**NOSQL ASSIGMENT-2**

**By Gaurav mathur**

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

**CODE ->**

Use university

db.createCollection(“Students”)

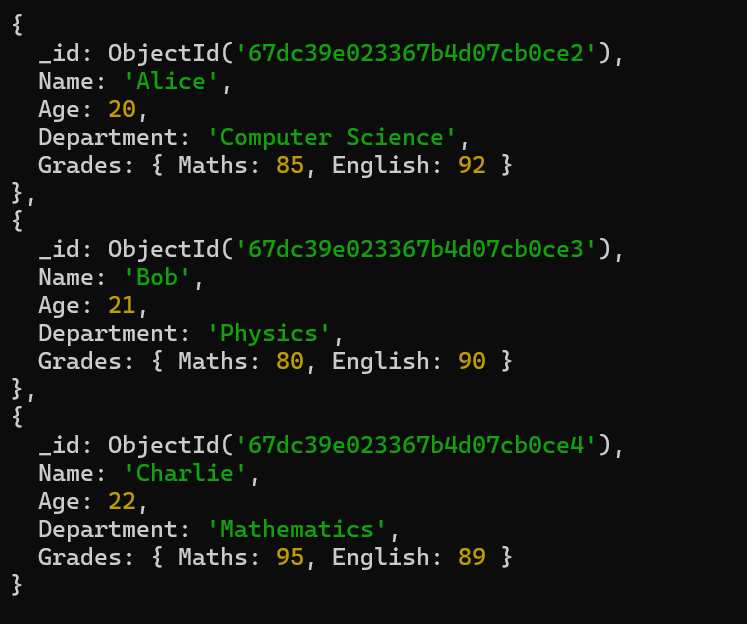
db.students.insertMany

([{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}}

])



**#2 Write a query to display all students who are in the Computer Science department**.

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

A computer code on a black background

AI-generated content may be incorrect.

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

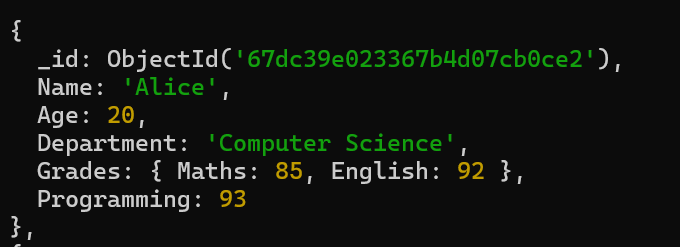
db.Students.updateOne(

{ name: "Alice" },

{ $set: { "grades.programming": 93 } }

A screen shot of a computer

AI-generated content may be incorrect.



**#4 Write a query to increment the age of all students by 1?**

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

A screen shot of a computer

AI-generated content may be incorrect.A computer screen shot of a computer program

AI-generated content may be incorrect.

**#5 Write a query to delete all students who are 23 years old?**

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

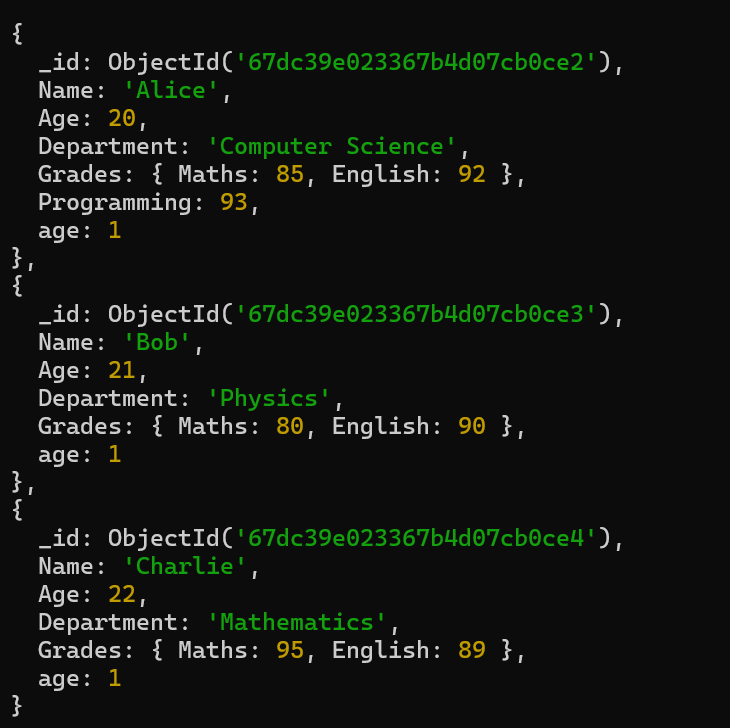


A computer screen shot of a computer code

AI-generated content may be incorrect.

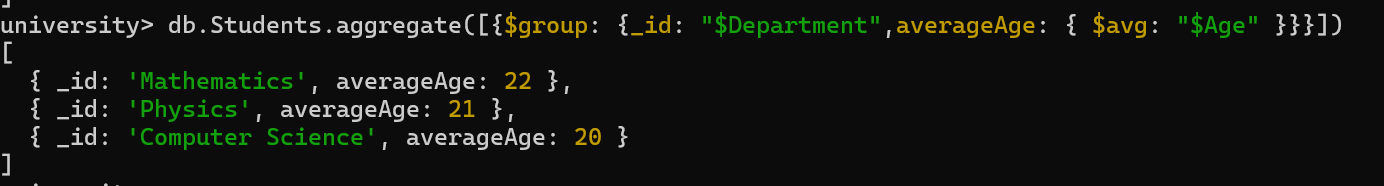
**#6 Write a query to create an index on the name field of the students collection?**

db.Students.find().sort()



**#7 Write an aggregation query to group students by their department and calculate the average age in each department?**

db.Students.aggregate([{$group: {\_id: "$Department",averageAge: { $avg: "$Age" }}}])



**#8 Write a query to find all students who have scored more than 90 in any subject?**

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

A computer screen shot of a computer code

AI-generated content may be incorrect.

**#9 Write a query to add a new field graduated set to false for all students who are in the Mathematics department?**

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

A screen shot of a computer code

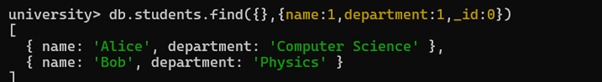
AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**#10 How can you retrieve only the name and department fields for all students, excluding the \_id field?**

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

****