PROJECT – 01

NAME – Garima Dwivedi

BATCH - BCADS23

ROLL NO – 1240258179

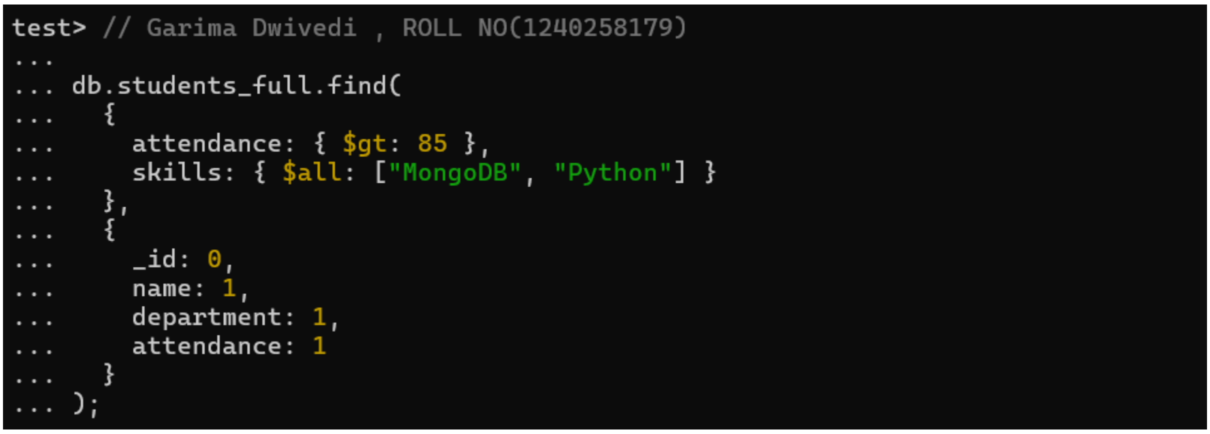
->Write a query for each questions-

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 ".

QUERY:

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



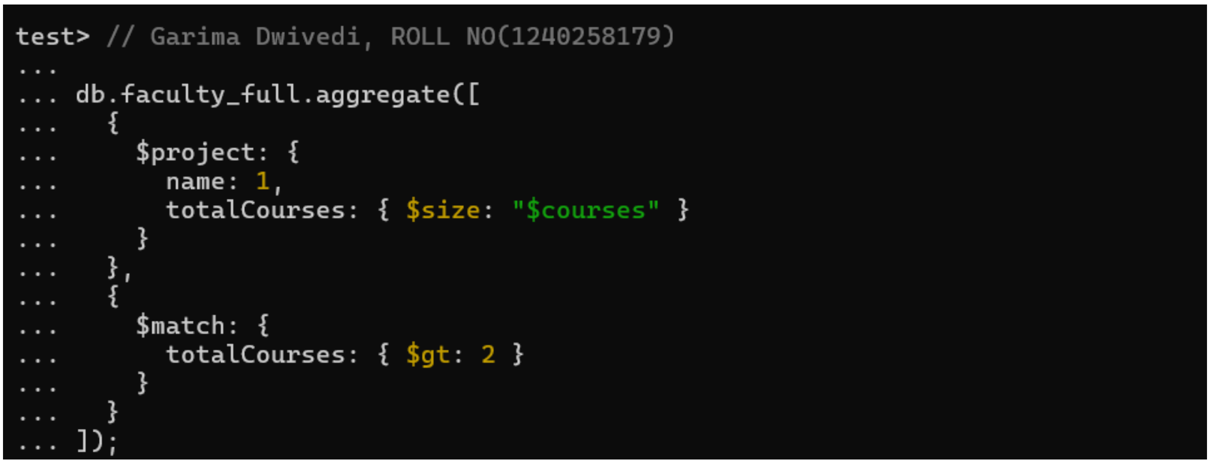
```
test> // Garima Dwivedi , ROLL NO(1240258179)
...
... db.students_full.find(
...   {
...     attendance: { $gt: 85 },
...     skills: { $all: ["MongoDB", "Python"] }
...   },
...   {
...     _id: 0,
...     name: 1,
...     department: 1,
...     attendance: 1
...   }
... );
```

(output was not found as there was no student with the sprcified data)

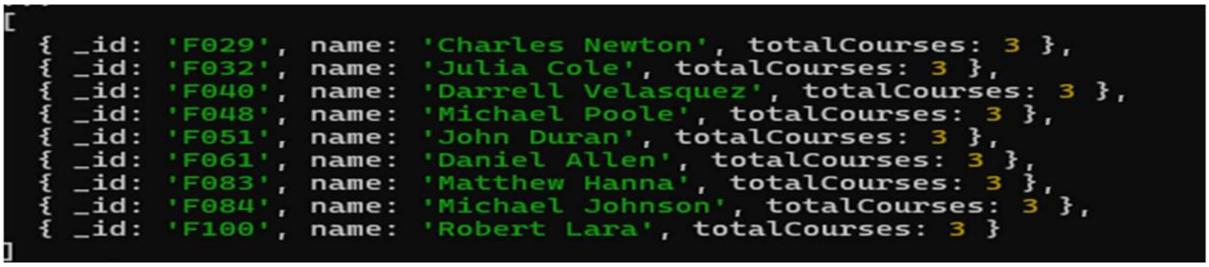
Q2. Show all faculty who are teaching more than 2 courses. Display their names and the total number of courses they teach.

QUERY:

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



```
test> // Garima Dwivedi, ROLL NO(1240258179)
...
... db.faculty_full.aggregate([
...   {
...     $project: {
...       name: 1,
...       totalCourses: { $size: "$courses" }
...     }
...   },
...   {
...     $match: {
...       totalCourses: { $gt: 2 }
...     }
...   }
... ]);
```



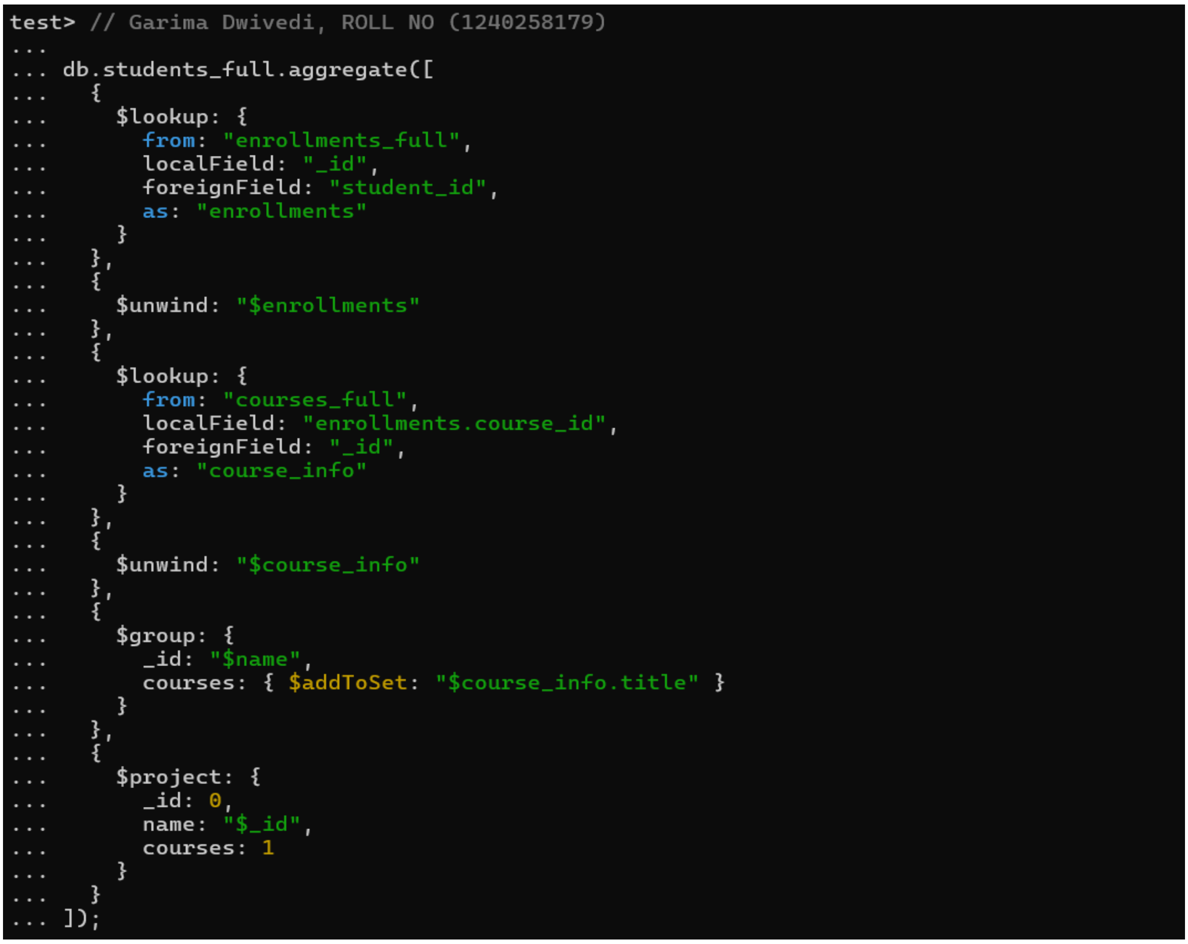
```
[
  { _id: 'F029', name: 'Charles Newton', totalCourses: 3 },
  { _id: 'F032', name: 'Julia Cole', totalCourses: 3 },
  { _id: 'F040', name: 'Darrell Velasquez', totalCourses: 3 },
  { _id: 'F048', name: 'Michael Poole', totalCourses: 3 },
  { _id: 'F051', name: 'John Duran', totalCourses: 3 },
  { _id: 'F061', name: 'Daniel Allen', totalCourses: 3 },
  { _id: 'F083', name: 'Matthew Hanna', totalCourses: 3 },
  { _id: 'F084', name: 'Michael Johnson', totalCourses: 3 },
  { _id: 'F100', name: 'Robert Lara', totalCourses: 3 }
]
```

2. Joins ($lookup) and Aggregations

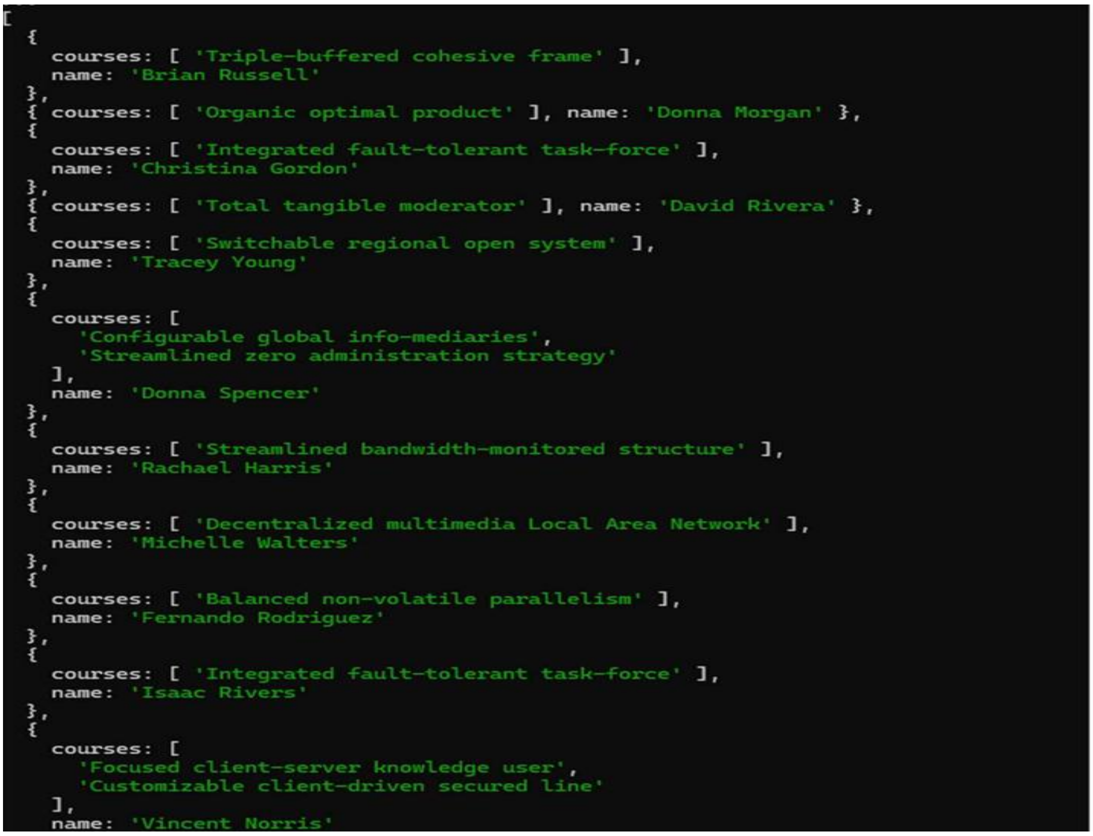
Q3. Write a query to show each student’s name along with the course titles they are enrolled in (use $lookup between enrolments, students, and courses).

QUERY:

```
db.students_full.aggregate([
  {
    $lookup: {
      from: "enrollments_full",
      localField: "_id",
      foreignField: "student_id",
      as: "enrollments"
    }},
  {
    $unwind: "$enrollments"
  },
  {
    $lookup: {
      from: "courses_full",
      localField: "enrollments.course_id",
      foreignField: "_id",
      as: "course_info"
    }},
  {
    $unwind: "$course_info"
  },
  {
    $group: {
      _id: "$name",
      courses: { $addToSet: "$course_info.title" }
    }},
  {
    $project: {
      _id: 0,
      name: "$_id",
      courses: 1
    }
  }]);
```



```
test> // Garima Dwivedi, ROLL NO (1240258179)
...
... db.students_full.aggregate([
...   {
...     $lookup: {
...       from: "enrollments_full",
...       localField: "_id",
...       foreignField: "student_id",
...       as: "enrollments"
...     }
...   },
...   {
...     $unwind: "$enrollments"
...   },
...   {
...     $lookup: {
...       from: "courses_full",
...       localField: "enrollments.course_id",
...       foreignField: "_id",
...       as: "course_info"
...     }
...   },
...   {
...     $unwind: "$course_info"
...   },
...   {
...     $group: {
...       _id: "$name",
...       courses: { $addToSet: "$course_info.title" }
...     }
...   },
...   {
...     $project: {
...       _id: 0,
...       name: "$_id",
...       courses: 1
...     }
...   }
... ]);
```



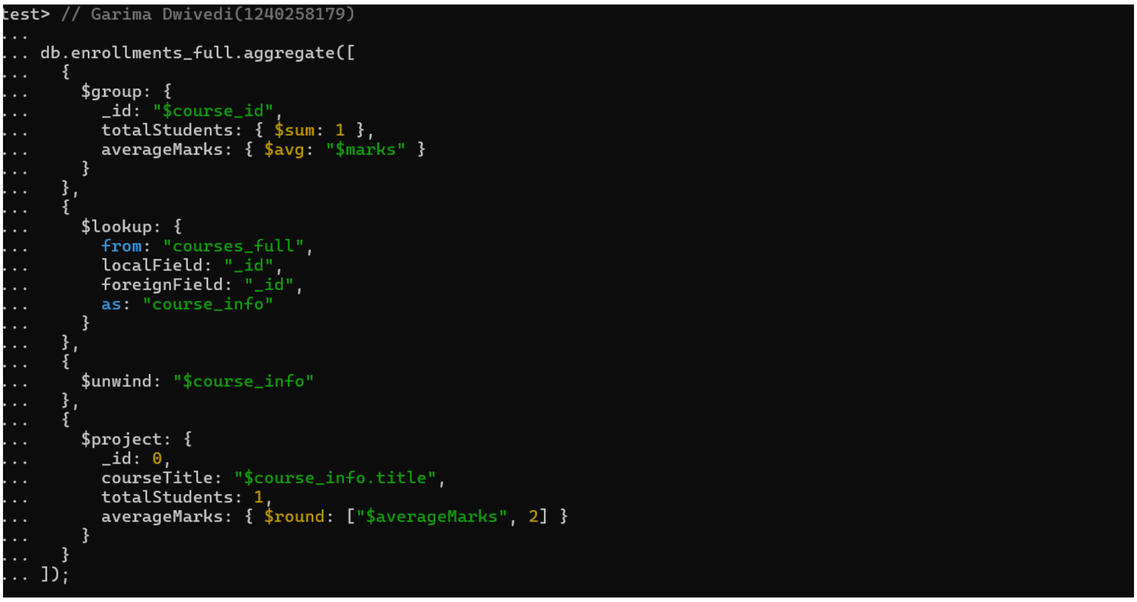
```
[
  {
    courses: [ 'Triple-buffered cohesive frame' ],
    name: 'Brian Russell'
  },
  { courses: [ 'Organic optimal product' ], name: 'Donna Morgan' },
  {
    courses: [ 'Integrated fault-tolerant task-force' ],
    name: 'Christina Gordon'
  },
  { courses: [ 'Total tangible moderator' ], name: 'David Rivera' },
  {
    courses: [ 'Switchable regional open system' ],
    name: 'Tracey Young'
  },
  {
    courses: [
      'Configurable global info-mediaries',
      'Streamlined zero administration strategy'
    ],
    name: 'Donna Spencer'
  },
  {
    courses: [ 'Streamlined bandwidth-monitored structure' ],
    name: 'Rachael Harris'
  },
  {
    courses: [ 'Decentralized multimedia Local Area Network' ],
    name: 'Michelle Walters'
  },
  {
    courses: [ 'Balanced non-volatile parallelism' ],
    name: 'Fernando Rodriguez'
  },
  {
    courses: [ 'Integrated fault-tolerant task-force' ],
    name: 'Isaac Rivers'
  },
  {
    courses: [
      'Focused client-server knowledge user',
      'Customizable client-driven secured line'
    ],
    name: 'Vincent Norris'
```

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

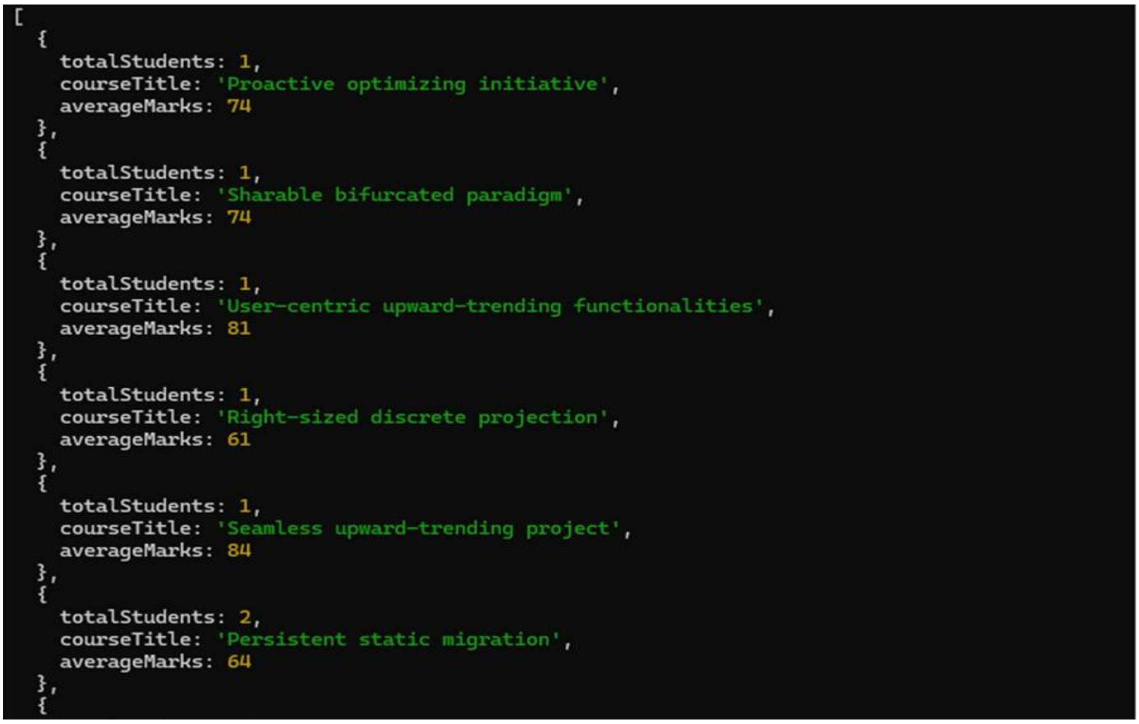
QUERY:

```
db.enrollments_full.aggregate([
  { $group: {
      _id: "$course_id",
      totalStudents: { $sum: 1 },
      averageMarks: { $avg: "$marks" }
    }
  },
  {
   $lookup: {
      from: "courses_full",
      localField: "_id",
      foreignField: "_id",
      as: "course_info"
```

```
 }},
{ $unwind: "$course_info"},
{$project: {
   _id: 0,
   courseTitle: "$course_info.title",
   totalStudents: 1,
   averageMarks: { $round: ["$averageMarks", 2] }
 }}]);
```



```
test> // Garima Dwivedi(1240258179)
...
... db.enrollments_full.aggregate([
...   {
...     $group: {
...       _id: "$course_id",
...       totalStudents: { $sum: 1 },
...       averageMarks: { $avg: "$marks" }
...     }
...   },
...   {
...     $lookup: {
...       from: "courses_full",
...       localField: "_id",
...       foreignField: "_id",
...       as: "course_info"
...     }
...   },
...   {
...     $unwind: "$course_info"
...   },
...   {
...     $project: {
...       _id: 0,
...       courseTitle: "$course_info.title",
...       totalStudents: 1,
...       averageMarks: { $round: ["$averageMarks", 2] }
...     }
...   }
... ]);
```



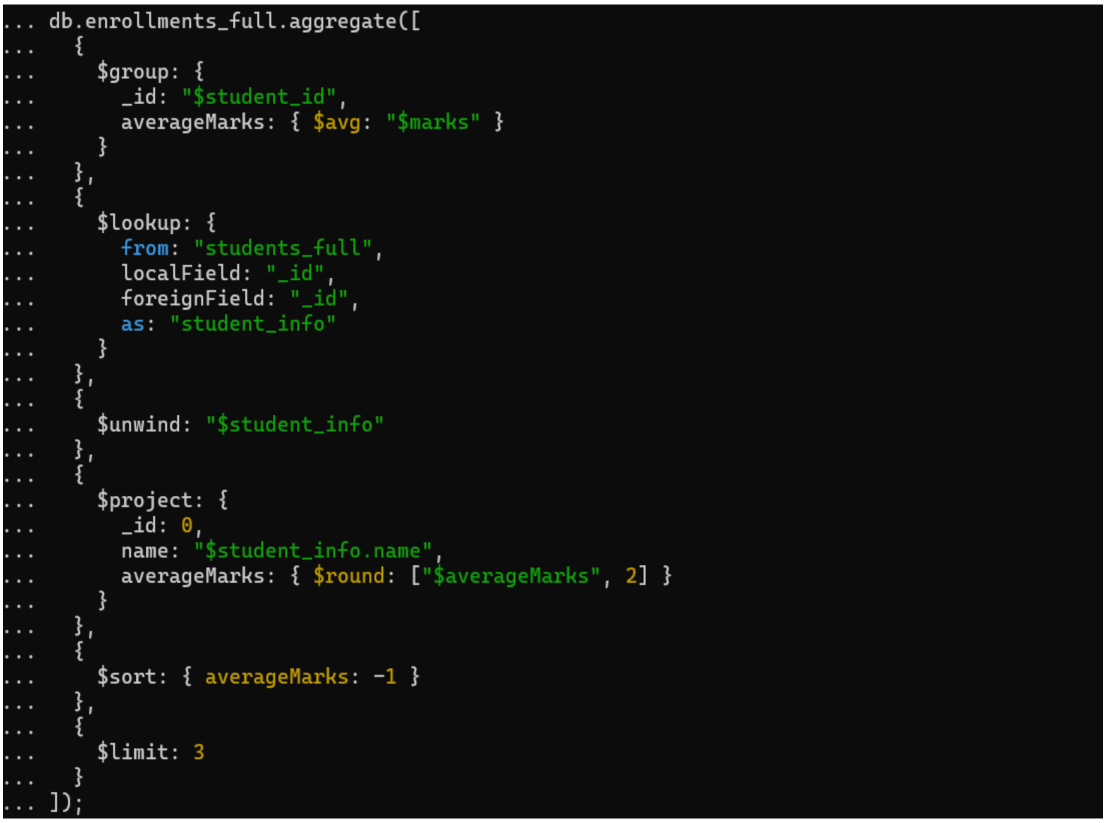
```
[
  {
    totalStudents: 1,
    courseTitle: 'Proactive optimizing initiative',
    averageMarks: 74
  },
  {
    totalStudents: 1,
    courseTitle: 'Sharable bifurcated paradigm',
    averageMarks: 74
  },
  {
    totalStudents: 1,
    courseTitle: 'User-centric upward-trending functionalities',
    averageMarks: 81
  },
  {
    totalStudents: 1,
    courseTitle: 'Right-sized discrete projection',
    averageMarks: 61
  },
  {
    totalStudents: 1,
    courseTitle: 'Seamless upward-trending project',
    averageMarks: 84
  },
  {
    totalStudents: 2,
    courseTitle: 'Persistent static migration',
    averageMarks: 64
  },
  {
```

3. Grouping, Sorting, and Limiting

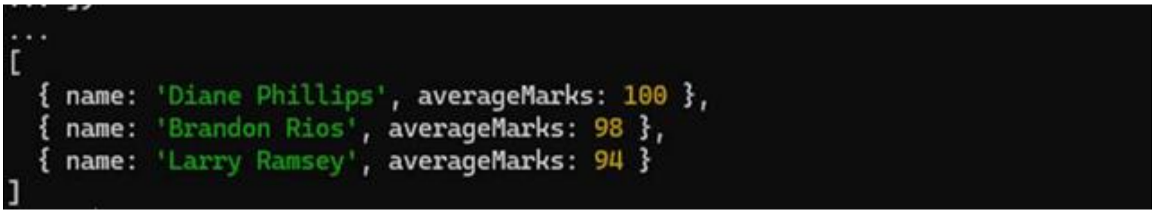
Q5. Find the top 3 students with the highest average marks across all enrolled courses.

QUERY:

```
db.enrollments_full.aggregate([
 {$group: {
    _id: "$student_id",
    averageMarks: { $avg: "$marks" }}
 },{$lookup: {
    from: "students_full",
    localField: "_id",
    foreignField: "_id",
    as: "student_info"
   }},
 {
  $unwind: "$student_info"
 },
 {
  $project: {
    _id: 0,
    name: "$student_info.name",
    averageMarks: { $round: ["$averageMarks", 2] }
   }
 },
 {
  $sort: { averageMarks: -1 }
 },
 {
  $limit: 3
 }
```



```
... db.enrollments_full.aggregate([
...   {
...     $group: {
...       _id: "$student_id",
...       averageMarks: { $avg: "$marks" }
...     }
...   },
...   {
...     $lookup: {
...       from: "students_full",
...       localField: "_id",
...       foreignField: "_id",
...       as: "student_info"
...     }
...   },
...   {
...     $unwind: "$student_info"
...   },
...   {
...     $project: {
...       _id: 0,
...       name: "$student_info.name",
...       averageMarks: { $round: ["$averageMarks", 2] }
...     }
...   },
...   {
...     $sort: { averageMarks: -1 }
...   },
...   {
...     $limit: 3
...   }
... ]);
```

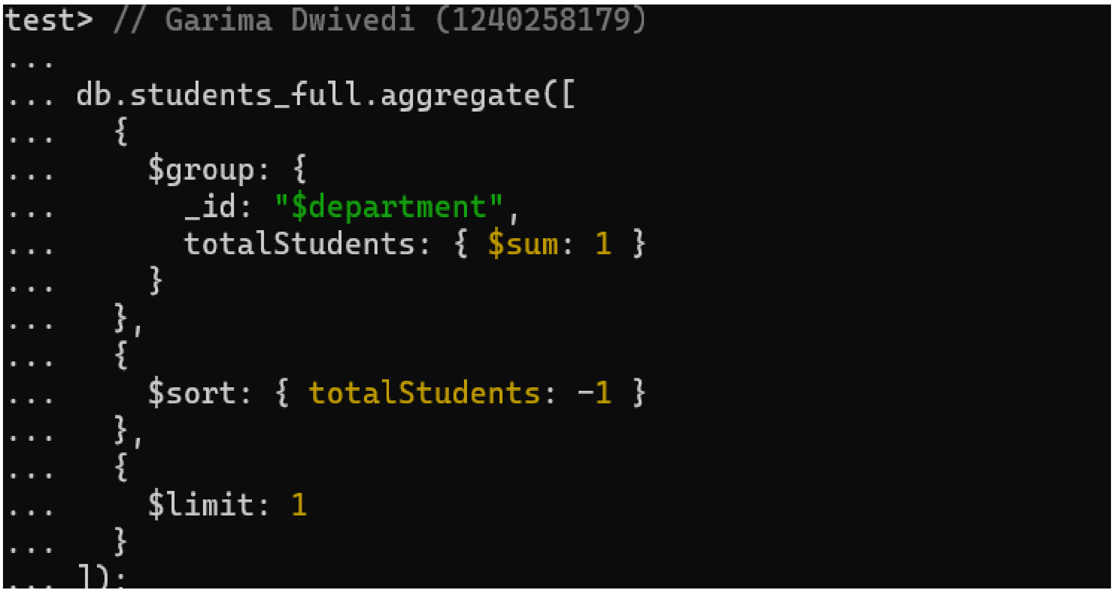


```
...
[
  { name: 'Diane Phillips', averageMarks: 100 },
  { name: 'Brandon Rios', averageMarks: 98 },
  { name: 'Larry Ramsey', averageMarks: 94 }
]
```

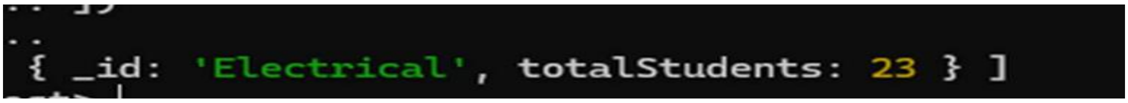
Q6. Count how many students are in each department. Display the department with the highest number of students.

QUERY:

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



```
test> // Garima Dwivedi (1240258179)
...
... db.students_full.aggregate([
...   {
...     $group: {
...       _id: "$department",
...       totalStudents: { $sum: 1 }
...     }
...   },
...   {
...     $sort: { totalStudents: -1 }
...   },
...   {
...     $limit: 1
...   }
... ]);
```



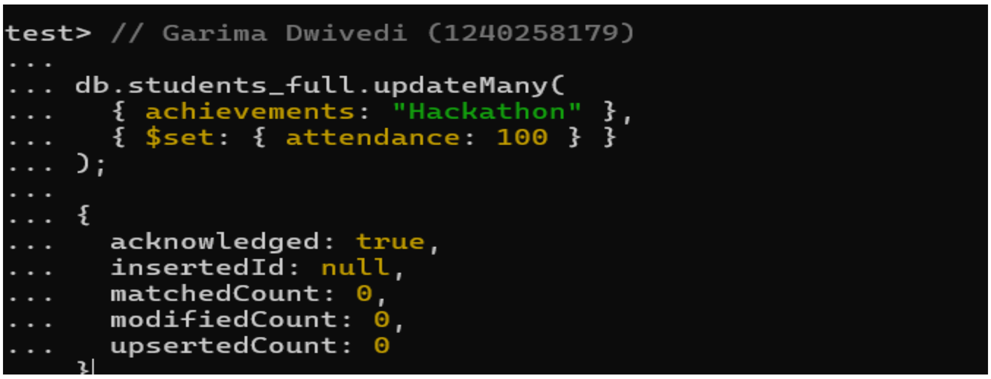
```
..
 { _id: 'Electrical', totalStudents: 23 } ]
```

4. Update, Upsert, and Delete

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

QUERY:

```
db.students_full.updateMany(
  { achievements: "Hackathon" },
  { $set: { attendance: 100 } }
);
```

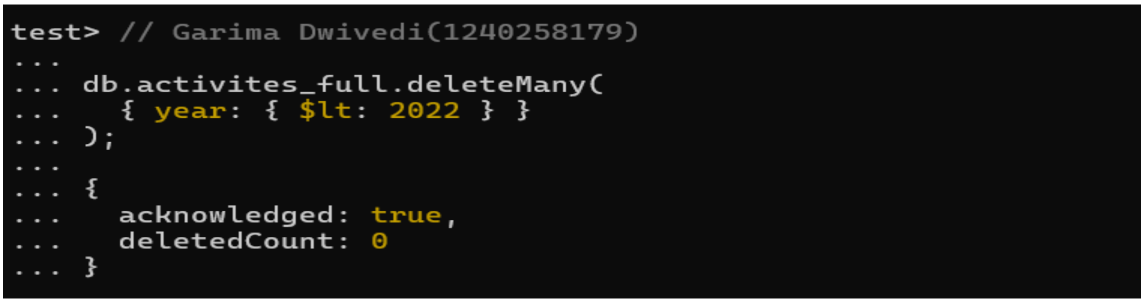


```
test> // Garima Dwivedi (1240258179)
...
... db.students_full.updateMany(
...   { achievements: "Hackathon" },
...   { $set: { attendance: 100 } }
... );
...
... {
...   acknowledged: true,
...   insertedId: null,
...   matchedCount: 0,
...   modifiedCount: 0,
...   upsertedCount: 0
    }
```

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

QUERY:

```
db.activites_full.deleteMany(
  { year: { $lt: 2022 } }
);
```

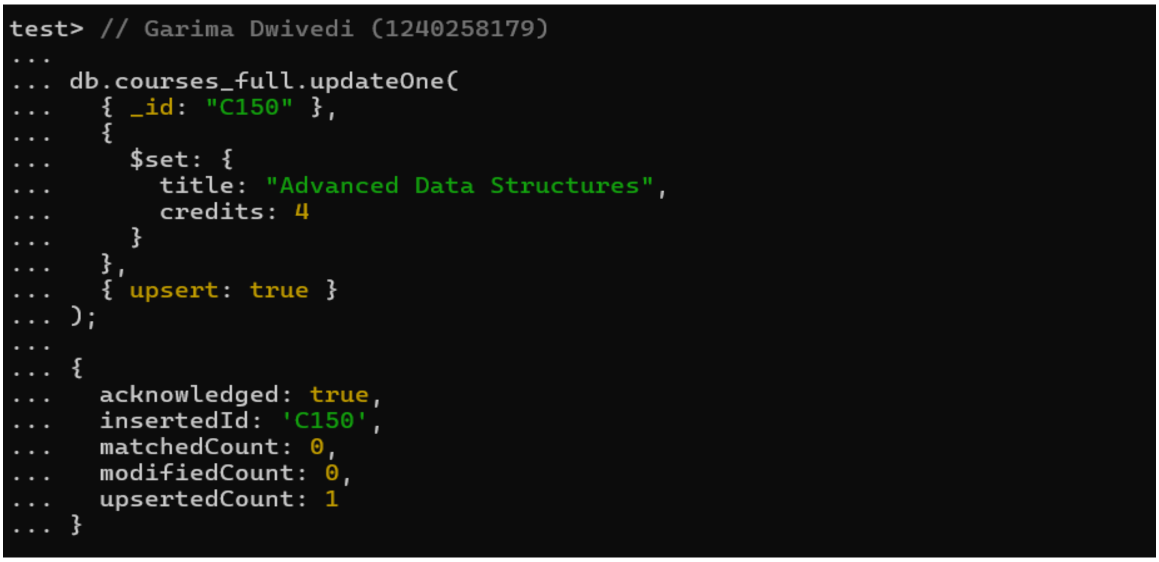


```
test> // Garima Dwivedi(1240258179)
...
... db.activites_full.deleteMany(
...   { year: { $lt: 2022 } }
... );
...
... {
...   acknowledged: true,
...   deletedCount: 0
... }
```

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



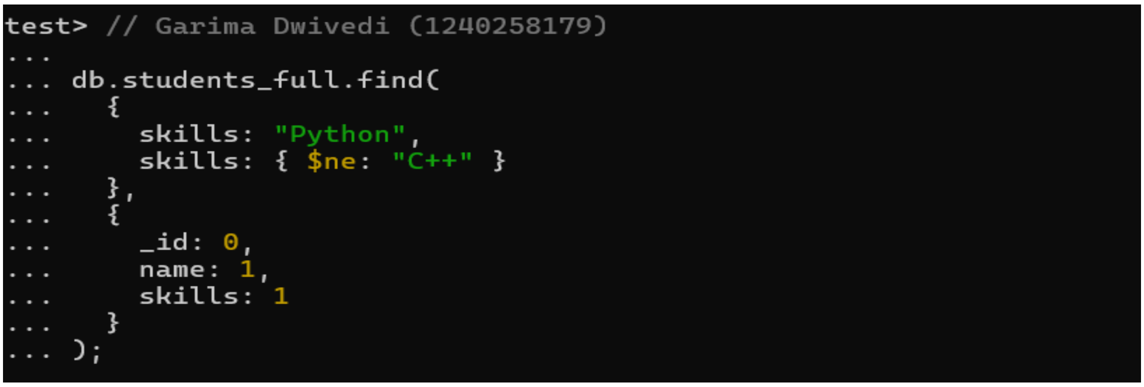
```
test> // Garima Dwivedi (1240258179)
...
... db.courses_full.updateOne(
...   { _id: "C150" },
...   {
...     $set: {
...       title: "Advanced Data Structures",
...       credits: 4
...     }
...   },
...   { upsert: true }
... );
...
... {
...   acknowledged: true,
...   insertedId: 'C150',
...   matchedCount: 0,
...   modifiedCount: 0,
...   upsertedCount: 1
... }
```

5. Array & Operator Usage

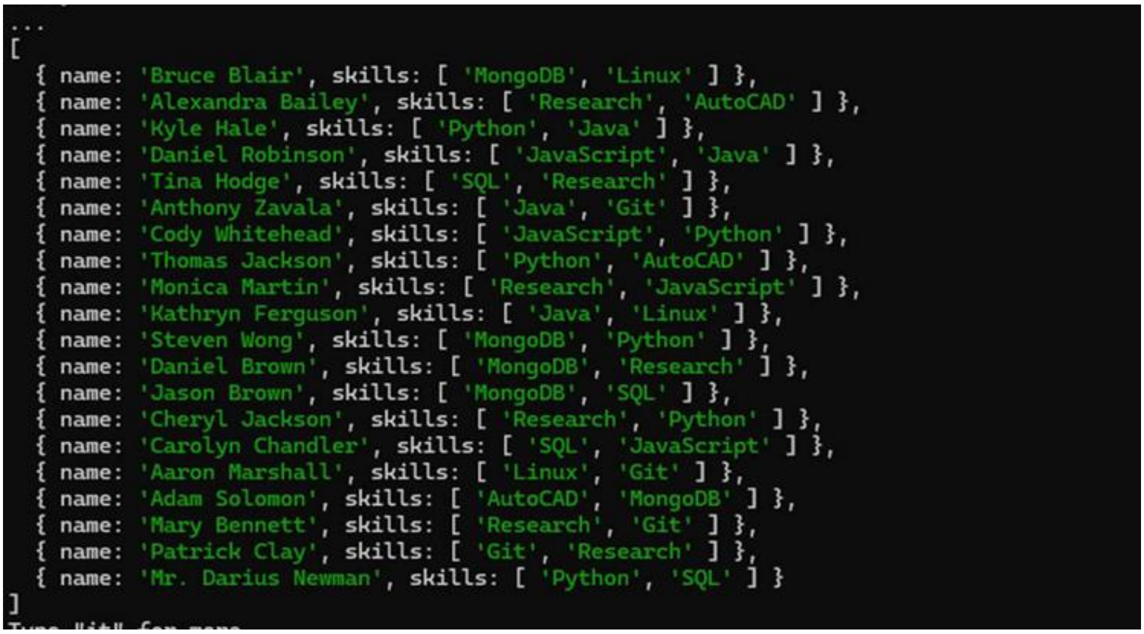
Q10. Find all students who have " Python " as a skill but not " C++".

QUERY:

```
db.students_full.find(
 {
   skills: "Python",
   skills: { $ne: "C++" }
 },{
   _id: 0,
   name: 1,
   skills: 1
}
);
```



```
test> // Garima Dwivedi (1240258179)
...
... db.students_full.find(
...   {
...     skills: "Python",
...     skills: { $ne: "C++" }
...   },
...   {
...     _id: 0,
...     name: 1,
...     skills: 1
...   }
... );
```



```
...
[
  { name: 'Bruce Blair', skills: [ 'MongoDB', 'Linux' ] },
  { name: 'Alexandra Bailey', skills: [ 'Research', 'AutoCAD' ] },
  { name: 'Kyle Hale', skills: [ 'Python', 'Java' ] },
  { name: 'Daniel Robinson', skills: [ 'JavaScript', 'Java' ] },
  { name: 'Tina Hodge', skills: [ 'SQL', 'Research' ] },
  { name: 'Anthony Zavala', skills: [ 'Java', 'Git' ] },
  { name: 'Cody Whitehead', skills: [ 'JavaScript', 'Python' ] },
  { name: 'Thomas Jackson', skills: [ 'Python', 'AutoCAD' ] },
  { name: 'Monica Martin', skills: [ 'Research', 'JavaScript' ] },
  { name: 'Kathryn Ferguson', skills: [ 'Java', 'Linux' ] },
  { name: 'Steven Wong', skills: [ 'MongoDB', 'Python' ] },
  { name: 'Daniel Brown', skills: [ 'MongoDB', 'Research' ] },
  { name: 'Jason Brown', skills: [ 'MongoDB', 'SQL' ] },
  { name: 'Cheryl Jackson', skills: [ 'Research', 'Python' ] },
  { name: 'Carolyn Chandler', skills: [ 'SQL', 'JavaScript' ] },
  { name: 'Aaron Marshall', skills: [ 'Linux', 'Git' ] },
  { name: 'Adam Solomon', skills: [ 'AutoCAD', 'MongoDB' ] },
  { name: 'Mary Bennett', skills: [ 'Research', 'Git' ] },
  { name: 'Patrick Clay', skills: [ 'Git', 'Research' ] },
  { name: 'Mr. Darius Newman', skills: [ 'Python', 'SQL' ] }
]
Type "it" for more
```

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

QUERY:

db.students_full.find(

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

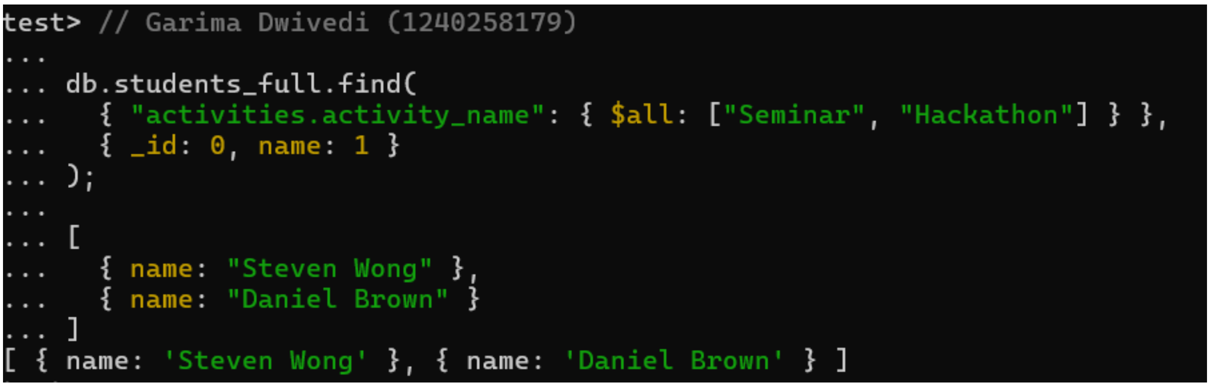
{ _id: 0, name: 1 }

);

[{ name: “Steven Wong” }

{ name: “Daniel Brown”}

]



```
test> // Garima Dwivedi (1240258179)
...
... db.students_full.find(
...   { "activities.activity_name": { $all: ["Seminar", "Hackathon"] } },
...   { _id: 0, name: 1 }
... );
...
... [
...   { name: "Steven Wong" },
...   { name: "Daniel Brown" }
... ]
[ { name: 'Steven Wong' }, { name: 'Daniel Brown' } ]
```

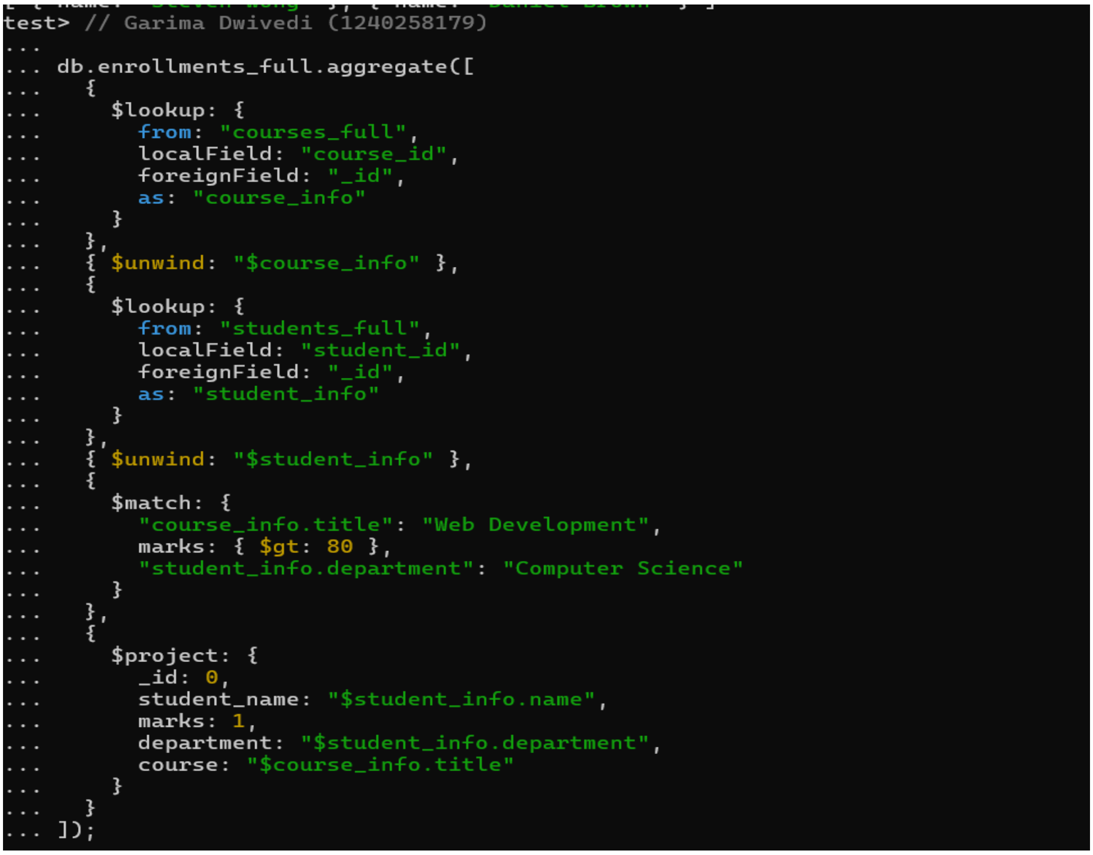
6. Subdocuments and Nested Conditions

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

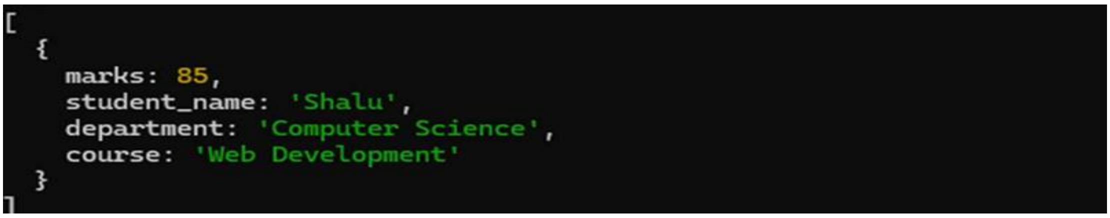
QUERY:

```
db.enrollments_full.aggregate([
  {
    $lookup: {
      from: "courses_full",
      localField: "course_id",
      foreignField: "_id",
      as: "course_info"
    }
  },
  { $unwind: "$course_info" },
  {
    $lookup: {
      from: "students_full",
      localField: "student_id",
      foreignField: "_id",
      as: "student_info"
    }
  },
  { $unwind: "$student_info" },
  {
    $match: {
      "course_info.title": "Web Development",
      marks: { $gt: 80 },
      "student_info.department": "Computer Science"
    }
```

```
},
{ $project: {
  _id: 0,
  student_name: "$student_info.name",
  marks: 1,
  department: "$student_info.department",
```



```
test> // Garima Dwivedi (1240258179)
...
... db.enrollments_full.aggregate([
...   {
...     $lookup: {
...       from: "courses_full",
...       localField: "course_id",
...       foreignField: "_id",
...       as: "course_info"
...     }
...   },
...   { $unwind: "$course_info" },
...   {
...     $lookup: {
...       from: "students_full",
...       localField: "student_id",
...       foreignField: "_id",
...       as: "student_info"
...     }
...   },
...   { $unwind: "$student_info" },
...   {
...     $match: {
...       "course_info.title": "Web Development",
...       marks: { $gt: 80 },
...       "student_info.department": "Computer Science"
...     }
...   },
...   {
...     $project: {
...       _id: 0,
...       student_name: "$student_info.name",
...       marks: 1,
...       department: "$student_info.department",
...       course: "$course_info.title"
...     }
...   }
... ]);
```



```
[
  {
    marks: 85,
    student_name: 'Shalu',
    department: 'Computer Science',
    course: 'Web Development'
  }
]
```

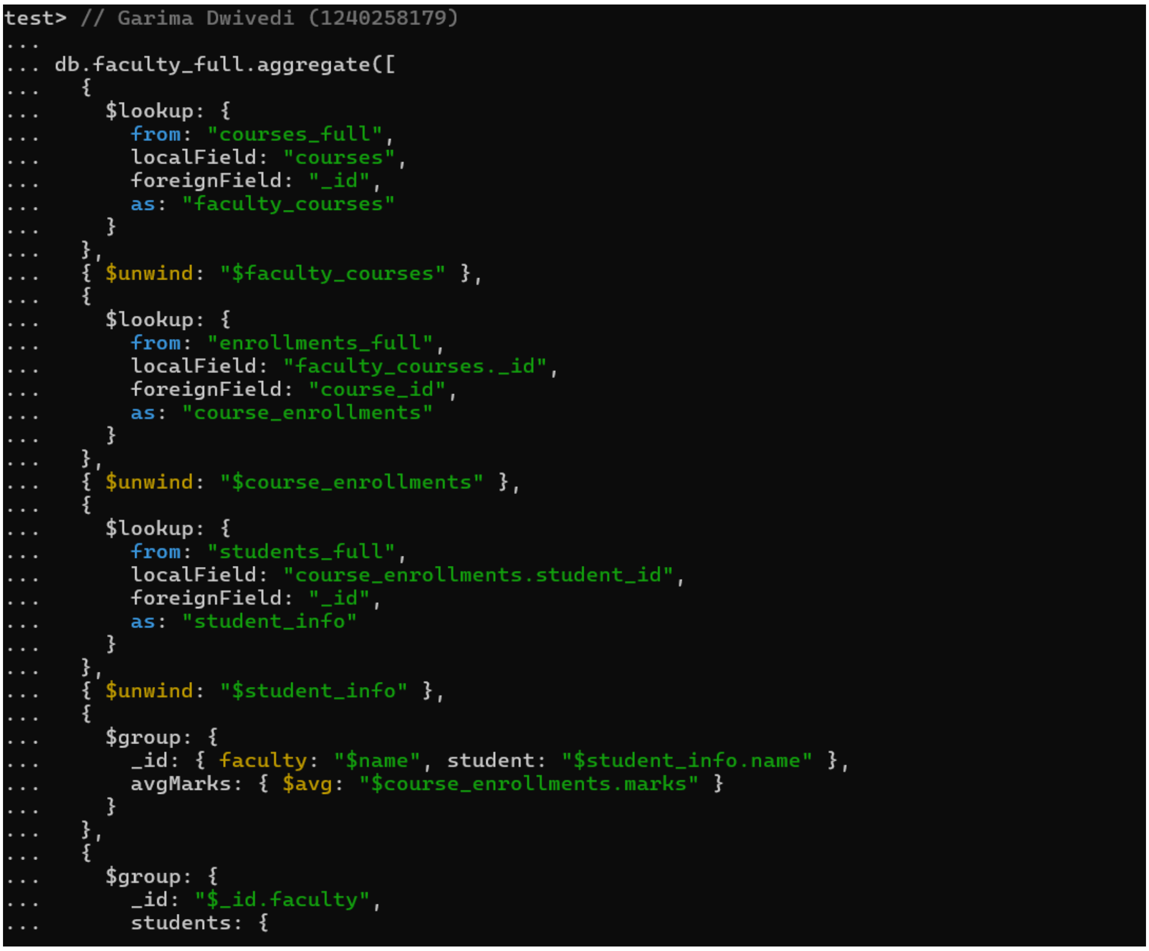
7. Advanced Aggregation (Challenge Level)

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

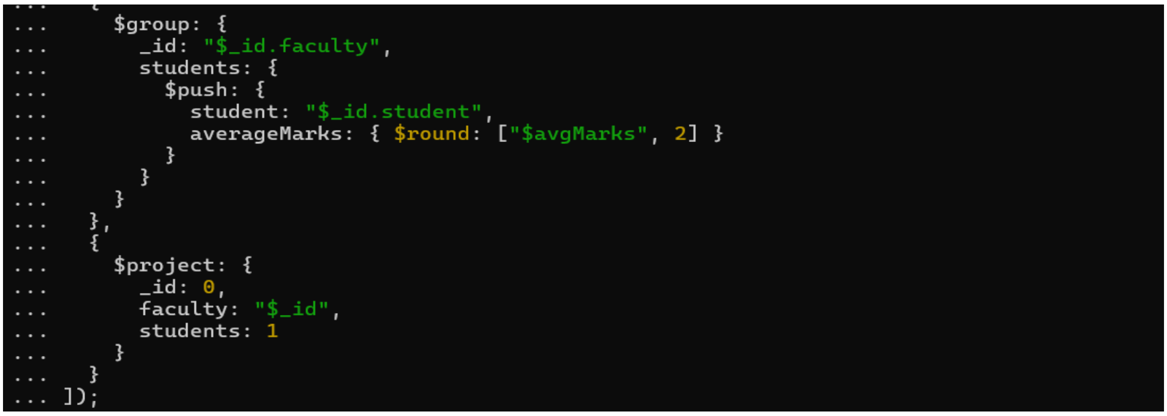
QUERY:

db.faculty_full.aggregate([

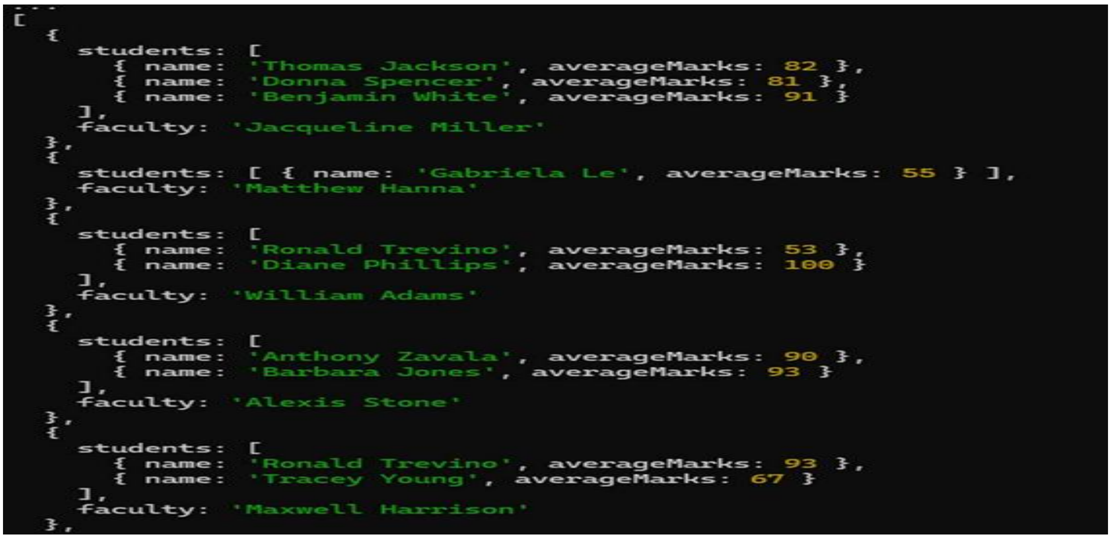
```
{
  $lookup: {
    from: "courses_full",
    localField: "courses",
    foreignField: "_id",
    as: "faculty_courses"
  }
},
{ $unwind: "$faculty_courses" },
{
  $lookup: {
    from: "enrollments_full",
    localField: "faculty_courses._id",
    foreignField: "course_id",
    as: "course_enrollments"
  }
},
{ $unwind: "$course_enrollments" },
{
  $lookup: {
    from: "students_full",
    localField: "course_enrollments.student_id",
    foreignField: "_id",
    as: "student_info"
  }
},
{ $unwind: "$student_info" },
{
  $group: {
      _id: { faculty: "$name", student: "$student_info.name" },
      avgMarks: { $avg: "$course_enrollments.marks" }
    }
  },
  {
    $group: {
      _id: "$_id.faculty",
      students: {
        $push: {
          student: "$_id.student",
          averageMarks: { $round: ["$avgMarks", 2] }
        }
      }
    }
  },
  {
    $project: {
      _id: 0,
      faculty: "$_id",
      students: 1
    }
  }
]);
```



```
test> // Garima Dwivedi (1240258179)
...
... db.faculty_full.aggregate([
...   {
...     $lookup: {
...       from: "courses_full",
...       localField: "courses",
...       foreignField: "_id",
...       as: "faculty_courses"
...     }
...   },
...   { $unwind: "$faculty_courses" },
...   {
...     $lookup: {
...       from: "enrollments_full",
...       localField: "faculty_courses._id",
...       foreignField: "course_id",
...       as: "course_enrollments"
...     }
...   },
...   { $unwind: "$course_enrollments" },
...   {
...     $lookup: {
...       from: "students_full",
...       localField: "course_enrollments.student_id",
...       foreignField: "_id",
...       as: "student_info"
...     }
...   },
...   { $unwind: "$student_info" },
...   {
...     $group: {
...       _id: { faculty: "$name", student: "$student_info.name" },
...       avgMarks: { $avg: "$course_enrollments.marks" }
...     }
...   },
...   {
...     $group: {
...       _id: "$_id.faculty",
...       students: {
```



```
...   {
...     $group: {
...       _id: "$_id.faculty",
...       students: {
...         $push: {
...           student: "$_id.student",
...           averageMarks: { $round: ["$avgMarks", 2] }
...         }
...       }
...     }
...   },
...   {
...     $project: {
...       _id: 0,
...       faculty: "$_id",
...       students: 1
...     }
...   }
... ]);
```

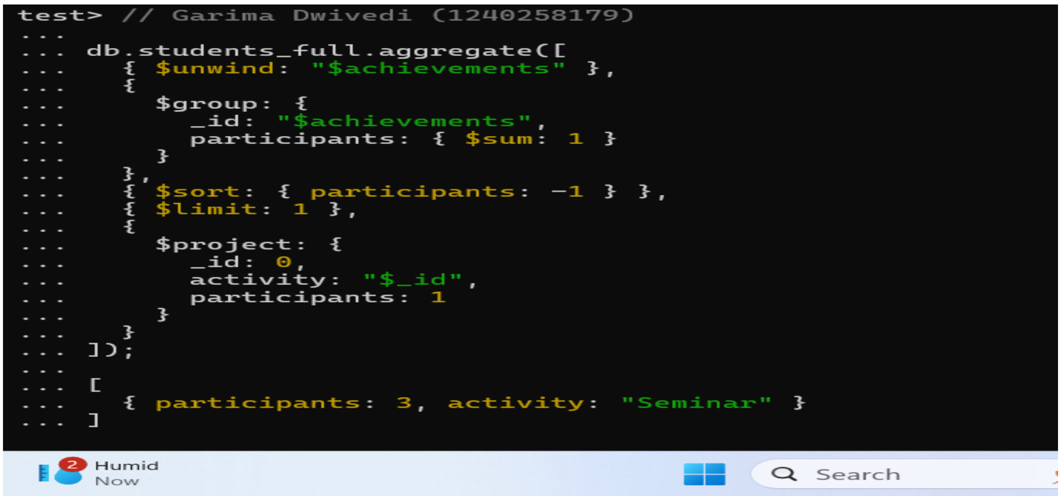


```
...
[
  {
    students: [
      { name: 'Thomas Jackson', averageMarks: 82 },
      { name: 'Donna Spencer', averageMarks: 81 },
      { name: 'Benjamin White', averageMarks: 91 }
    ],
    faculty: 'Jacqueline Miller'
  },
  {
    students: [ { name: 'Gabriela Le', averageMarks: 55 } ],
    faculty: 'Matthew Hanna'
  },
  {
    students: [
      { name: 'Ronald Trevino', averageMarks: 53 },
      { name: 'Diane Phillips', averageMarks: 100 }
    ],
    faculty: 'William Adams'
  },
  {
    students: [
      { name: 'Anthony Zavala', averageMarks: 90 },
      { name: 'Barbara Jones', averageMarks: 93 }
    ],
    faculty: 'Alexis Stone'
  },
  {
    students: [
      { name: 'Ronald Trevino', averageMarks: 93 },
      { name: 'Tracey Young', averageMarks: 67 }
    ],
    faculty: 'Maxwell Harrison'
  },
```

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

QUERY:

```
db.students_full.aggregate([
  { $unwind: "$achievements" },
  {
   $group: {
    _id: "$achievements",
    participants: { $sum: 1 }
   }},
  { $sort: { participants: -1 } },
  { $limit: 1 },{
   $project: {_id: 0;activity: "$_id", participants: 1 } }]
```



```
test> // Garima Dwivedi (1240258179)
...
... db.students_full.aggregate([
...     { $unwind: "$achievements" },
...     {
...       $group: {
...         _id: "$achievements",
...         participants: { $sum: 1 }
...       }
...     },
...     { $sort: { participants: -1 } },
...     { $limit: 1 },
...     {
...       $project: {
...         _id: 0,
...         activity: "$_id",
...         participants: 1
...       }
...     }
... ]);
...
... [
...   { participants: 3, activity: "Seminar" }
... ]
```