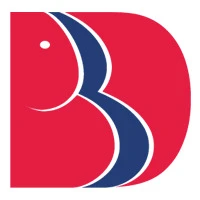
BABU BANARASI DAS UNIVERSITY



SESSION:2025-26

Name: Naincy Saroj

Roll NO: 58

Program: BCA (DS&AI)

Batch: BCADS24

Semester: 3rd

Subject Teacher: Ankit Verma

Signature: -

INDEX

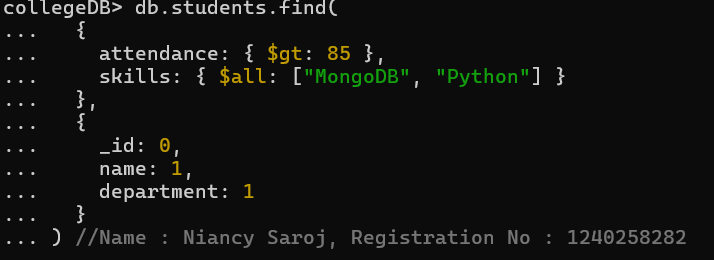
|  |  |  |  |
| --- | --- | --- | --- |
| Sr.no |  | Faculty Signature | Remarks |
| 1 | **Complex Filters &Projections** |  |  |
| 2 | **Joins ($lookup) and Aggregations** |  |  |
| 3 | **Grouping, Sorting, and Limiting** |  |  |
| 4 | **Update, Upsert, and Delete** |  |  |
| 5 | **Array & Operator Usage** |  |  |
| 6 | **Subdocuments and Nested Conditions** |  |  |
| 7 | **Advanced Aggregation**  **(Challenge Level)** |  |  |

ASSESSMENT TASK

**1.Complex Filters & Projections**

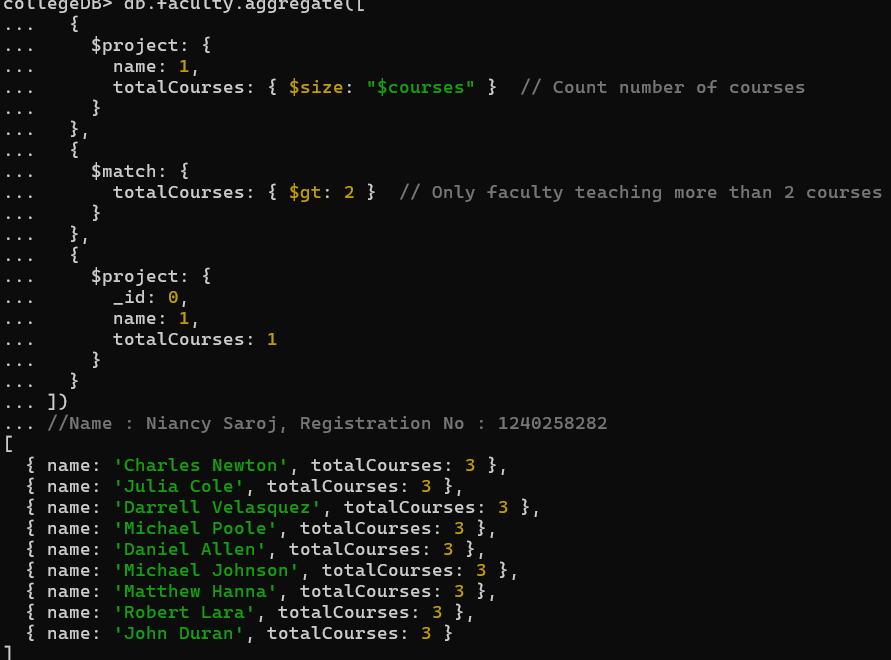
Q1. List the names and departments of students who have more than 85% attendance and are skilled in both "MongoDB" and "Python".

**Output:**



* attendance: { $gt: 85 } → selects students with attendance greater than 85%.
* skills: {$all: ["MongoDB", "Python"] } → selects students who have both "MongoDB" and "Python" in their skills array.
* In the projection part, specify which fields to display:
* { name: 1, department: 1, \_id: 0 } → shows only name and department, hides the \_id field.

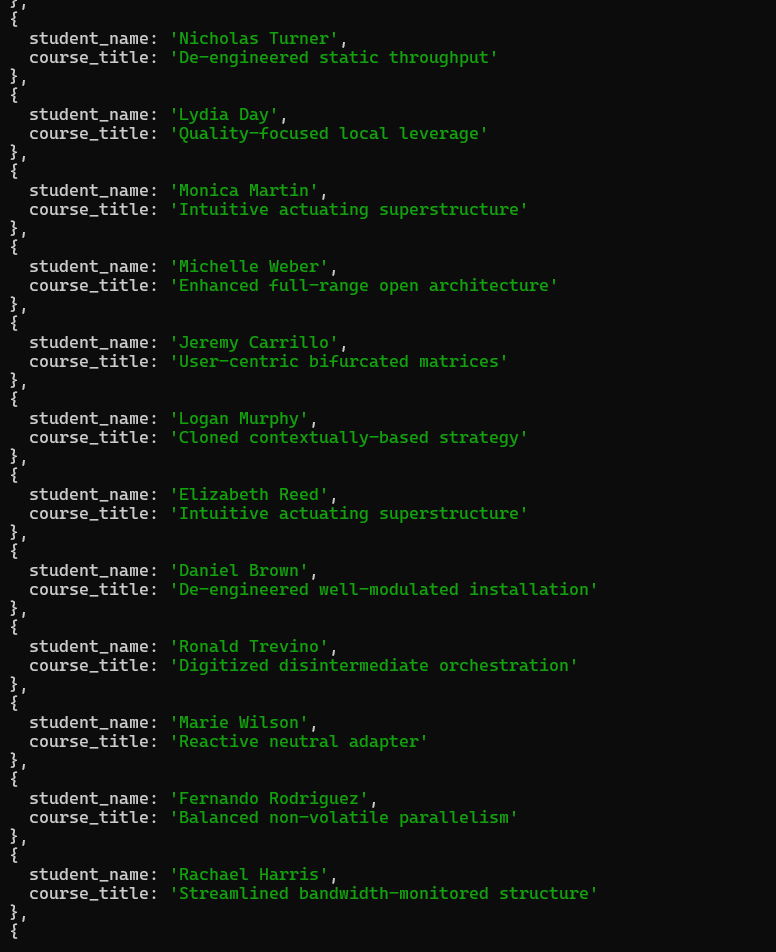
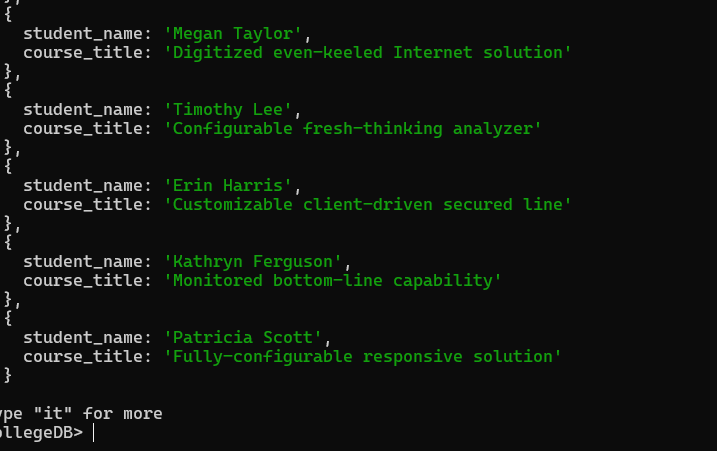
Q2. Show all faculty who are teaching more than 2 courses. Display their names and the total number of courses they teach.



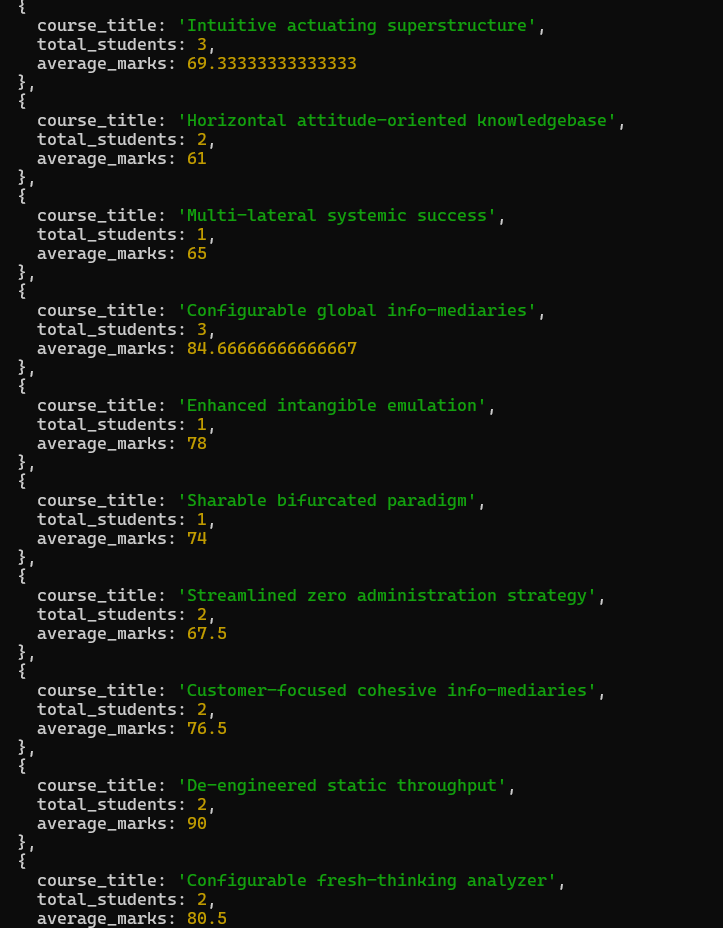
* $size: "$courses" → counts how many courses each faculty teaches
* Use $match to filter faculty members who are teaching more than 2 courses, using total Courses: { $gt: 2 }.
* Use $project to display only the fields needed in output: name and totalCourses, hide \_id.

**2. Joins ($lookup) and Aggregations**

Q3. 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).

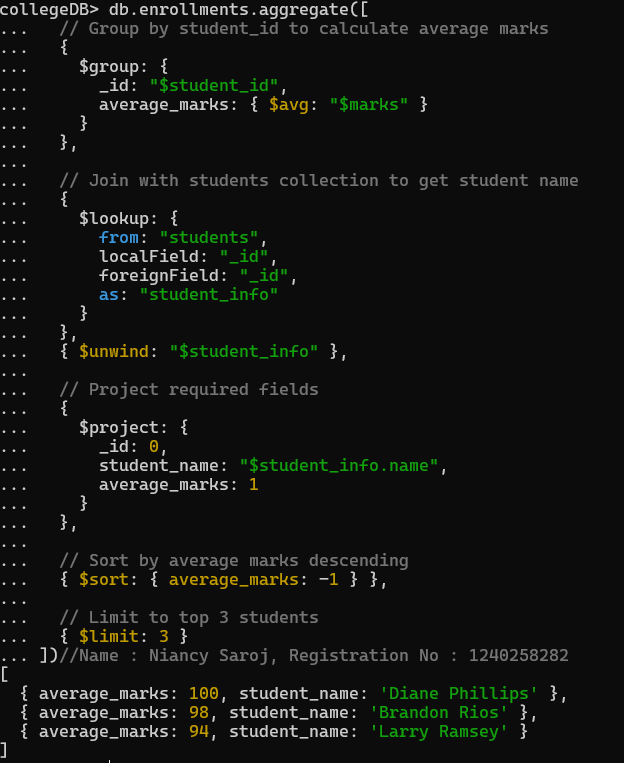
* Use $project to show only what we need:
* Student’s name → from student\_info.name
* Course title → from course\_info.title
* Hide \_id so output looks clean.

Q4. For each course, display the course title, number of students enrolled, and average marks (use $group) 

* total \_students → counts how many students are enrolled using $sum: 1
* avg\_marks → finds average marks using $avg: "$marks".
* Use $lookup to join this data with the courses collection.

3**.Grouping, Sorting, and Limiting**

Q5. Find the top 3 students with the highest average marks across all enrolled courses

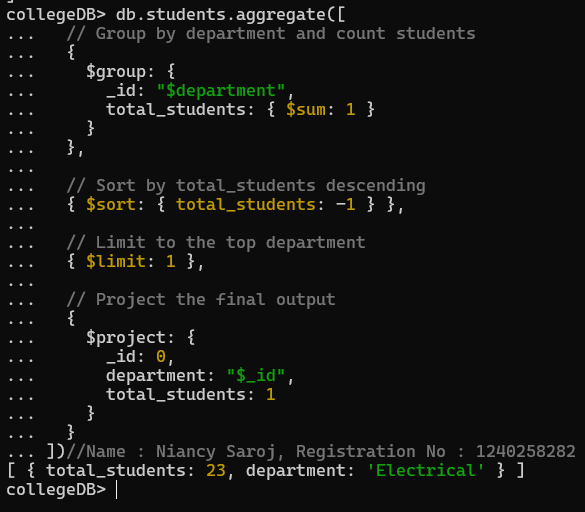


* Use $group to collect marks of each student by student\_id.
* Use $avg: "$marks" to calculate average marks for each student.
* Use $lookup to join with the students collection.

Q6. Count how many students are in each department. Display the department with the highest number of students.

Output:

* total\_students: { $sum: 1 } → counts how many students are there in each department
* Use $sort: { total\_students: -1 } to arrange the results in descending order (highest first).



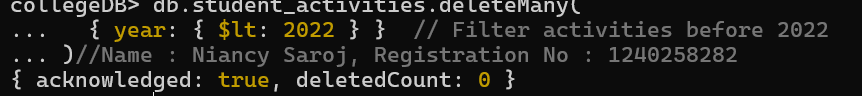
**4.Update, Upsert, and Delete**

Q7. Update attendance to 100% for all students who won any "Hackathon".



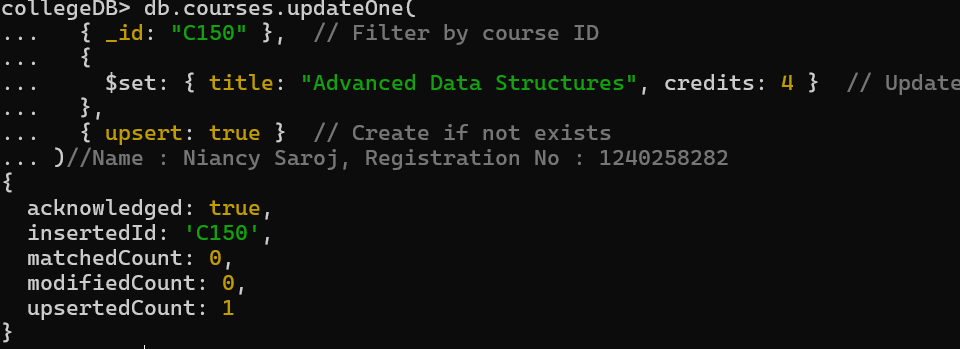
* The filter condition is { "activities.name": "Hackathon" }
* { $set: { attendance: 100 } } → sets attendance to 100%.

Q8. Delete all student activity records where the activity year is before 2022

.

* The filter condition is { year: { $lt: 2022 } }
* This selects all activity records where the year field is less than 20
* Run the query - all old activity records will be removed.

Q9. 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".



* Filter: { \_id: "C150" } → finds the course with ID "C150".
* Update Part: { $set: { title: "Advanced Data Structures", credits: 4 } }
* Upsert Option: { upsert: true }

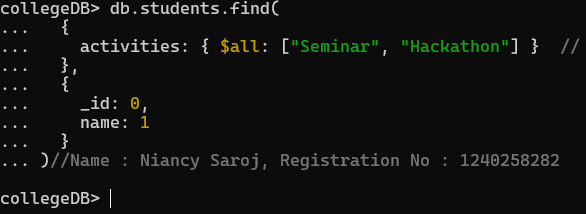
**5. Array & Operator Usage**

Q10. Find all students who have "Python" as a skill but not "C++".

* skills: "Python" → selects students who have "Python" in their skills array.
* skills: { $ne: "C++" } → ensures the student does not have "C++" in their skills array.

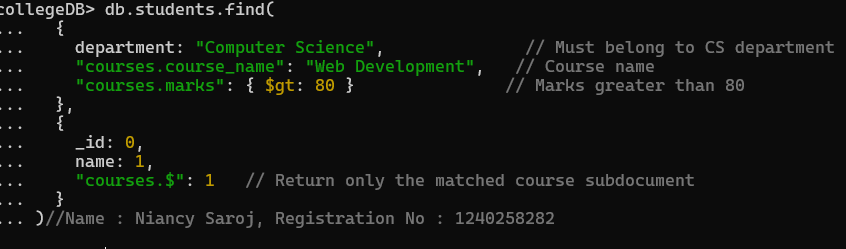
Q11. Return names of students who participated in "Seminar" and "Hackathon" both.



* Condition: activities: { $all: ["Seminar", "Hackathon"] }
* $all ensures the student has both "Seminar" and "Hackathon" in their activities array.
* Projection: { \_id: 0, name: 1 }

**6.Subdocuments and Nested Conditions**

Q12. Find students who scored more than 80 in "Web Development" only if they belong to the "Computer Science" department.



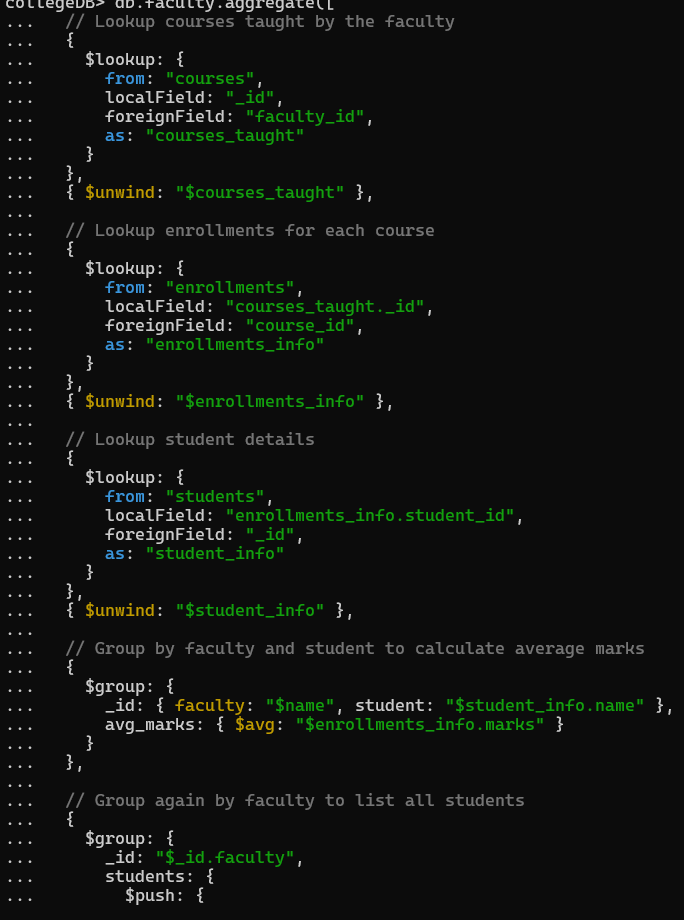
"courses.course\_name": "Web Development" → ensures the student has that course.

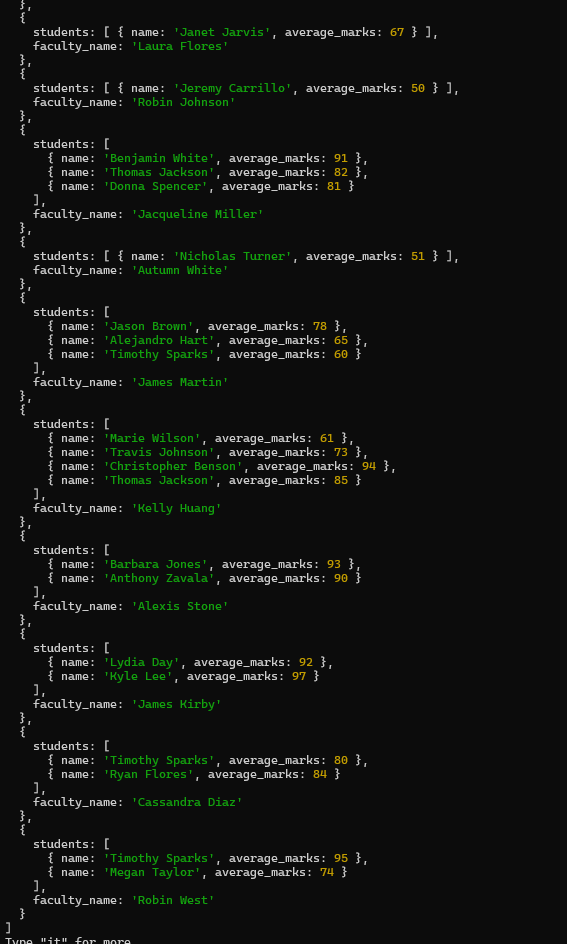
"courses.marks": { $gt: 80 } → ensures marks in that course are greater than 80.

Projection: { \_id: 0, name: 1, "course

**7. Advanced Aggregation (Challenge Level)**

Q13. For each faculty member, list the names of all students enrolled in their courses along with average marks per student per faculty.



Q14. Show the most popular activity type (e.g., Hackathon, Seminar, etc.) by number of student participants.

