Mongo DB Assignment 2

1. Create a database named university and a collection named students. Insert multiple student documents with fields: name, age, department, and grades.



db.student.insertMany([{name:"Alice",age:20,department:"Computer Science",grades:{"math":85,"english":92}},

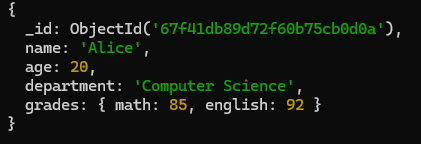
{name:"Bob",age:21,department:"Physics",grades:{"math":80,"english":90}},

{name:"Charlie",age:22,department:"Mathematics",grades:{"math":95,"english":89}}

])

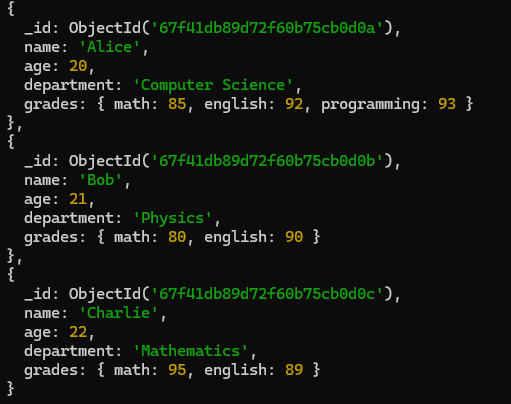
1. Write a query to display all students who are in the Computer Science department.

db.student.find({department:"Computer Science"})



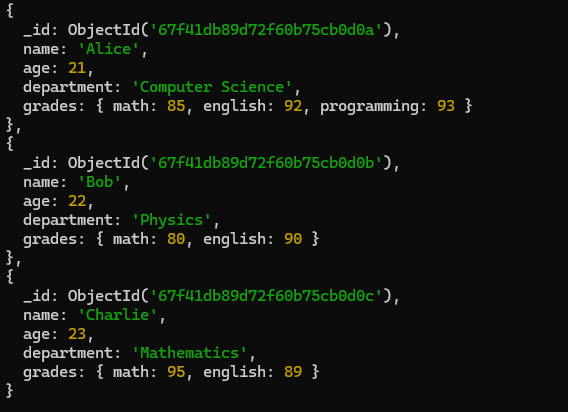
1. Write a query to update the grades of a student named Alice by adding a new subject programming with a grade of 93.

db.student.updateOne( { name: "Alice" }, { $set: { "grades.programming": 93 } })



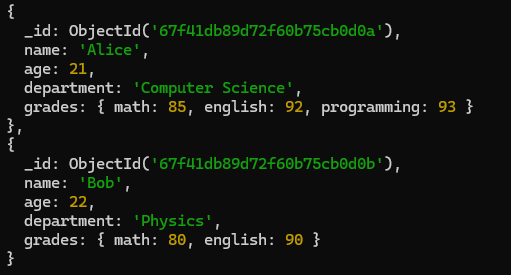
4. Write a query to increment the age of all students by 1.

db.student.updateMany({},{$inc:{age:1}})



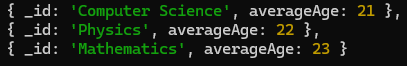
5. Write a query to delete all students who are 23 years old.

db.student.deleteMany({age:{$eq:23}})



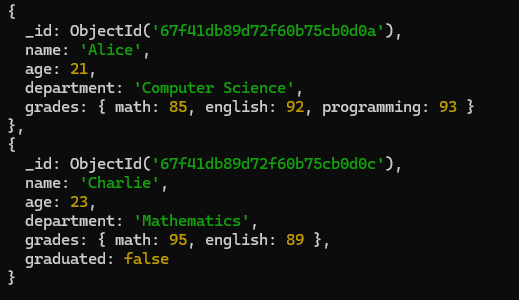
6. Write an aggregation query to group students by their department and calculate the average age in each department.

db.student.aggregate([ { $group: { \_id: "$department", averageAge: { $avg: "$age" } } }] )



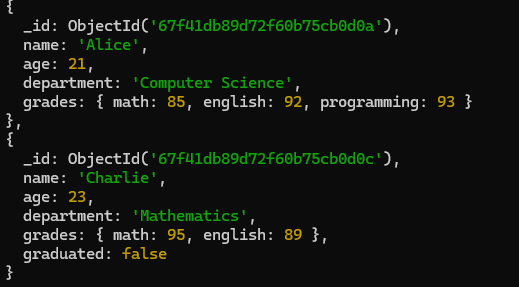
7. Write a query to find all students who have scored more than 90 in any subject.

db.student.find({$or:[{"grades.math":{$gt:90}},{"grades.english":{$gt:90}},{"grades.programming":{$gt:90}}]})



8. Write a query to add a new field graduated set to false for all students who are in the Mathematics department.

db.student.updateMany({department:"Mathematics"},{$set:{graduated:false}})



9. How can you retrieve only the name and department fields for all students, excluding the \_id field?

db.student.find({},{name:1,department:1,\_id:0})

