

BABU BANARSI DAS UNIVERSITY



NO SQL and DBAAS 101
(BCADS N13202)

Project

Submitted To:

Mr. Ankit Verma

Submitted by:

Name: Rajat Mishra

Roll no: 1240258350

Class: BCADS 25

Project

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

Query:

```
db.students.find({attendance:{$gt:85},skills:{$all:
["MongoDB","Python"]}}, {name:1,department:1,_id:
0})
```

Output:

```
students_full
rajatmishra> db.students.find({attendance:{$gt:85},skills:{$all:
... ["MongoDB","Python"]}}, {name:1,department:1,_id:
... 0})

rajatmishra> //Rajat Mishra 1240258350
```

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

Query:

```
db.faculty_full.aggregate([{$project: {_id:0,name:
1,total_courses: { $size: "$courses" } }},{$match:
{total_courses: { $gt: 2 } } }])
```

Output:

```
rajatmishra> db.faculty_full.aggregate([{$project: {_id:0,name:
... 1,total_courses: { $size: "$courses" } }},{$match:
... {total_courses: { $gt: 2 } } }])
[
  { name: 'Charles Newton', total_courses: 3 },
  { name: 'Julia Cole', total_courses: 3 },
  { name: 'Darrell Velasquez', total_courses: 3 },
  { name: 'Michael Poole', total_courses: 3 },
  { name: 'Matthew Hanna', total_courses: 3 },
  { name: 'John Duran', total_courses: 3 },
  { name: 'Daniel Allen', total_courses: 3 },
  { name: 'Michael Johnson', total_courses: 3 },
  { name: 'Robert Lara', total_courses: 3 }
]
rajatmishra> //Rajat Mishra 1240258350
```

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).

Query:

```
db.enrollments.aggregate([ { $lookup: { from:
"students", localField: "student_id", foreignField:
"_id", as: "student" } }, { $unwind: "$student" }, {
$lookup: { from: "courses", localField:
"course_id", foreignField: "_id", as: "course" } }, {
$unwind: "$course" }, { $project: { _id: 0,
student_name: "$student.name",
course_title: "$course.title" } } ] )
```

Output:

```
[ { course_title: 'Reactive neutral adapter'
  {
    student_name: 'Alexandra Bailey',
    course_title: 'Reactive neutral adapter'
  }, course_title: 'Sharable bifurcated paradigm'
  {
    student_name: 'Megan Taylor',
    course_title: 'Sharable bifurcated paradigm'
  }, course_title: 'Focused user-facing paradigm'
  {
    student_name: 'Alejandro Hart',
    course_title: 'Focused user-facing paradigm'
  }, course_title: 'Focused user-facing paradigm'
  {
    student_name: 'Timothy Sparks',
    course_title: 'Focused user-facing paradigm'
  }, course_title: 'Balanced asynchronous framework'
  {
    student_name: 'Juan Morris',
    course_title: 'Balanced asynchronous framework'
  }, course_title: 'Organic optimal product'
  {
    student_name: 'Donna Morgan',
    course_title: 'Organic optimal product'
  }, course_title: 'Fully-configurable responsive solution'
  {
    student_name: 'Patricia Scott',
    course_title: 'Fully-configurable responsive solution'
  }, course_title: 'Horizontal attitude-oriented knowledgebase'
  {
    student_name: 'Carolyn Chandler',
    course_title: 'Horizontal attitude-oriented knowledgebase'
```

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

Query:

```
db.courses.aggregate([ { $lookup: { from:
  "enrollments", localField: "_id", foreignField:
  "course_id", as: "enrolled_students" } }, {
  $project: { _id:0,title: 1, number_of_students: {
    $size: "$enrolled_students" }, average_marks:
    { $avg: "$enrolled_students.marks" } } } ] );
```

Output:

```
[
  {
    title: 'Profit-focused high-level capability',
    number_of_students: 2,
    average_marks: 58.5
  },
  {
    title: 'Multi-lateral systemic success',
    number_of_students: 1,
    average_marks: 65
  },
  {
    title: 'Reduced 5thgeneration complexity',
    number_of_students: 0,
    average_marks: null
  },
  {
    title: 'Enhanced real-time website',
    number_of_students: 0,
    average_marks: null
  },
  {
    title: 'Streamlined scalable policy',
    number_of_students: 2,
    average_marks: 71.5
  },
  {
    title: 'Integrated fault-tolerant task-force',
    number_of_students: 2,
```

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

Query:

```
db.enrollments.aggregate([ { $lookup: { from:
  "students", localField: "student_id", foreignField:
  "_id", as: "studentInfo" } }, { $unwind:
  "$studentInfo" }, { $group: { _id:
  "$student_id", name: { $first:
  "$studentInfo.name" }, department: { $first:
  "$studentInfo.department" }, avgMarks: { $avg:
  "$marks" } } }, { $sort: { avgMarks: -1 } }, {
  $limit: 3 } ] )
```

Output:

```
[
  {
    _id: 'S080',
    name: 'Diane Phillips',
    department: 'Civil',
    avgMarks: 100
  },
  {
    _id: 'S046',
    name: 'Brandon Rios',
    department: 'Biotechnology',
    avgMarks: 98
  },
  {
    _id: 'S068',
    name: 'Larry Ramsey',
    department: 'Biotechnology',
    avgMarks: 94
  }
]
```

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

Query:

```
db.students.aggregate([ { $group: { _id:
"$department", totalStudents: { $sum: 1 } } }, {
$sort: { totalStudents: -1 } }, { $limit: 1 } ] )
```

Output:

```
[ { _id: 'Electrical', totalStudents: 23 } ]
```

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

Query:

```
db.students.updateMany( { awards: {  
  $elemMatch: { name: "Hackathon", winner: true }  
}}, { $set: { attendance: 100 } } );
```

Output:

```
rajatmishra> db.students.updateMany( { awards: {  
... $elemMatch: { name: "Hackathon", winner:  
... true } } }, { $set: { attendance: 100 } } ); // Rajat Mishra 1240258350  
{  
  acknowledged: true,  
  insertedId: null,  
  matchedCount: 0,  
  modifiedCount: 0,  
  upsertedCount: 0  
}  
rajatmishra> █
```


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

Query:

```
db.activities.deleteMany({ activityYear: { $lt: 2022 } })
```

Output:

```
/
rajatmishra> db.activities.deleteMany({ activityYear: { $lt:
... 2022 } }) // Rajat Mishra 1240258350
{ acknowledged: true, deletedCount: 0 }
rajatmishra> █
```

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".

Query:

```
db.courses.updateOne( { _id:"C150" }, { $set: {  
  title: "Advanced Data structures", credits: 4 } },  
  { upsert: true } );
```

Output:

```
rajatmishra> db.courses.updateOne( { _id:"C150" }, { $set: {  
... title: "Advanced Data structures", credits: 4 } },  
... { upsert: true } ); // Rajat Mishra 1240258350  
{  
  acknowledged: true,  
  insertedId: 'C150',  
  matchedCount: 0,  
  modifiedCount: 0,  
  upsertedCount: 1  
}  
rajatmishra>
```

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

Query:

```
db.students.find({ skills: { $in: ["Python"], $nin: ["C++"] } })
```

Output:

```
[
  {
    _id: 'S004',
    name: 'Kyle Hale',
    dob: '2000-10-20',
    department: 'Electrical',
    skills: [ 'Python', 'Java' ],
    attendance: 79.78
  },
  {
    _id: 'S008',
    name: 'Cody Whitehead',
    dob: '2003-11-25',
    department: 'Biotechnology',
    skills: [ 'JavaScript', 'Python' ],
    attendance: 92.03
  },
  {
    _id: 'S009',
    name: 'Thomas Jackson',
    dob: '2002-10-25',
    department: 'Electrical',
    skills: [ 'Python', 'AutoCAD' ],
    attendance: 96.64
  }
]
```

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

Query:

```
db.students.find( { participation: { $all:
  ["Seminar", "Hackathon"] } }, { name: 1, _id: 0 } );
```

Output:

```
rajatmishra> db.students.find( { participation: { $all:
... ["Seminar", "Hackathon"] } }, { name: 1, _id: 0 } ); // Rajat Mishra 1240258350
rajatmishra> █
```

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

Query:

```
db.enrollments.aggregate([ { $match: { marks: {
$gt: 80 } } }, { $lookup: { from: "students",
localField: "student_id", foreignField: "_id", as:
"student_info" } }, { $unwind: "$student_info" }, {
$match: { "student_info.department": "Computer
Science" } }, { $project: { _id: 0, name:
"$student_info.name", department:
"$student_info.department", marks: 1 } } ] );
```

Output:

```
[
  {
    marks: 100,
    name: 'Nicholas Turner',
    department: 'Computer Science'
  },
  { marks: 82, name: 'Mary Bennett', department: 'Computer Science' },
  { marks: 82, name: 'Bruce Blair', department: 'Computer Science' },
  {
    marks: 91,
    name: 'Michelle Walters',
    department: 'Computer Science'
  },
  { marks: 91, name: 'David Rivera', department: 'Computer Science' },
  { marks: 81, name: 'Donna Spencer', department: 'Computer Science' }
]
```

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

Query:

```
db.faculty.aggregate([ { $unwind: "$courses" }, {
  $lookup: { from: "courses", localField: "courses",
    foreignField: "_id", as: "course_info" } }, { $unwind:
    "$course_info" }, { $lookup: { from: "enrollments",
      localField: "course_info._id", foreignField:
        "course_id", as: "enrollment_info" } }, { $unwind:
          "$enrollment_info" }, { $lookup: { from: "students",
            localField: "enrollment_info.student_id", foreignField:
              "_id", as: "student_info" } }, { $unwind: "$student_info"
            }, { $group: { _id: { faculty_name: "$name",
              student_name: "$student_info.name" },
              average_marks: { $avg: "$enrollment_info.marks" } } },
            { $group: { _id: "$_id.faculty_name", students: {
              $push: { name: "$_id.student_name", average_marks:
                "$average_marks" } } } }, { $replaceRoot: { newRoot: {
              faculty_name: "$_id", students: "$students" } } } ] );
```

Output:

```
[
  {
    faculty_name: 'Joshua Reyes',
    students: [
      { name: 'Bruce Blair', average_marks: 82 },
      { name: 'Chelsey Davis', average_marks: 54 }
    ]
  },
  {
    faculty_name: 'Jessica Black',
    students: [ { name: 'Bruce Blair', average_marks: 78 } ]
  },
  {
    faculty_name: 'Ann Porter MD',
    students: [
      { name: 'Benjamin White', average_marks: 59 },
      { name: 'Steven Booth', average_marks: 69 }
    ]
  }
]
```

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

Query:

```
db.activities.aggregate([ { $addFields: {  
  participants_array: { $cond: { if: { $isArray:  
    "$participants" }, then: "$participants", else: {  
      $cond: { if: { $isArray: "$student_ids" }, then:  
        "$student_ids", else: { $cond: { if: { $isArray:  
          "$students" }, then: "$students", else: [] } } } } }  
    }, activity_name: { $ifNull: ["$activity_type", {  
      $ifNull: ["$type", "$activityName"] } } } } }, {  
    $project: { activity_name: 1, participant_count: {  
      $size: "$participants_array" } } }, { $group: {  
      _id: "$activity_name", total_participants: { $sum:  
        "$participant_count" } } }, { $sort: {  
        total_participants: -1 } }, { $limit: 1 }, { $project:  
        { _id: 0, activity_type: "$_id", total_participants:  
          1 } } } ] );
```


Output:

```
rajatmishra> db.activities.aggregate([ { $addFields: {  
... participants_array: { $cond: { if: { $isArray:  
... "$participants" }, then: "$participants", else: {  
... $cond: { if: { $isArray: "$student_ids" }, then:  
... "$student_ids", else: { $cond: { if: { $isArray:  
... "$students" }, then: "$students", else: [] } } } } }  
... }, activity_name: { $ifNull: ["$activity_type", {  
... $ifNull: ["$type", "$activityName"] } ] } }, {  
... $project: { activity_name: 1, participant_count: {  
... $size: "$participants_array" } } }, { $group: {  
... _id: "$activity_name", total_participants: {  
... $sum: "$participant_count" } } }, { $sort: {  
... total_participants: -1 } }, { $limit: 1 }, { $project:  
... { _id: 0, activity_type: "$_id", total_participants:  
... 1 } } ] ]; // Rajat Mishra 1240258350
```