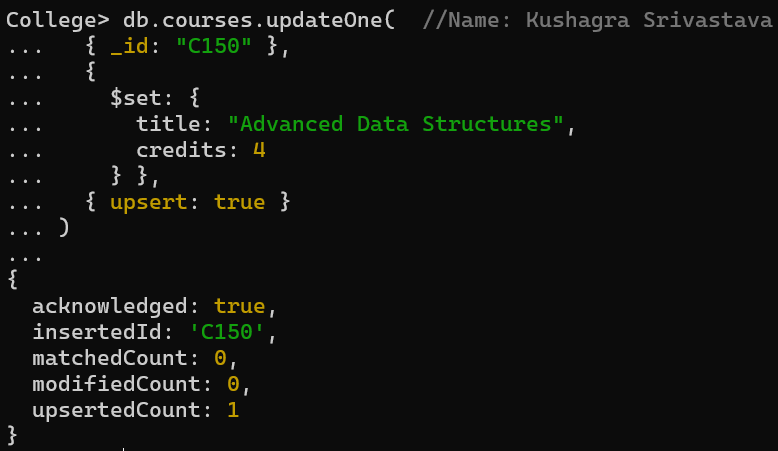
$set: {

title: "Advanced Data Structures",

credits: 4

} },

{ upsert: true }

****)

**Output:**

**Question 10** **Find all students who have "Python" as a skill but not "C++".**

**Solution:**

db.students.find( //Name: Kushagra Srivastava

{ skills: {$in: ["Python"], $nin: ["C++"] }

}

)

**A screen shot of a computer program

AI-generated content may be incorrect.**

**Output:**

**Question 11: . Return names of students who participated in "Seminar" and "Hackathon" both.**

**Solution:**

db.activities.find(

{ activities: { $all: ["Seminar", "Hackathon"] } } )

**A black screen with white text and green text

AI-generated content may be incorrect.Output:**

**Question 12: Find students who scored more than 80 in "Web Development" only if they belong to the "Computer Science" department.**

**Solution:**

db.students.find({ //Name: Kushagra Srivastava

department: "Computer Science",

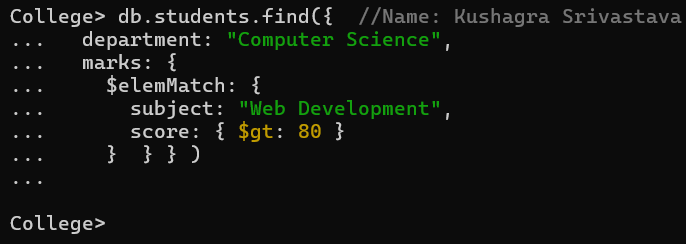
marks: {

$elemMatch: {

subject: "Web Development",

score: { $gt: 80 }

} } } )

**Output:**

**Question 13: For each faculty member, list the names of all students enrolled in their courses along with average marks per student per faculty.**

**Solution:**

db.faculty.aggregate([ //Name: Kushagra Srivastava

{

$lookup: {

from: "courses",

localField: "\_id",

foreignField: "faculty\_id",

as: "courses\_taught"

} },

{ $unwind: "$courses\_taught" },

{

$lookup: {

from: "enrollments",

localField: "courses\_taught.\_id",

foreignField: "course\_id",

as: "student\_enrollments"

} },

{ $unwind: "$student\_enrollments" },

{

$lookup: {

from: "students",

localField: "student\_enrollments.student\_id",

foreignField: "\_id",

as: "student\_details"

} },

{ $unwind: "$student\_details" },

{

$group: {

\_id: {

facultyName: "$name",

studentName: "$student\_details.name"

},

averageMarks: { $avg: "$student\_enrollments.marks" }

} },

{

$project: {

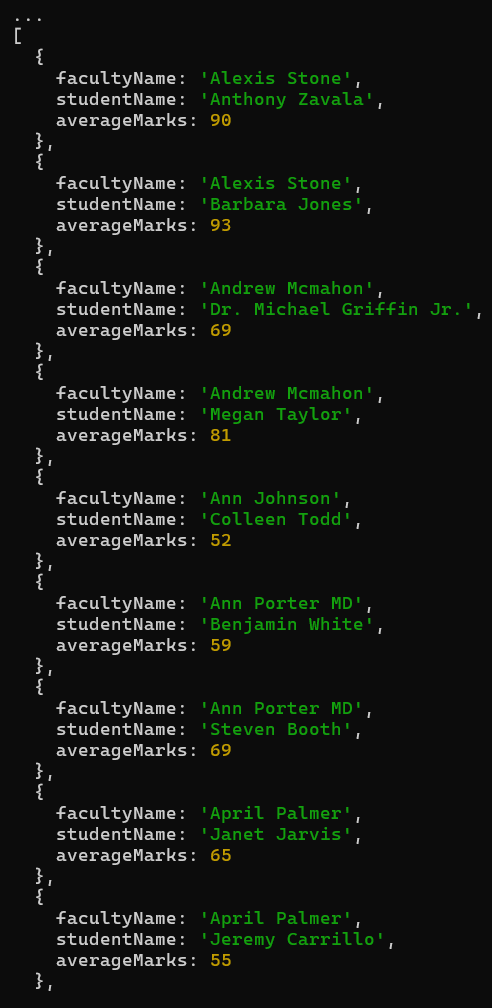
\_id: 0,

facultyName: "$\_id.facultyName",

studentName: "$\_id.studentName",

averageMarks: { $round: ["$averageMarks", 2] }

} },

**** { $sort: { facultyName: 1, studentName: 1 } } ] )

**A screen shot of a computer program

AI-generated content may be incorrect.Output:**

**Question 14: Show the most popular activity type (e.g., Hackathon, Seminar, etc.) by number of student participants.**

**Solution:**

db.activities.aggregate( [ //Name: Kushagra Srivastava

{ //Roll No- 1240248242

$group: {

\_id: "$type",

totalParticipants: { $sum: 1 }

} },

{ $sort: { totalParticipants: -1 } },

{ $limit: 1 },

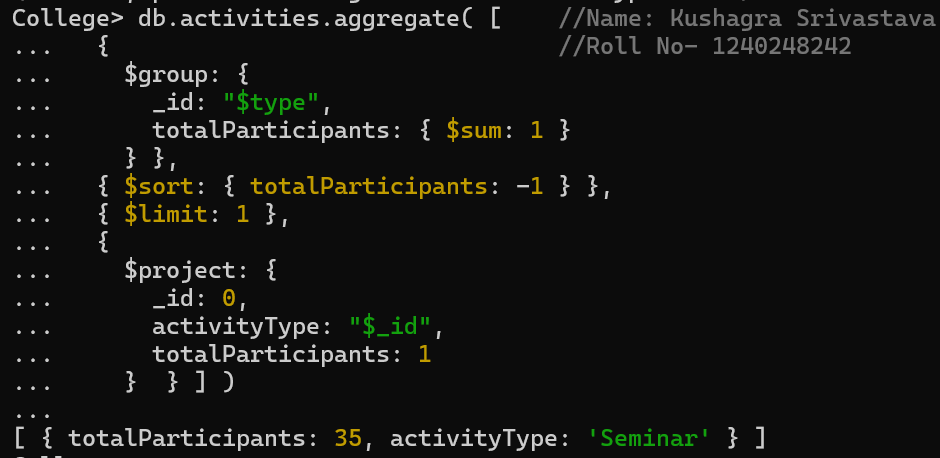
{

$project: {

\_id: 0,

activityType: "$\_id",

totalParticipants: 1

**** } } ] )

**Output:**