

**DASC 5333 Database Systems for Data Science**  
**CSCI 4333 Design of Database Systems**  
**Spring 2023**  
**Homework #8**

## Simple MongoDB Assignment

Create the 'toyu' database in MongoDB.

[1] Download the file: [toyu-db.gz](https://www.mongodb.com/try/download/database-tools).

[2] Ensure that you have download MongoDB tools: command line utilities including import and export, <https://www.mongodb.com/try/download/database-tools>.

[3] Run the command: `mongorestore --archive="toyu-db.gz" --gzip --nsFrom='toyu.*' --nsTo='toyu.*'`

Note that the design of toyu is not the typical way one would design a MongoDB. Instead, it is intended to look like the toyu MySQL database for ease of comparison.

Construct JS code *that works inside Mongosh* for the following data problem. Use mongosh to test your solution. Do you develop standalone Node JS program Put your solution in a JS file (such as h8sol.js) and turn it in through Blackboard. It may be necessary for you to add a .txt extension (such as h8sol.js.txt). The TA will execute your .js submission.

[1] S Show the stuId, classId and grade of all enrollment with a grade of "A", "A-", "B+", "B", or "B-" in the following manner.

```
[
  { stuId: 100000, classId: 10000, grade: 'A' },
  { stuId: 100002, classId: 10000, grade: 'B-' },
  { stuId: 100000, classId: 10001, grade: 'A' },
  { stuId: 100001, classId: 10001, grade: 'A-' },
  { stuId: 100000, classId: 10002, grade: 'B+' },
  { stuId: 100002, classId: 10002, grade: 'B+' },
  { stuId: 100004, classId: 10003, grade: 'A' },
  { stuId: 100000, classId: 10004, grade: 'A-' },
  { stuId: 100004, classId: 10004, grade: 'B+' },
  { stuId: 100005, classId: 10004, grade: 'A-' },
  { stuId: 100005, classId: 10005, grade: 'A-' },
  { stuId: 100006, classId: 10005, grade: 'A' },
  { stuId: 100005, classId: 10006, grade: 'B+' },
  { stuId: 100007, classId: 10008, grade: 'A-' }
]
```

[2] Show the classId and the number of students enrolled in the class in the following manner.

```
[
  { classId: 10000, 'number of students': 3 },
  { classId: 10006, 'number of students': 1 },
  { classId: 10003, 'number of students': 4 },
  { classId: 10002, 'number of students': 2 },
]
```

```
{ classId: 11001, 'number of students': 1 },
{ classId: 10001, 'number of students': 2 },
{ classId: 10007, 'number of students': 2 },
{ classId: 10004, 'number of students': 4 },
{ classId: 10008, 'number of students': 1 },
{ classId: 10005, 'number of students': 2 }
]
```

[3] Show the name, and deptCode of all faculty members who have the substring "an" in their first names or last names in JSON form in the following manner.

```
[
{ deptCode: 'CSCI', faculty: 'Mary Tran' },
{ deptCode: 'CSCI', faculty: 'Sharon Mannes' },
{ deptCode: 'CINF', faculty: 'Daniel Kim' },
{ deptCode: 'CINF', faculty: 'Andrew Byre' }
]
```

[4] Show the name, and deptCode of all faculty members who have the substring "an" in their first names or last names in the following manner.

Faculty with the substring "an" in their names.

and then

```
[1] Mary Tran: CSCI
[2] Sharon Mannes: CSCI
[3] Daniel Kim: CINF
[4] Andrew Byre: CINF
```

[5] (Bonus: 20%) Show the name, and deptCode of all faculty members who have the substring "an" in their first names or last names in the following manner. The full department names are included.

Faculty with the substring "an" in their names.

and then

```
[1] Mary Tran: Computer Science (CSCI)
[2] Sharon Mannes: Computer Science (CSCI)
[3] Daniel Kim: Computer Information Systems (CINF)
[4] Andrew Byre: Computer Information Systems (CINF)
```