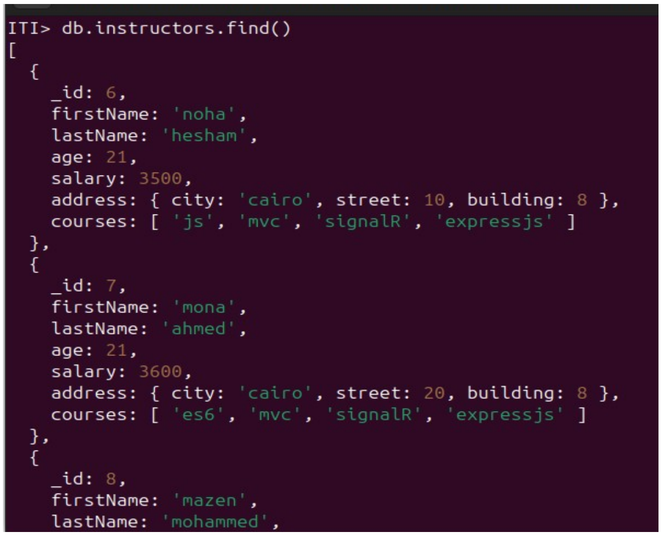
Ahmed Ehab Ahmed
MongoDB Lab2

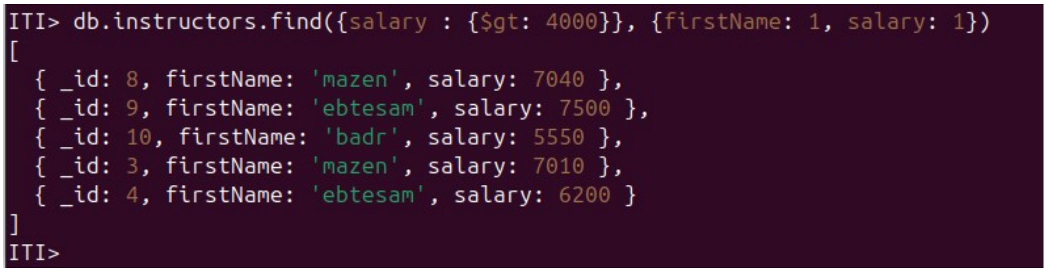
1- run following queries:

a- Display all documents in instructors collection



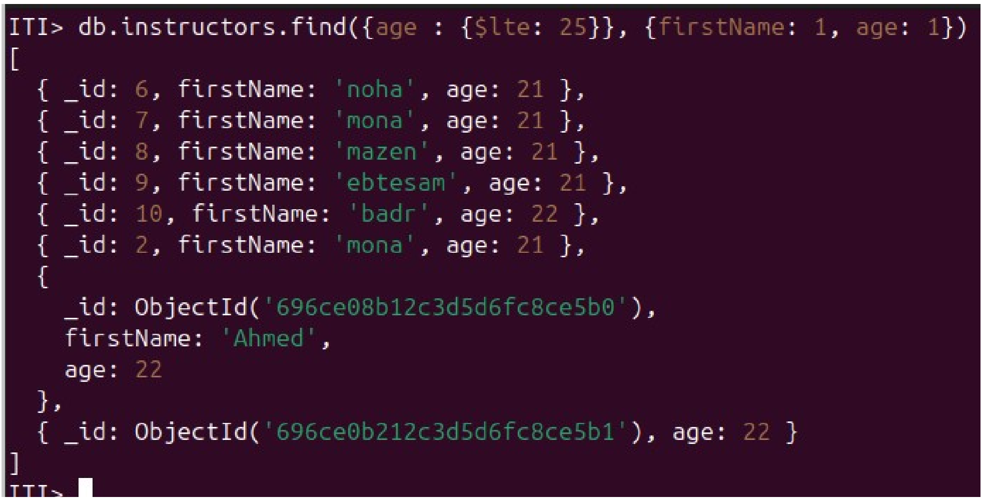
```
ITI> db.instructors.find()
[
  {
    _id: 6,
    firstName: 'noha',
    lastName: 'hesham',
    age: 21,
    salary: 3500,
    address: { city: 'cairo', street: 10, building: 8 },
    courses: [ 'js', 'mvc', 'signalR', 'expressjs' ]
  },
  {
    _id: 7,
    firstName: 'mona',
    lastName: 'ahmed',
    age: 21,
    salary: 3600,
    address: { city: 'cairo', street: 20, building: 8 },
    courses: [ 'es6', 'mvc', 'signalR', 'expressjs' ]
  },
  {
    _id: 8,
    firstName: 'mazen',
    lastName: 'mohammed',
```

b- Display all instructors with salaries greater than 4000 (only show firstName and salary)



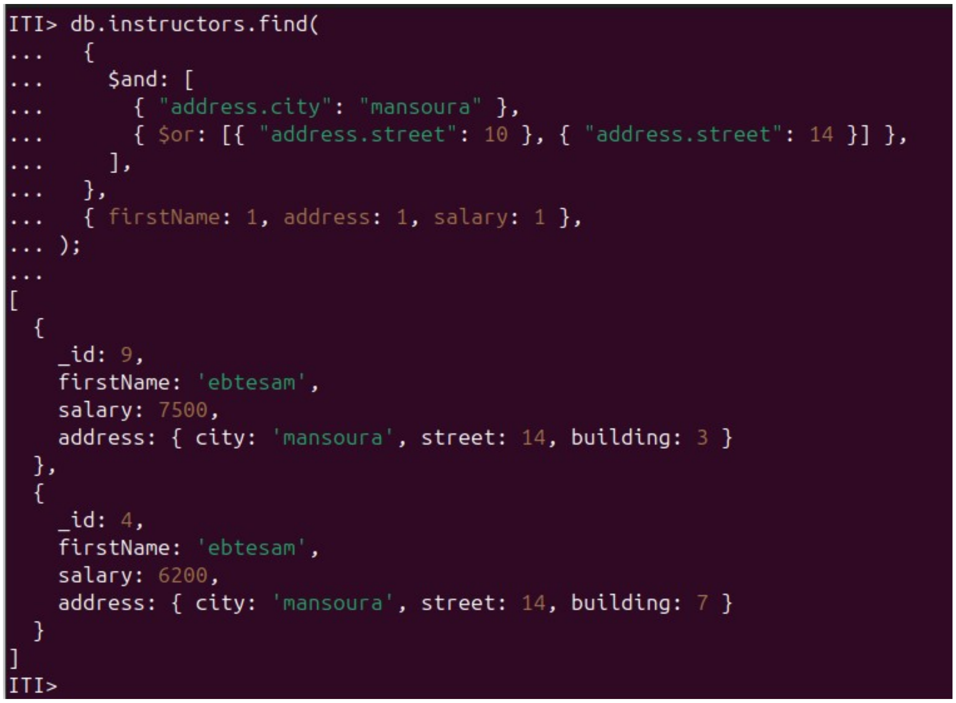
```
ITI> db.instructors.find({salary : {$gt: 4000}}, {firstName: 1, salary: 1})
[
  { _id: 8, firstName: 'mazen', salary: 7040 },
  { _id: 9, firstName: 'ebtesam', salary: 7500 },
  { _id: 10, firstName: 'badr', salary: 5550 },
  { _id: 3, firstName: 'mazen', salary: 7010 },
  { _id: 4, firstName: 'ebtesam', salary: 6200 }
]
ITI>
```

c- Display all instructors with ages less than or equal 25.



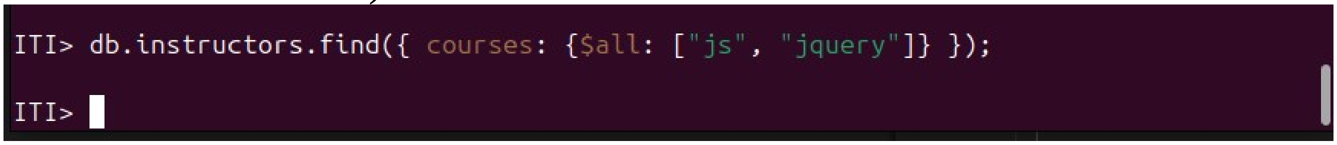
```
ITI> db.instructors.find({age : {$lte: 25}}, {firstName: 1, age: 1})
[
  { _id: 6, firstName: 'noha', age: 21 },
  { _id: 7, firstName: 'mona', age: 21 },
  { _id: 8, firstName: 'mazen', age: 21 },
  { _id: 9, firstName: 'ebtesam', age: 21 },
  { _id: 10, firstName: 'badr', age: 22 },
  { _id: 2, firstName: 'mona', age: 21 },
  {
    _id: ObjectId('696ce08b12c3d5d6fc8ce5b0'),
    firstName: 'Ahmed',
    age: 22
  },
  { _id: ObjectId('696ce0b212c3d5d6fc8ce5b1'), age: 22 }
]
ITI>
```

d- Display all instructors with city mansoura and sreet number 10 or 14 only display (firstname,address,salary).



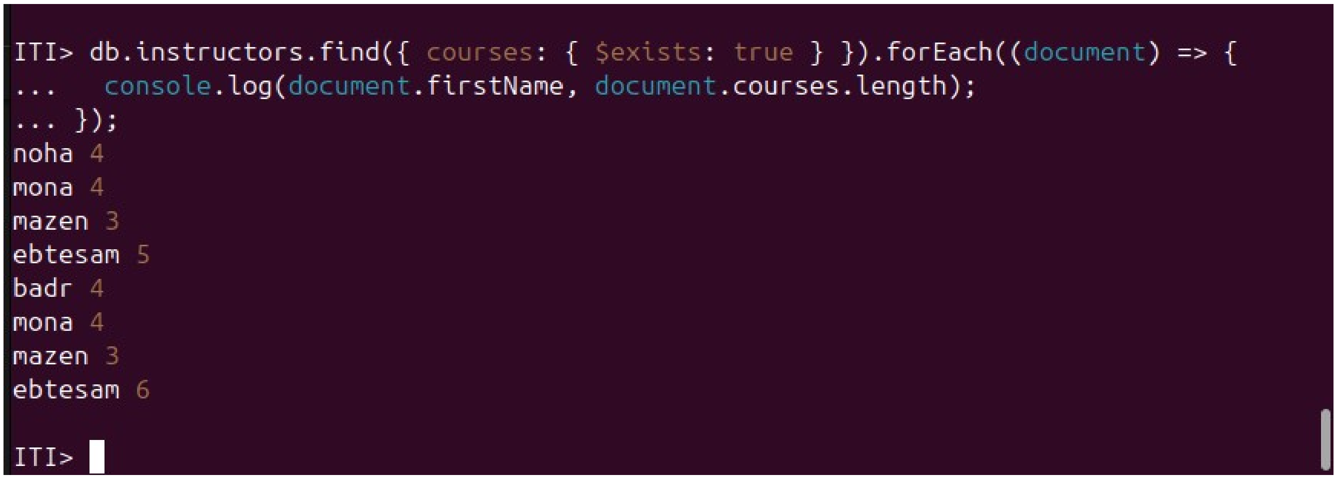
```
ITI> db.instructors.find(
...   {
...     $and: [
...       { "address.city": "mansoura" },
...       { $or: [{ "address.street": 10 }, { "address.street": 14 }] },
...     ],
...   },
...   { firstName: 1, address: 1, salary: 1 },
... );
...
[
  {
    _id: 9,
    firstName: 'ebtesam',
    salary: 7500,
    address: { city: 'mansoura', street: 14, building: 3 }
  },
  {
    _id: 4,
    firstName: 'ebtesam',
    salary: 6200,
    address: { city: 'mansoura', street: 14, building: 7 }
  }
]
ITI>
```

e- Display all instructors that have js and jquery in their courses (both of them not one of them).



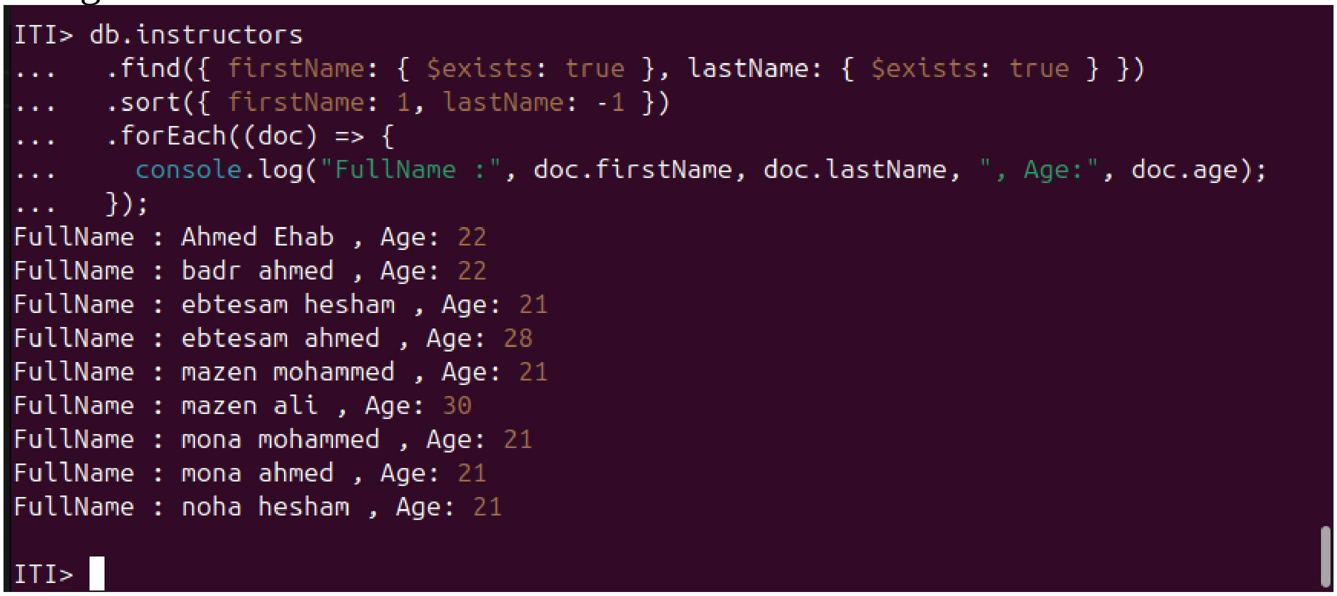
```
ITI> db.instructors.find({ courses: {$all: ["js", "jquery"]} });

ITI> 
```

f- Display number of courses for each instructor and display first name with number of courses.



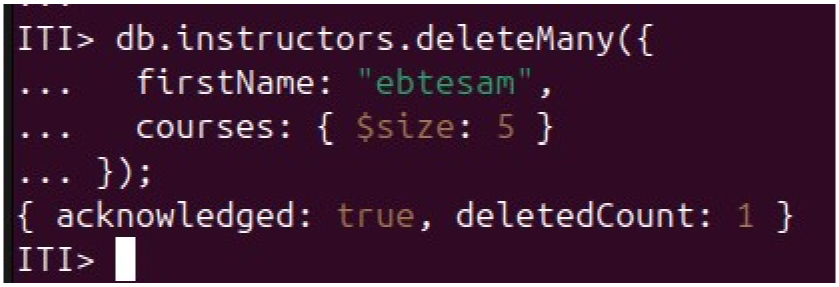
```
ITI> db.instructors.find({ courses: { $exists: true } }).forEach((document) => {
...   console.log(document.firstName, document.courses.length);
... });
noha 4
mona 4
mazen 3
ebtesam 5
badr 4
mona 4
mazen 3
ebtesam 6

ITI> 
```

g- Write mongodb query to get all instructors that have firstName and lastName properties , sort them by firstName ascending then by lastName descending and finally display their data FullName and age Note: use mongoDB sort function



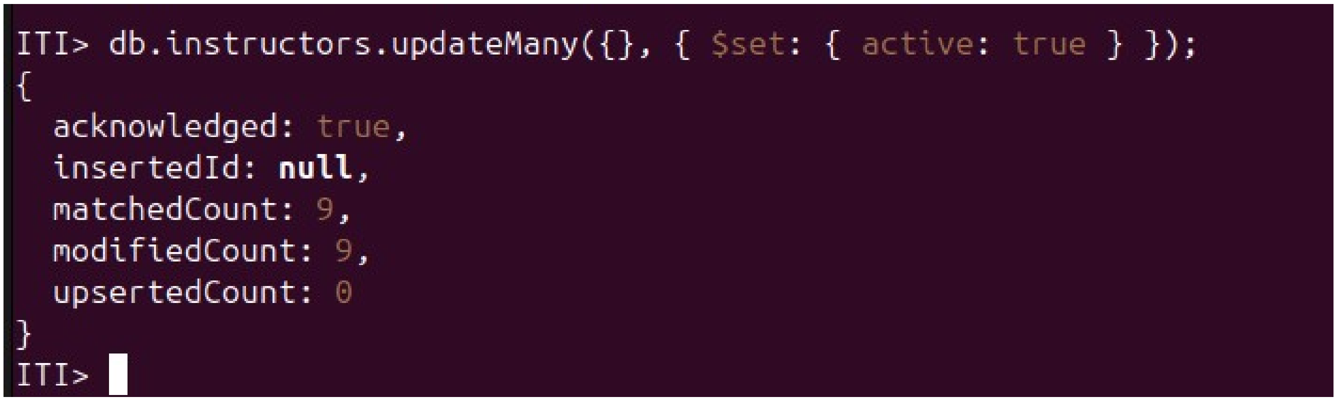
```
ITI> db.instructors
...   .find({ firstName: { $exists: true }, lastName: { $exists: true } })
...   .sort({ firstName: 1, lastName: -1 })
...   .forEach((doc) => {
...     console.log("FullName :", doc.firstName, doc.lastName, ", Age:", doc.age);
...   });
FullName : Ahmed Ehab , Age: 22
FullName : badr ahmed , Age: 22
FullName : ebtesam hesham , Age: 21
FullName : ebtesam ahmed , Age: 28
FullName : mazen mohammed , Age: 21
FullName : mazen ali , Age: 30
FullName : mona mohammed , Age: 21
FullName : mona ahmed , Age: 21
FullName : noha hesham , Age: 21

ITI>
```

h- Delete instructor with first name “ebtesam” and has only 5 courses in courses array



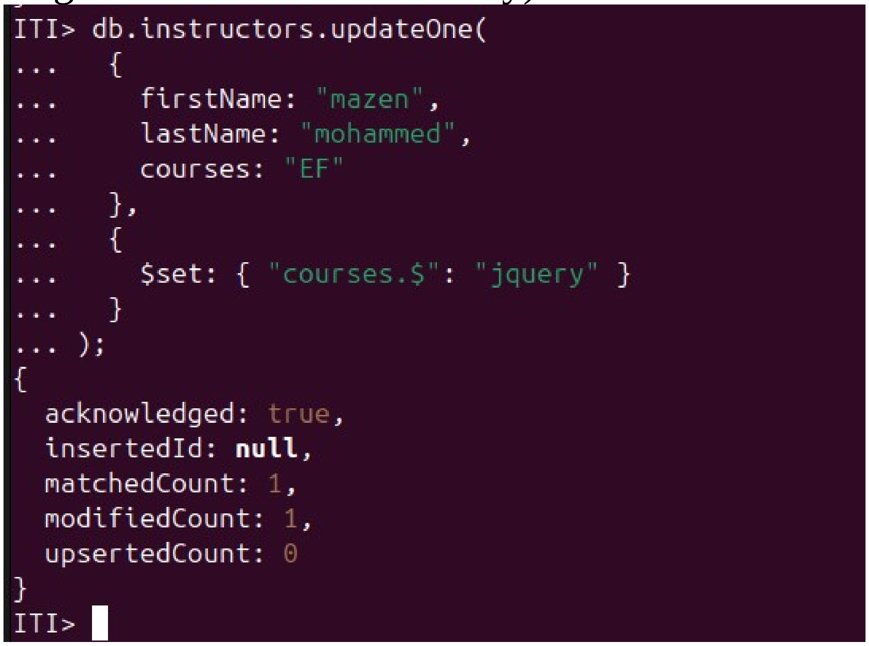
```
ITI> db.instructors.deleteMany({
...   firstName: "ebtesam",
...   courses: { $size: 5 }
... });
{ acknowledged: true, deletedCount: 1 }
ITI>
```

i- Add active property to all instructors and set its value to true.



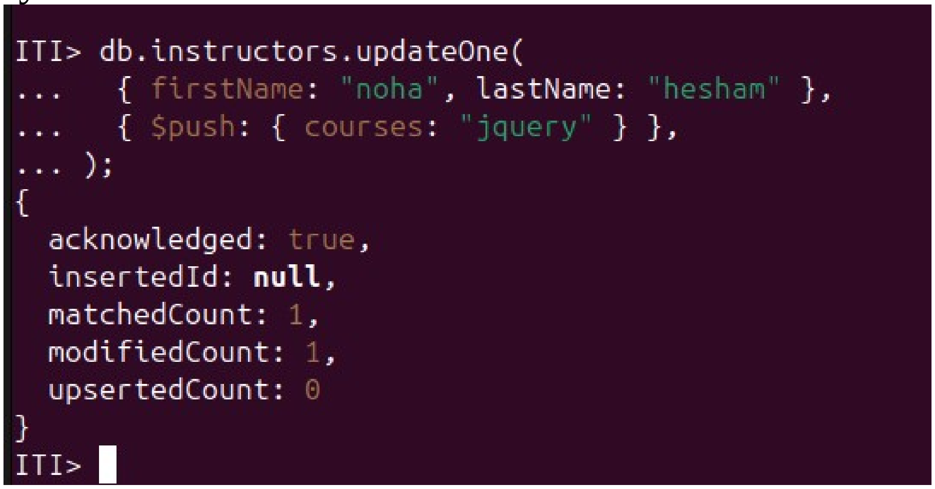
```
ITI> db.instructors.updateMany({}, { $set: { active: true } });
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 9,
  modifiedCount: 9,
  upsertedCount: 0
}
ITI>
```

j- Change “EF” course to “jquery” for “mazen mohammed” instructor (without knowing EF Index in courses array)



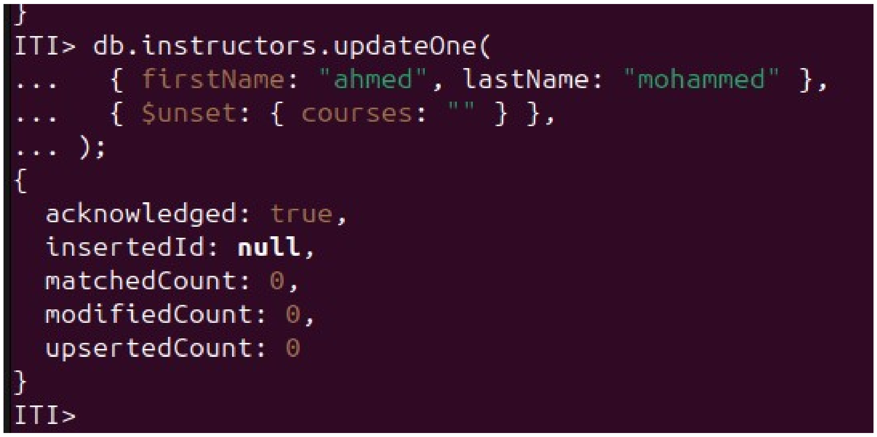
```
ITI> db.instructors.updateOne(
...   {
...     firstName: "mazen",
...     lastName: "mohammed",
...     courses: "EF"
...   },
...   {
...     $set: { "courses.$": "jquery" }
...   }
... );
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
ITI> 
```

k- Add jquery course for “noha hesham”.



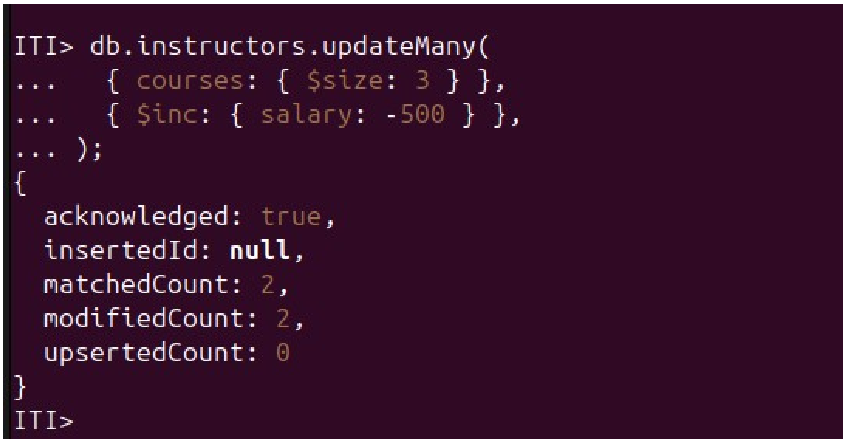
```
ITI> db.instructors.updateOne(
...   { firstName: "noha", lastName: "hesham" },
...   { $push: { courses: "jquery" } },
... );
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
ITI> 
```

l- Remove courses property for “ahmed mohammed ” instructor



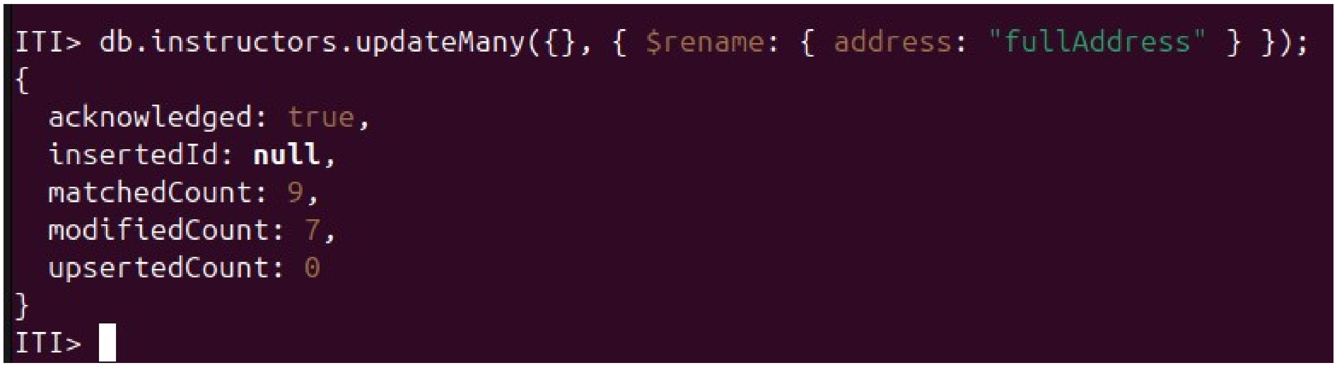
```
}
ITI> db.instructors.updateOne(
...   { firstName: "ahmed", lastName: "mohammed" },
...   { $unset: { courses: "" } },
... );
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 0,
  modifiedCount: 0,
  upsertedCount: 0
}
ITI>
```

m- Decrease salary by 500 for all instructors that has only 3 courses in their list ($inc)



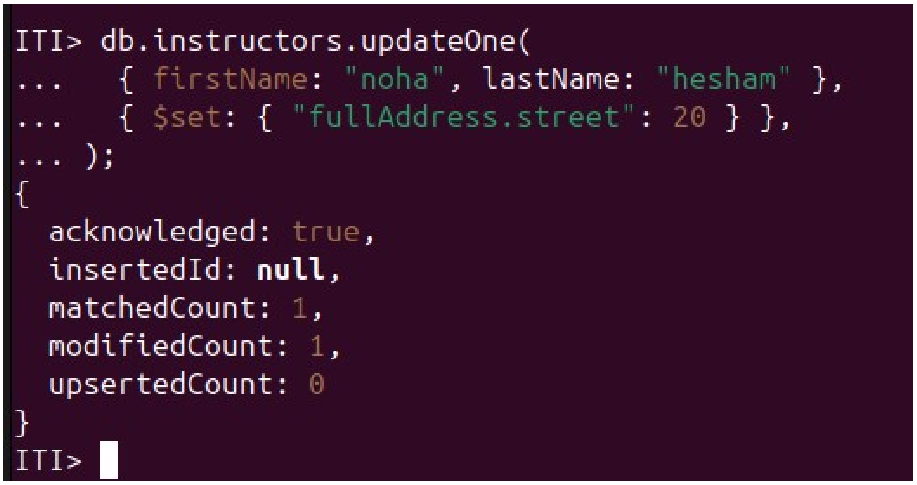
```
ITI> db.instructors.updateMany(
...   { courses: { $size: 3 } },
...   { $inc: { salary: -500 } },
... );
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 2,
  modifiedCount: 2,
  upsertedCount: 0
}
ITI>
```

n- Rename address field for all instructors to fullAddress.



```
ITI> db.instructors.updateMany({}, { $rename: { address: "fullAddress" } });
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 9,
  modifiedCount: 7,
  upsertedCount: 0
}
ITI>
```

o- Change street number for noha hesham to 20



```
ITI> db.instructors.updateOne(
...   { firstName: "noha", lastName: "hesham" },
...   { $set: { "fullAddress.street": 20 } },
... );
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
ITI>
```