

Big Data Management Systems - 2nd Project 2025: MongoDB Aggregation



Maria Schoinaki, BSc Student

Department of Informatics, Athens University of
Economics and Business

p3210191@aueb.gr



Nikos Mitsakis, BSc Student

Department of Informatics, Athens University of
Economics and Business

p3210122@aueb.gr

Below is the write-up template with each question, the exact command used, the observed output, and space for a short explanatory comment when needed. ****We ran the queries in both our machines and had slightly different results (due to the random seed).****

Project Description

This project involves creating a MongoDB collection of student data, performing aggregation queries, and transforming the documents according to specific requirements. The data was loaded from a randomized JavaScript file (`prep.js`) that created a large collection (`students`) with ~10,000 entries.

Setup & Environment

Installed Software and Tools

- MongoDB Community Server 8.0.8
- MongoDB Shell (mongosh) 2.5.0
- MongoDB Compass (for GUI-based visualization)
- Visual Studio Code (with MongoDB extension)

Operating System

- Windows 10

Steps Taken

1. Installed MongoDB Community Server.
2. Installed MongoDB Shell (mongosh) separately.
3. Added necessary paths to the system PATH variable.
4. Started MongoDB service.
5. Verified successful connection using `mongosh` .

6. Loaded `prep.js` into a new database `studentsDB`.

Tasks Completed

1. Load Data

Command Typed:

```
use studentsDB
load("C:\\\\Users\\\\mitsa\\\\Desktop\\\\8th_Semester\\\\Big_Data_Management_Systems\\\\PROJECTS\\\\DBMS-Projects\\\\project_2_MongoDB\\\\data\\\\prep.js")
```

Output:

- DeprecationWarning about `insert()` command, but data loaded successfully.
- Verified with:

```
db.students.findOne()
```

2. Count Students Currently Taking Classes

Command Typed:

```
db.students.countDocuments({ "courses.course_status": "In Progress" })
```

Output: 8833 students

```
8833
```

3. Count Students with Completed Classes

Command Typed:

```
db.students.countDocuments({ "courses.course_status": "Completed" })
```

Output: 9373 students

```
9373
```

4. Group Students by Home City

Command Typed:

```
db.students.aggregate([
  { $group: { _id: "$home_city", number_of_students: { $sum: 1 } } }
])
```

Output: List of cities with student counts

```
[
  { _id: 'Patra', number_of_students: 504 },
  { _id: 'Preveza', number_of_students: 491 },
  { _id: 'Kalamata', number_of_students: 449 },
  { _id: 'Arta', number_of_students: 488 },
  { _id: 'Pyrgos', number_of_students: 509 },
  { _id: 'Irakleio', number_of_students: 480 },
  { _id: 'Messolongi', number_of_students: 481 },
  { _id: 'Athina', number_of_students: 512 },
  { _id: 'Halkida', number_of_students: 500 },
  { _id: 'Kavala', number_of_students: 505 },
]
```

```
{ _id: 'Mytilini', number_of_students: 509 },
{ _id: 'Florina', number_of_students: 505 },
{ _id: 'Ioannina', number_of_students: 522 },
{ _id: 'Rethymno', number_of_students: 475 },
{ _id: 'Agrinio', number_of_students: 528 },
{ _id: 'Katerini', number_of_students: 492 },
{ _id: 'Thyra', number_of_students: 546 },
{ _id: 'Larissa', number_of_students: 496 },
{ _id: 'Thessaloniki', number_of_students: 497 },
{ _id: 'Chania', number_of_students: 511 }
]
```

5. Find Most Popular Hobby

Command Typed:

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

Output: 'guitar' with 1305 occurrences

```
[ { _id: 'guitar', count: 1305 } ]
```

6. Find GPA of the Best Student

Command Typed:

```

db.students.aggregate([
  {
    $addFields: {
      gpa: {
        $avg: {
          $map: {
            input: {
              $filter: {
                input: "$courses",
                as: "course",
                cond: { $eq: ["$$course.course_status", "Completed"] }
            }
          }
        }
      },
      completedCourse: {
        $in: "$$completedCourse.grade"
      }
    }
  },
  { $sort: { gpa: -1 } },
  { $limit: 1 },
  { $project: { first_name: 1, gpa: 1 } }
])

```

Output: Eleni with GPA = 10

```

[
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f89c'),

```

```
    first_name: 'Eleni',  
    gpa: 10  
  }  
]
```

7. Find Student with Most Grade 10's

Command Typed:

```
db.students.aggregate([  
  {  
    $addFields: {  
      tens: {  
        $size: {  
          $filter: {  
            input: "$courses",  
            as: "course",  
            cond: { $eq: ["$$course.grade", 10] }  
          }  
        }  
      }  
    }  
  },  
  { $sort: { tens: -1 } },  
  { $limit: 1 },  
  { $project: { first_name: 1, tens: 1 } }  
])
```

Output: Clio with 6 grade 10's

```
[  
  {
```

```
_id: ObjectId('681079d2a9ee5b8ed4b60ae1'),
first_name: 'Clio',
tens: 6
}
]
```

8. Find Class with Highest Average GPA

Command Typed:

```
db.students.aggregate([
  { $unwind: "$courses" },
  { $match: { "courses.course_status": "Complete" } },
  { $group: { _id: "$courses.course_title", avg_grade: { $avg:
"$courses.grade" } } },
  { $sort: { avg_grade: -1 } },
  { $limit: 1 }
])
```

Output: 'Algorithms and Data Structures' with avg GPA ~7.814

```
[
  {
    _id: 'Algorithms and Data Structures',
    avg_grade: 7.813975448536355
  }
]
```

9. Find Class Dropped Most Times

Command Typed:


```
db.students.aggregate([
  { $unwind: "$courses" },
  { $match: { "courses.course_status": "Dropped" } },
  { $group: { _id: "$courses.course_title", drops: { $sum: 1 } } },
  { $sort: { drops: -1 } },
  { $limit: 1 }
])
```

Output: 'Algorithms and Data Structures' dropped 244 times

```
[ { _id: 'Algorithms and Data Structures', drops: 244 } ]
```

10. Completed Classes by Class Type

Command Typed:

```
db.students.aggregate([
  { $unwind: "$courses" },
  { $match: { "courses.course_status": "Complete" } },
  { $group: { _id: { $substrBytes: ["$courses.course_code", 0,
1] }, count: { $sum: 1 } } },
  { $sort: { count: -1 } }
])
```

Output: P: 6756, M: 4483, S: 4376, D: 3287, V: 2274, B: 1111

```
[
  { _id: 'P', count: 6756 },
  { _id: 'M', count: 4483 },
  { _id: 'S', count: 4376 },
  { _id: 'D', count: 3287 },
```

```
{ _id: 'V', count: 2274 },  
{ _id: 'B', count: 1111 }  
]
```

11. Add Field 'Hobbyist'

Command Typed:

```
db.students.aggregate([  
  {  
    $addFields: {  
      hobbyist: {  
        $gt: [{ $size: "$hobbies" }, 3]  
      }  
    }  
  },  
  { $project: { first_name: 1, hobbies: 1, hobbyist: 1 } }  
])
```

Output: Students flagged with `hobbyist: true` or `false`

```
[  
  {  
    _id: ObjectId('681079c8a9ee5b8ed4b5f899'),  
    first_name: 'Nikos',  
    hobbies: [ 'cinema', 'skydiving', 'guitar', 'poetry' ],  
    hobbyist: true  
  },  
  {  
    _id: ObjectId('681079c8a9ee5b8ed4b5f89a'),  
    first_name: 'Despoina',  
    hobbies: [ 'archaeology', 'board games' ],  
    hobbyist: false  
  }  
]
```

```
    hobbyist: false
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f89b'),
    first_name: 'Pavlos',
    hobbies: [ 'swimming', 'cinema', 'paintball' ],
    hobbyist: false
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f89c'),
    first_name: 'Eleni',
    hobbies: [ 'philately', 'swimming' ],
    hobbyist: false
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f89d'),
    first_name: 'Georgia',
    hobbies: [ 'model cars' ],
    hobbyist: false
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f89e'),
    first_name: 'Nikos',
    hobbies: [ 'skiing', 'board games' ],
    hobbyist: false
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f89f'),
    first_name: 'Clio',
    hobbies: [ 'coin collecting' ],
    hobbyist: false
  },
  },
```

```

{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a0'),
  first_name: 'Sokratis',
  hobbies: [ 'archaeology', 'board games' ],
  hobbyist: false
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a1'),
  first_name: 'Danae',
  hobbies: [ 'piano', 'model cars', 'board games', 'gardening'
],
  hobbyist: true
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a2'),
  first_name: 'Myrto',
  hobbies: [ 'World of Warcraft', 'coin collecting', 'swimming',
'cinema' ],
  hobbyist: true
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a3'),
  first_name: 'Alexandra',
  hobbies: [ 'gardening', 'watercolour painting', 'coin collectin
g' ],
  hobbyist: false
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a4'),
  first_name: 'Vangelis',
  hobbies: [ 'model cars', 'coin collecting' ],
  hobbyist: false
}

```

```

},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a5'),
  first_name: 'Alexandra',
  hobbies: [ 'ventriloquism', 'gardening', 'skydiving', 'hiking' ],
  hobbyist: true
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a6'),
  first_name: 'Myrto',
  hobbies: [ 'archaeology', 'piano' ],
  hobbyist: false
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a7'),
  first_name: 'Thanos',
  hobbies: [ 'skiing' ],
  hobbyist: false
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a8'),
  first_name: 'Iris',
  hobbies: [ 'poetry', 'swimming' ],
  hobbyist: false
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a9'),
  first_name: 'Nikos',
  hobbies: [ 'paintball', 'swimming' ],
  hobbyist: false
},
{

```

```

    _id: ObjectId('681079c8a9ee5b8ed4b5f8aa'),
    first_name: 'Kostas',
    hobbies: [ 'hiking', 'poetry' ],
    hobbyist: false
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f8ab'),
    first_name: 'Miltos',
    hobbies: [ 'AD&D' ],
    hobbyist: false
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f8ac'),
    first_name: 'Despoina',
    hobbies: [ 'model cars', 'coin collecting' ],
    hobbyist: false
  }
]

```

12. Add Field 'Completed Classes'

Command Typed:

```

db.students.aggregate([
  {
    $addFields: {
      completed_classes: {
        $size: {
          $filter: {
            input: "$courses",
            as: "course",
            cond: { $eq: ["$$course.course_status", "Complete"] }
          }
        }
      }
    }
  }
])

```

```

    }
  }
}
},
{ $project: { first_name: 1, completed_classes: 1 } }
])

```

Output: Field `completed_classes` showing number of completed courses

```

[
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f899'),
    first_name: 'Nikos',
    completed_classes: 3
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f89a'),
    first_name: 'Despoina',
    completed_classes: 1
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f89b'),
    first_name: 'Pavlos',
    completed_classes: 0
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f89c'),
    first_name: 'Eleni',
    completed_classes: 3
  },
]

```

```
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f89d'),
  first_name: 'Georgia',
  completed_classes: 4
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f89e'),
  first_name: 'Nikos',
  completed_classes: 2
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f89f'),
  first_name: 'Clio',
  completed_classes: 2
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a0'),
  first_name: 'Sokratis',
  completed_classes: 3
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a1'),
  first_name: 'Danae',
  completed_classes: 0
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a2'),
  first_name: 'Myrto',
  completed_classes: 3
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a3'),
```



```
    first_name: 'Alexandra',
    completed_classes: 1
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f8a4'),
    first_name: 'Vangelis',
    completed_classes: 3
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f8a5'),
    first_name: 'Alexandra',
    completed_classes: 1
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f8a6'),
    first_name: 'Myrto',
    completed_classes: 2
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f8a7'),
    first_name: 'Thanos',
    completed_classes: 4
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f8a8'),
    first_name: 'Iris',
    completed_classes: 2
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f8a9'),
    first_name: 'Nikos',
    completed_classes: 1
  }
]
```

```

},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8aa'),
  first_name: 'Kostas',
  completed_classes: 0
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8ab'),
  first_name: 'Miltos',
  completed_classes: 1
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8ac'),
  first_name: 'Despoina',
  completed_classes: 3
}
]

```

13. Final Document Transformation

Command Typed:

```

db.students.aggregate([
  {
    $addFields: {
      GPA: {
        $avg: {
          $map: {
            input: {
              $filter: {
                input: "$courses",
                as: "course",

```

```

        cond: { $eq: ["$$course.course_status", "Completed"] }
      }
    },
    as: "completedCourse",
    in: "$$completedCourse.grade"
  }
},
classesInProgress: {
  $size: {
    $filter: {
      input: "$courses",
      as: "course",
      cond: { $eq: ["$$course.course_status", "In Progress"] }
    }
  }
},
droppedClasses: {
  $size: {
    $filter: {
      input: "$courses",
      as: "course",
      cond: { $eq: ["$$course.course_status", "Dropped"] }
    }
  }
},
{ $project: { first_name: 1, GPA: 1, classesInProgress: 1, drop

```

```
pedClasses: 1 } }  
])
```

Output:

```
[  
  {  
    _id: ObjectId('681079c8a9ee5b8ed4b5f899'),  
    first_name: 'Nikos',  
    GPA: 8,  
    classesInProgress: 1,  
    droppedClasses: 0  
  },  
  {  
    _id: ObjectId('681079c8a9ee5b8ed4b5f89a'),  
    first_name: 'Despoina',  
    GPA: 4,  
    classesInProgress: 1,  
    droppedClasses: 1  
  },  
  {  
    _id: ObjectId('681079c8a9ee5b8ed4b5f89b'),  
    first_name: 'Pavlos',  
    GPA: null,  
    classesInProgress: 3,  
    droppedClasses: 0  
  },  
  {  
    _id: ObjectId('681079c8a9ee5b8ed4b5f89c'),  
    first_name: 'Eleni',  
    GPA: 10,  
    classesInProgress: 3,  
  }  
]
```

```
    droppedClasses: 1
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f89d'),
    first_name: 'Georgia',
    GPA: 8.5,
    classesInProgress: 1,
    droppedClasses: 0
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f89e'),
    first_name: 'Nikos',
    GPA: 9.5,
    classesInProgress: 0,
    droppedClasses: 2
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f89f'),
    first_name: 'Clio',
    GPA: 8,
    classesInProgress: 5,
    droppedClasses: 0
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f8a0'),
    first_name: 'Sokratis',
    GPA: 4.666666666666667,
    classesInProgress: 1,
    droppedClasses: 0
  },
  {
    _id: ObjectId('681079c8a9ee5b8ed4b5f8a1'),
```

```
first_name: 'Danae',
GPA: null,
classesInProgress: 3,
droppedClasses: 1
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a2'),
  first_name: 'Myrto',
  GPA: 6,
  classesInProgress: 1,
  droppedClasses: 0
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a3'),
  first_name: 'Alexandra',
  GPA: 8,
  classesInProgress: 3,
  droppedClasses: 1
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a4'),
  first_name: 'Vangelis',
  GPA: 5.666666666666667,
  classesInProgress: 2,
  droppedClasses: 0
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a5'),
  first_name: 'Alexandra',
  GPA: 8,
  classesInProgress: 3,
  droppedClasses: 0
}
```

```

},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a6'),
  first_name: 'Myrto',
  GPA: 5,
  classesInProgress: 2,
  droppedClasses: 0
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a7'),
  first_name: 'Thanos',
  GPA: 10,
  classesInProgress: 0,
  droppedClasses: 0
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a8'),
  first_name: 'Iris',
  GPA: 6.5,
  classesInProgress: 1,
  droppedClasses: 0
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8a9'),
  first_name: 'Nikos',
  GPA: 10,
  classesInProgress: 3,
  droppedClasses: 1
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8aa'),
  first_name: 'Kostas',

```

```
GPA: null,
classesInProgress: 4,
droppedClasses: 1
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8ab'),
  first_name: 'Miltos',
  GPA: 6,
  classesInProgress: 1,
  droppedClasses: 1
},
{
  _id: ObjectId('681079c8a9ee5b8ed4b5f8ac'),
  first_name: 'Despoina',
  GPA: 5.333333333333333,
  classesInProgress: 1,
  droppedClasses: 0
}
]
```

Conclusion

All project tasks outlined in [Proj2_MongoDB.pdf](#) have been successfully completed. The dataset was loaded, queried, grouped, transformed, and cleaned according to specifications. This project involved practical experience with the aggregation pipeline, filtering, grouping, mapping, and transforming MongoDB documents efficiently.

Notes

- Some students have `GPA: null` because they have no completed classes.
 - Queries were run inside MongoDB Shell (mongosh 2.5.0).
 - Database used: `studentsDB`, collection used: `students`.
 - Seed Variability: prep.js uses a random seed to generate the students collection, so numeric results (e.g., counts, GPA, most popular hobby) will differ slightly each time.
 - The GPA stage uses a `$cond` to avoid division by zero, returning `null` when no completed courses exist.
-

Appendix

Appendix A: Nikos's raw log excerpt

```
PS C:/Users/mitsa/Desktop/8th_Semester/Big_Data_Management_Systems/PROJECTS/DBMS-Projects/project_2_MongoDB
> mongosh
Current Mongosh Log ID: 68107919a9ee5b8ed4b5f898
Connecting to:      mongodb://127.0.0.1:27017/?...
test> use studentsDB
```

Appendix B: Maria's raw log excerpt

```
PS C:/Program Files/MongoDB/mongosh-2.5.1-win32-x64/bin
> ./mongosh.exe
Current Mongosh Log ID: 681b4d0ac16edfdcde6c4bcf
Connecting to:      mongodb://127.0.0.1:27017/?...
test> use studentsDB
```

