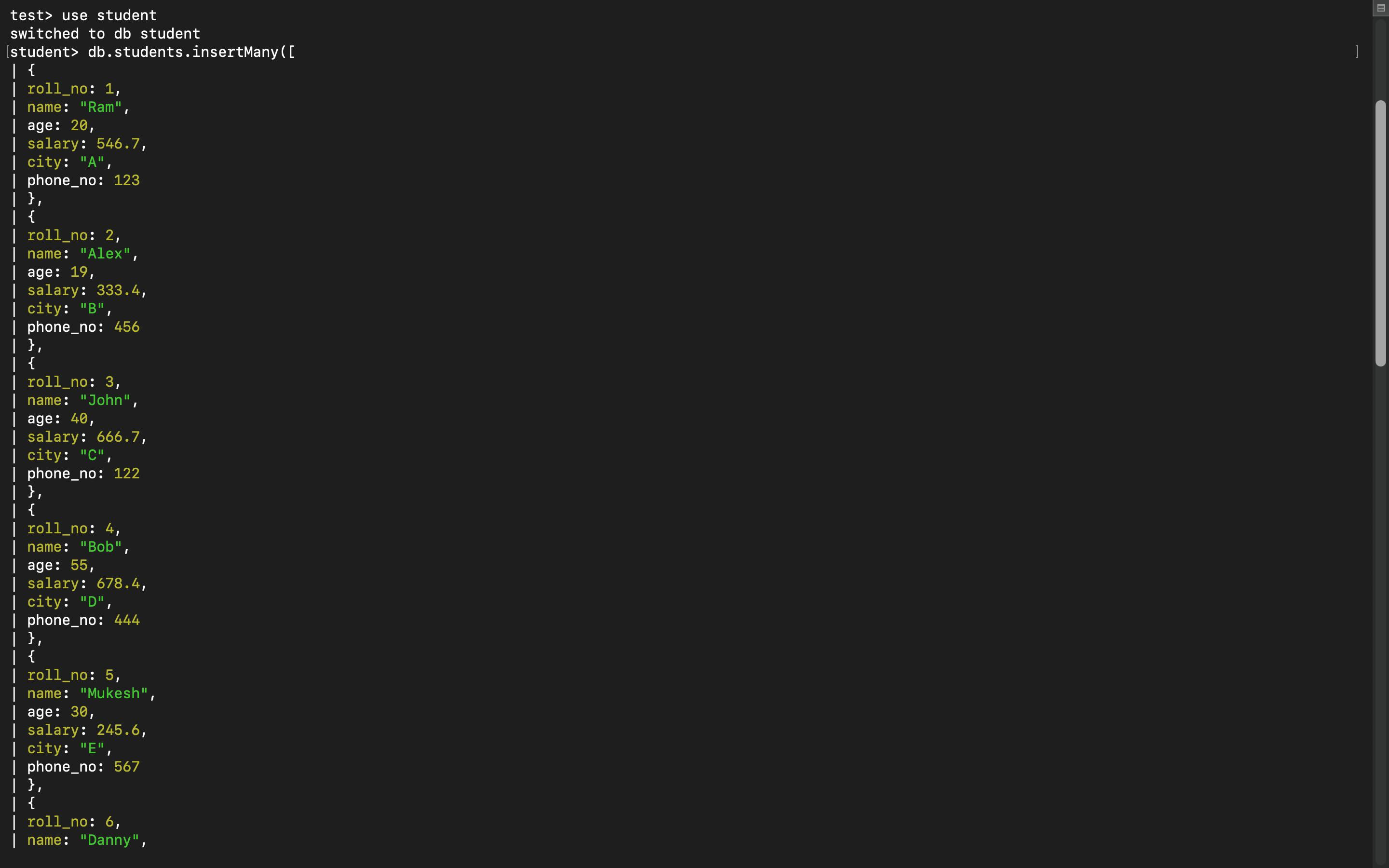
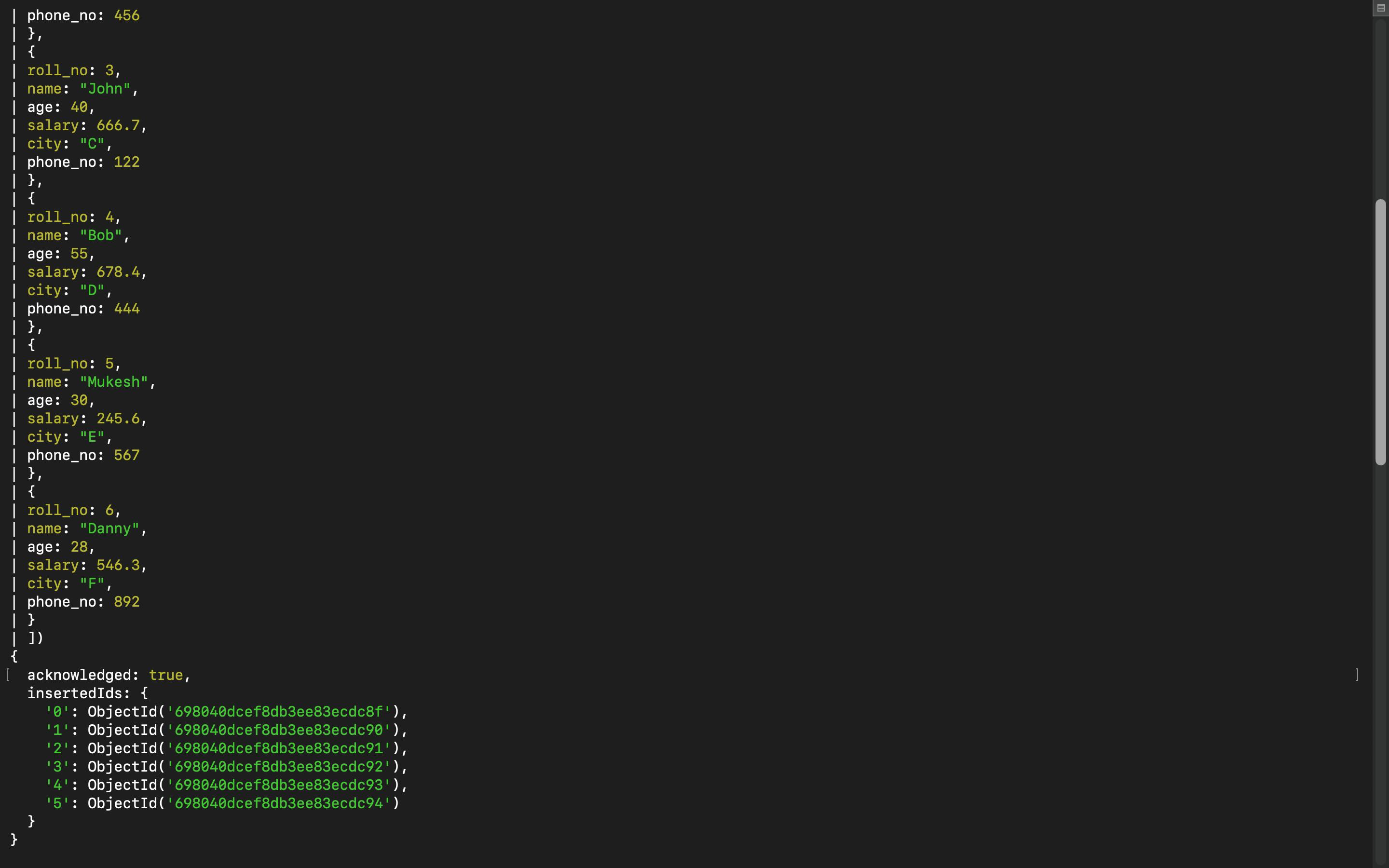
# LAB ASSIGNMENT 2

## Q1. Make a schema and insert 6 documents.

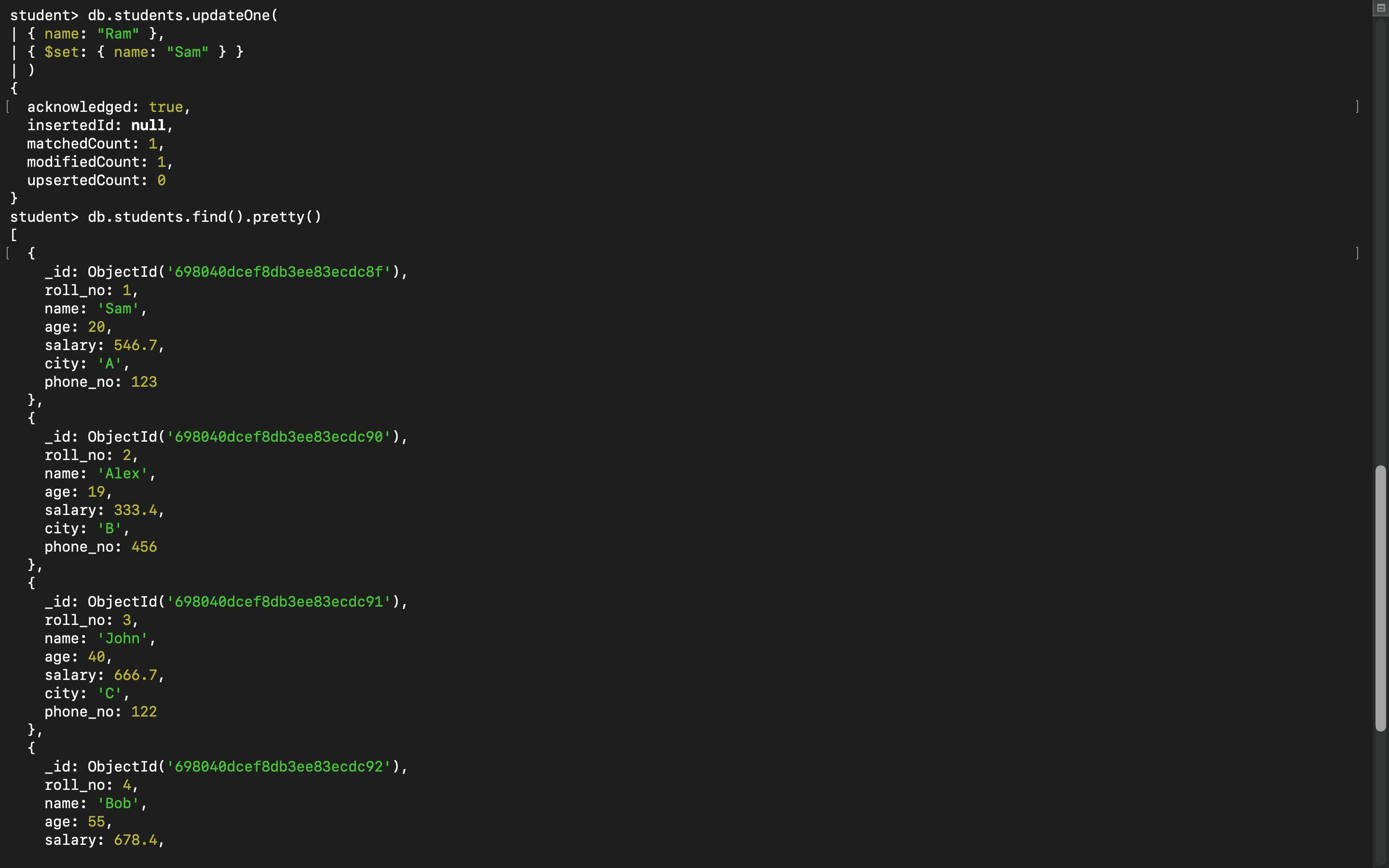
use LabDB  
db.students.insertMany([  
{ roll\_no:1, name:"Ram", age:20, salary:546.7, city:"A", phone\_no:123 },  
{ roll\_no:2, name:"Alex", age:19, salary:333.4, city:"B", phone\_no:456 },  
{ roll\_no:3, name:"John", age:40, salary:666.7, city:"C", phone\_no:122 },  
{ roll\_no:4, name:"Bob", age:55, salary:678.4, city:"D", phone\_no:444 },  
{ roll\_no:5, name:"Mukesh", age:30, salary:245.6, city:"E", phone\_no:567 },  
{ roll\_no:6, name:"Danny", age:28, salary:546.3, city:"F", phone\_no:892 }  
])





## Q2. Update name of RAM to SAM.

db.students.updateOne(  
{ name: "Ram" },  
{ $set: { name: "Sam" } }  
)



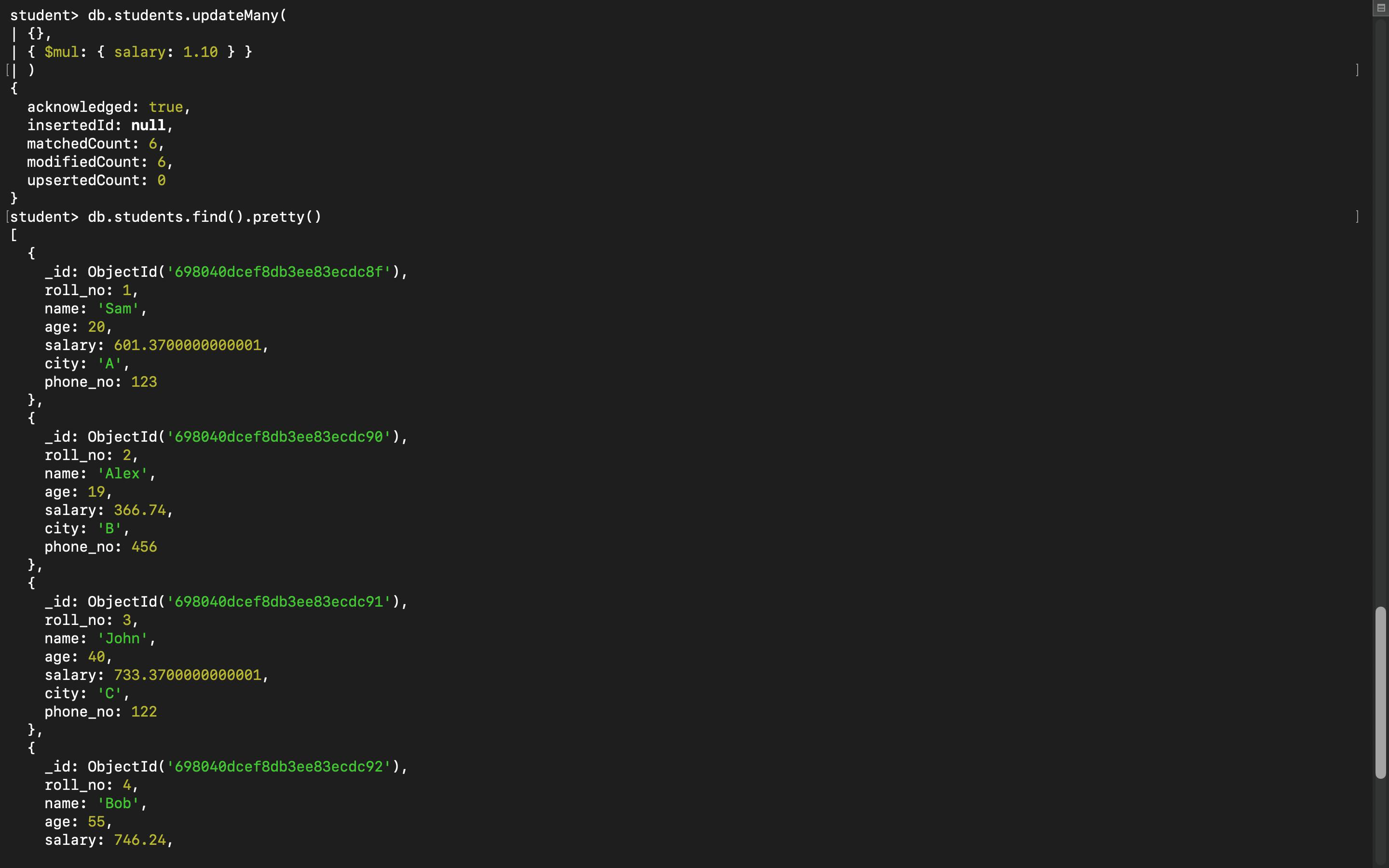
## Q3. Display only the cities present in the collection.

db.students.find(  
{},  
{ \_id: 0, city: 1 }  
)



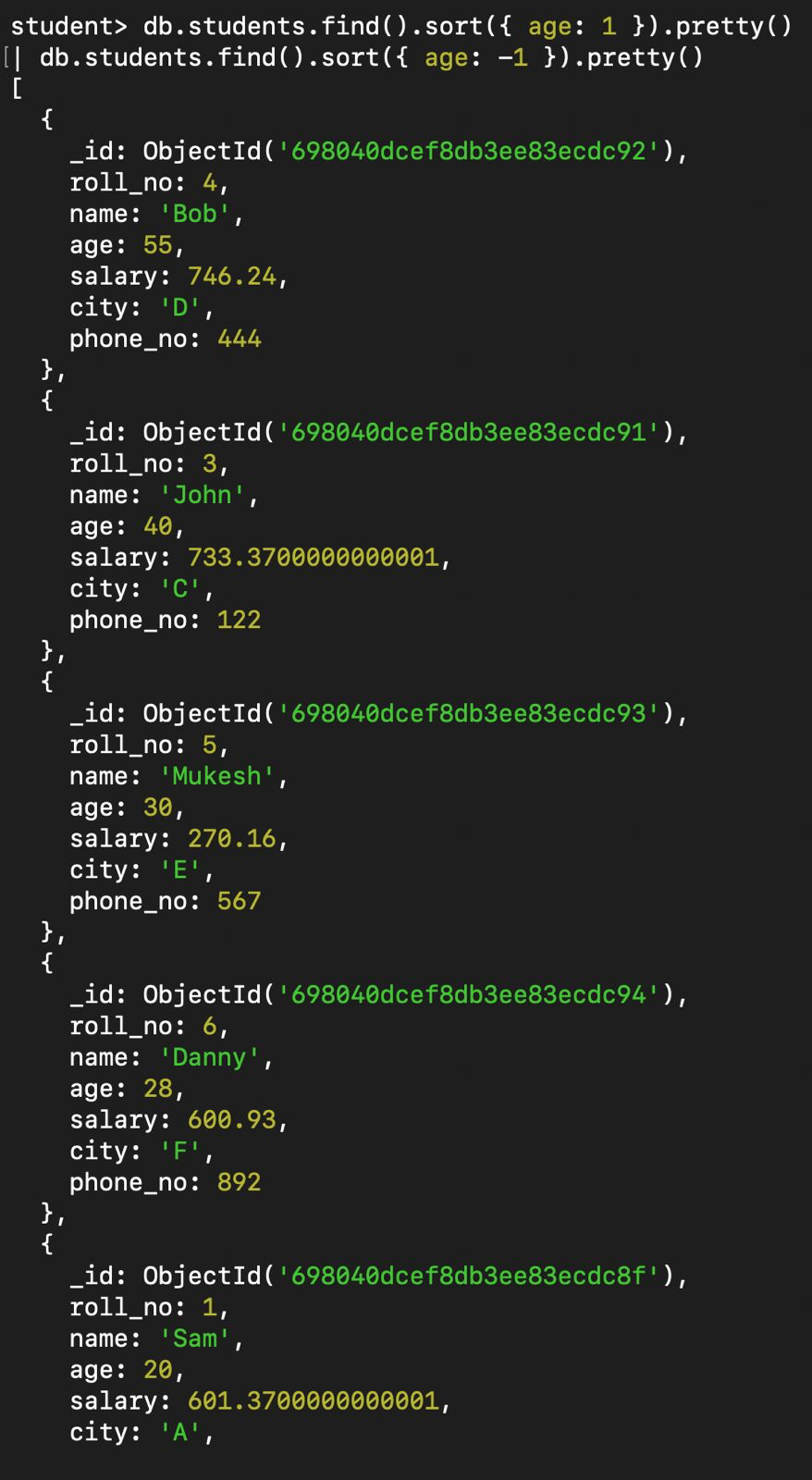
## Q4. Update salary by 10%.

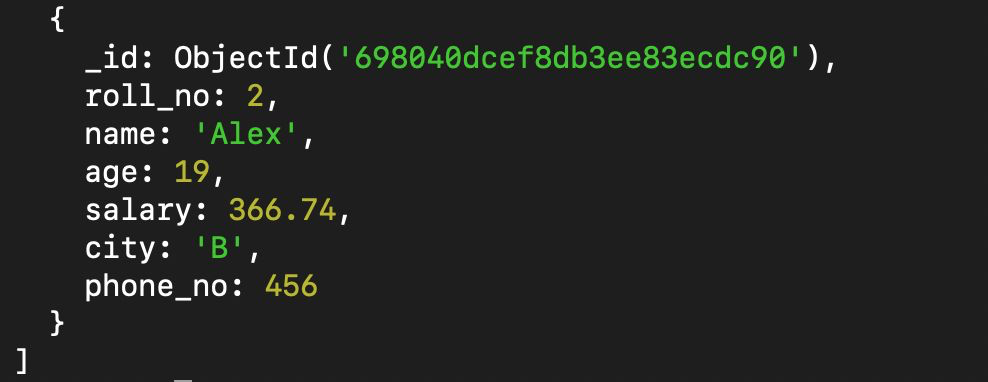
db.students.updateMany(  
{},  
{ $mul: { salary: 1.10 } }  
)



## Q5. Display documents in ascending and descending order of age.

db.students.find().sort({ age: 1 })  
db.students.find().sort({ age: -1 })





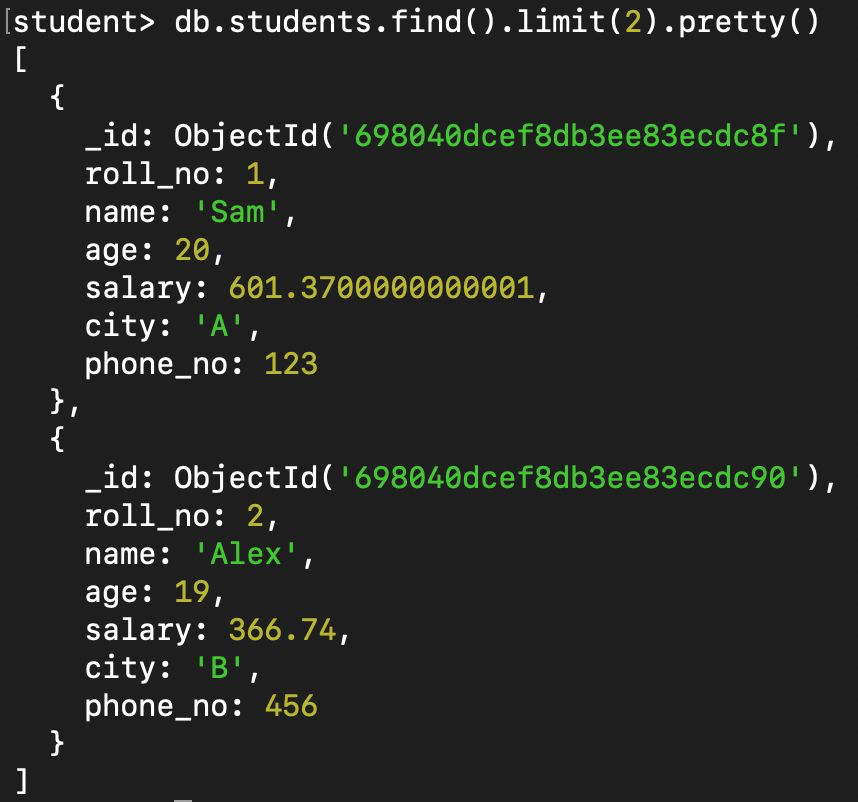
## Q6. Display documents with City A, B, C.

db.students.find(  
{ city: { $in: ["A", "B", "C"] } }  
)



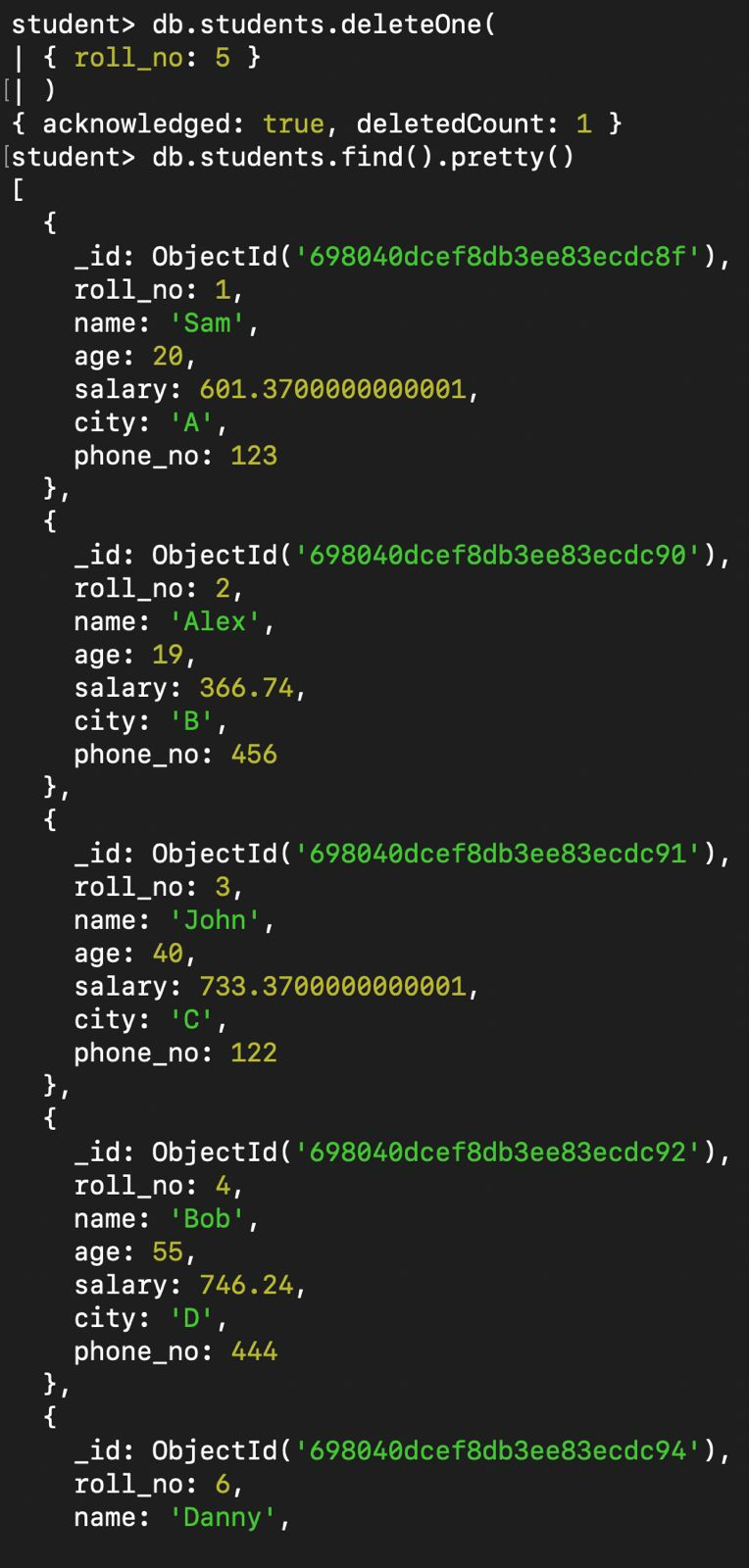
## Q7. Display only two documents from collection.

db.students.find().limit(2)



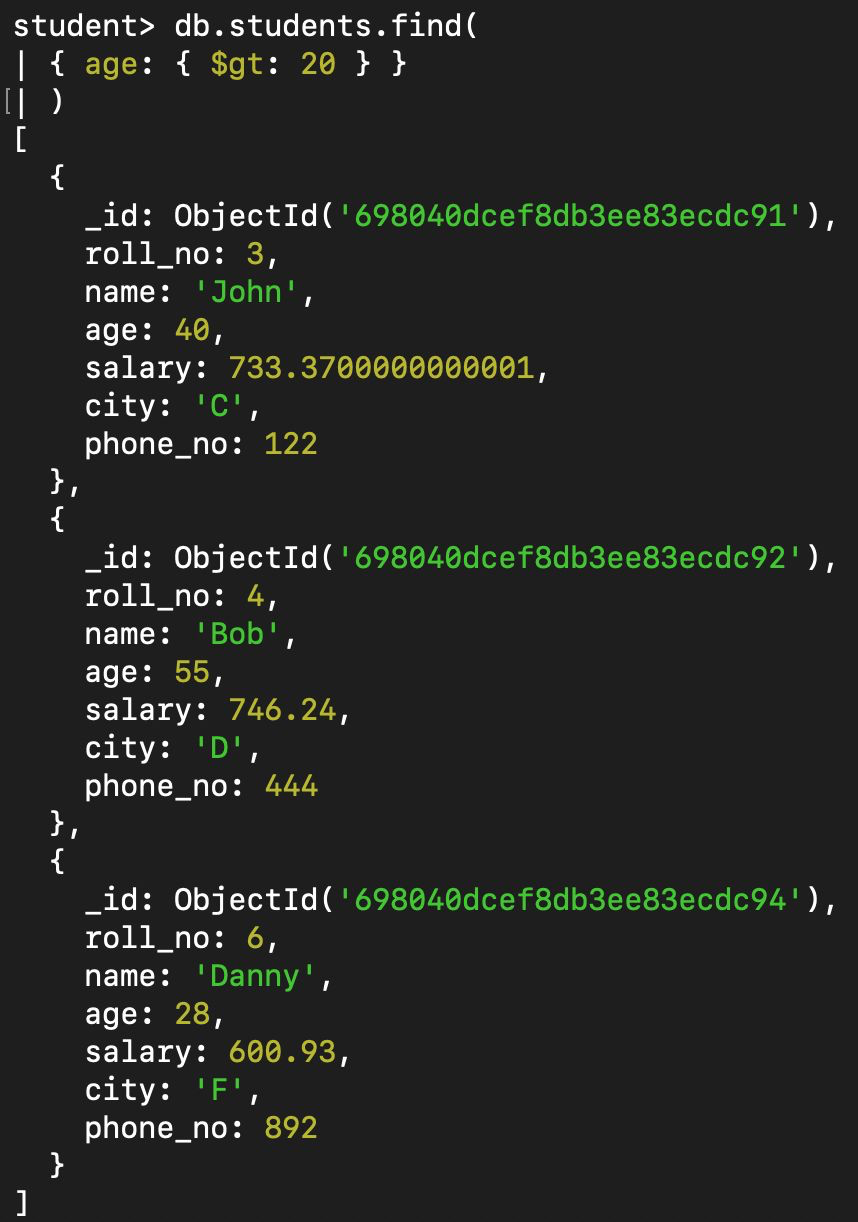
## Q8. Delete document with ROLL\_NO: 5.

db.students.deleteOne(  
{ roll\_no: 5 }  
)



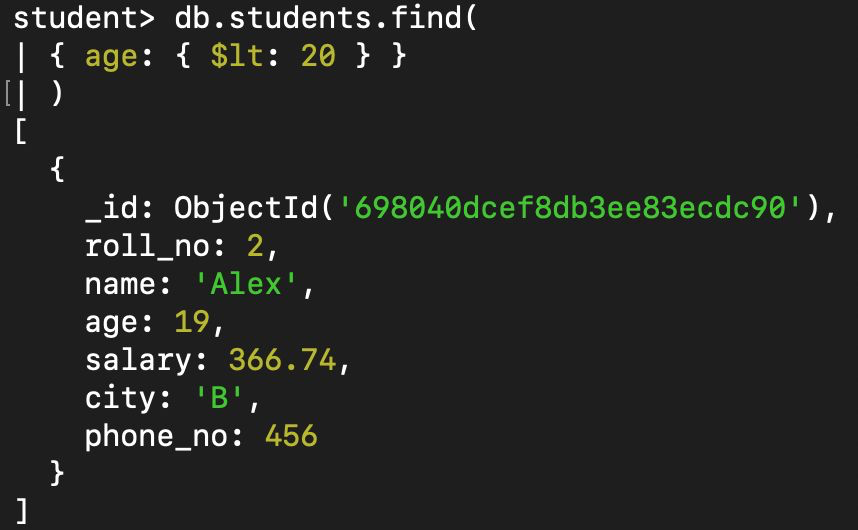
## Q9. Display documents with AGE greater than 20.

db.students.find(  
{ age: { $gt: 20 } }  
)



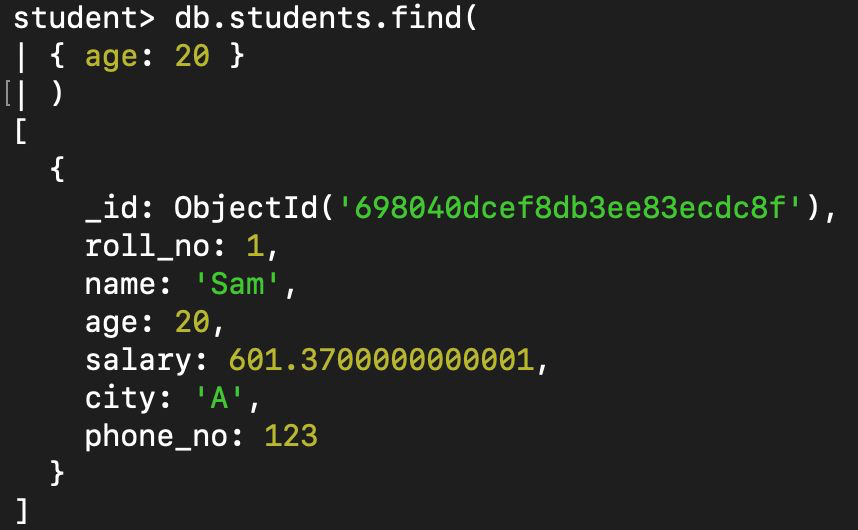
## Q10. Display documents with AGE less than 20.

db.students.find(  
{ age: { $lt: 20 } }  
)



## Q11. Display documents with AGE equal to 20.

db.students.find(  
{ age: 20 }  
)



## Q12. Display documents with AGE not equal to 20.

db.students.find(  
{ age: { $ne: 20 } }  
)



## Q13. Display documents with AGE greater than or equal to 30.

db.students.find(  
{ age: { $gte: 30 } }  
)

