Fall 2020 – Quiz 6 (MongoDB)

10 points, 10 minutes

Monday afternoon section

Consider a table in a relational database: student(id, name, age, gender). Here are example tuples of the table: (1, “John Smith”, 25, “M”) and (2, “David Smith”, 28, “M”), and (3, “Mary”, 30, null). Consider storing the content of the table in MongoDB as a collection called “student”. For each of the following questions, write a MongoDB operation to answer the question.

- Adding the example tuples shown above to the “student” collection.

  ```
  db.student.insertMany([
  {"_id":1, "name":"John Smith", "age":25, "gender":"M"},
  {"_id":2, "name":"David Smith", "age":28, "gender":"M"},
  {"_id":3, "name":"Mary", "age":30, "gender":null}])
  ```
  Note: if using 3 insert query, -0.5

- Updating the gender of Mary to “F”.
  ```
  db.student.update( {"name": "Mary"}, {$set: {"gender": "F"}})
  ```
- Find only the names of male students who are at least 25 years old and whose name contain “smith” (case insensitive).
  ```
  db.student.find( { $and: [{"age": {$gte: 25}}, {"name": /smith/i}, {"gender": "M"}] }, {"_id": 0,"name": 1})
  ```
  OR
  ```
  db.student.find( {"age": {$gte: 25}, "name": /smith/i, "gender": "M"}, {"_id": 0,"name": 1})
  ```
  Note: missing projection -1, missing condition -0.5/1 condition, missing $ -0.5/each, wrong {/} -0.5
- Find out for each gender, how many students who are at least 25 years old.

  ```
  db.student.aggregate(
    {$match:
      { "age": {$gte: 25}},
    },
    {$group:
      { _id: "$gender", total: { $sum: 1}}
    })
  ```
  Note: missing $ -0.5/each, wrong group by -1, wrong order for match and group -1