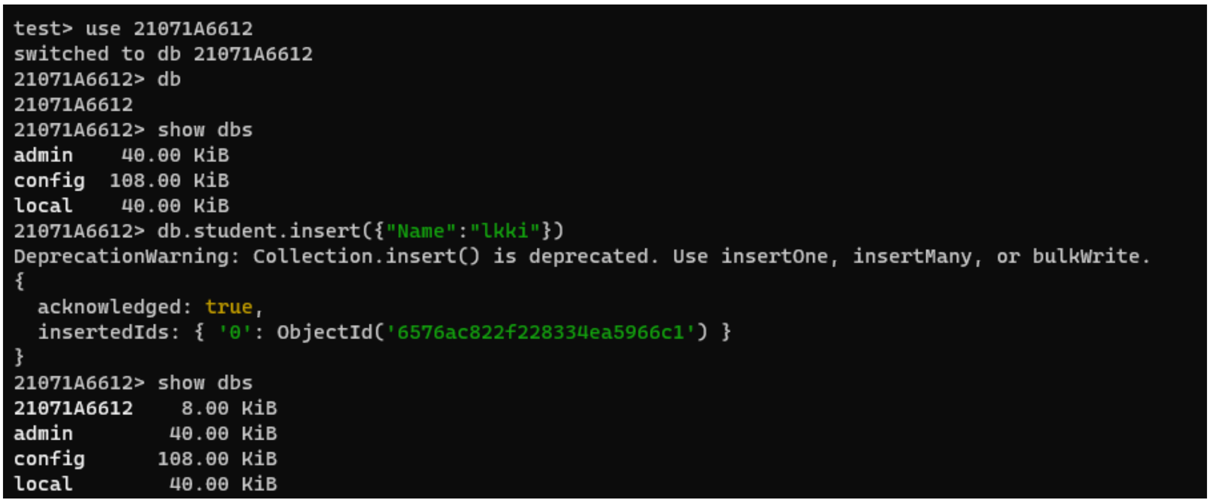
WEEK 12&13

MongoDB

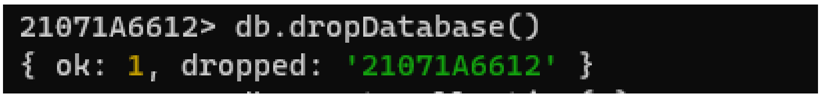
1) Create a database and explore basic database commands.

Create Database:



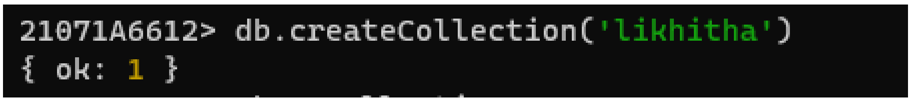
```
test> use 21071A6612
switched to db 21071A6612
21071A6612> db
21071A6612
21071A6612> show dbs
admin     40.00 KiB
config   108.00 KiB
local     40.00 KiB
21071A6612> db.student.insert({"Name":"lkki"})
DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or bulkWrite.
{
  acknowledged: true,
  insertedIds: { '0': ObjectId('6576ac822f228334ea5966c1') }
}
21071A6612> show dbs
21071A6612     8.00 KiB
admin         40.00 KiB
config       108.00 KiB
local         40.00 KiB
```

Drop Database:



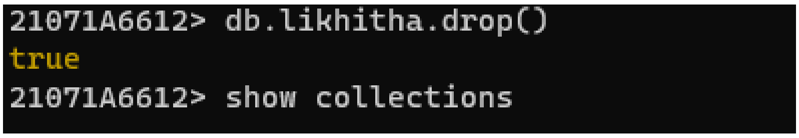
```
21071A6612> db.dropDatabase()
{ ok: 1, dropped: '21071A6612' }
```

Create Collection:



```
21071A6612> db.createCollection('likhitha')
{ ok: 1 }
```

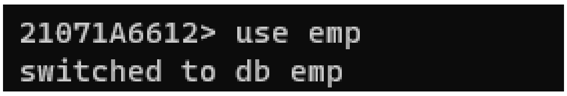
Drop Collection:



```
21071A6612> db.likhitha.drop()
true
21071A6612> show collections
```

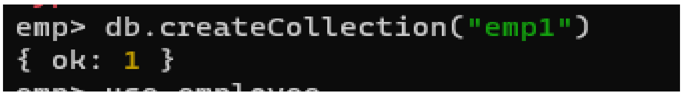
2) Create a database with sample dataset and work with MongoDB Query Language

Create Database:



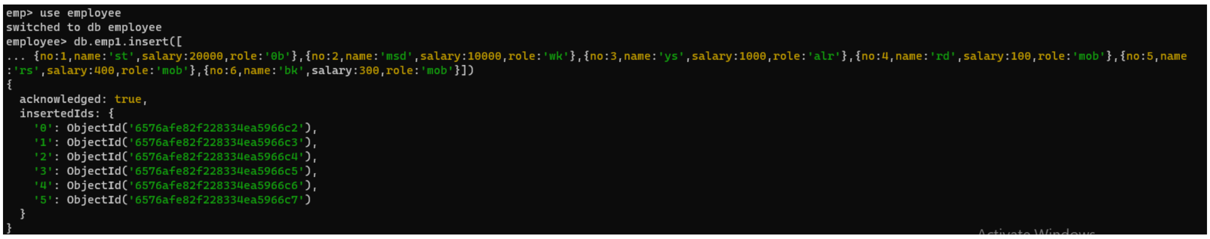
```
21071A6612> use emp
switched to db emp
```

Create Collection:



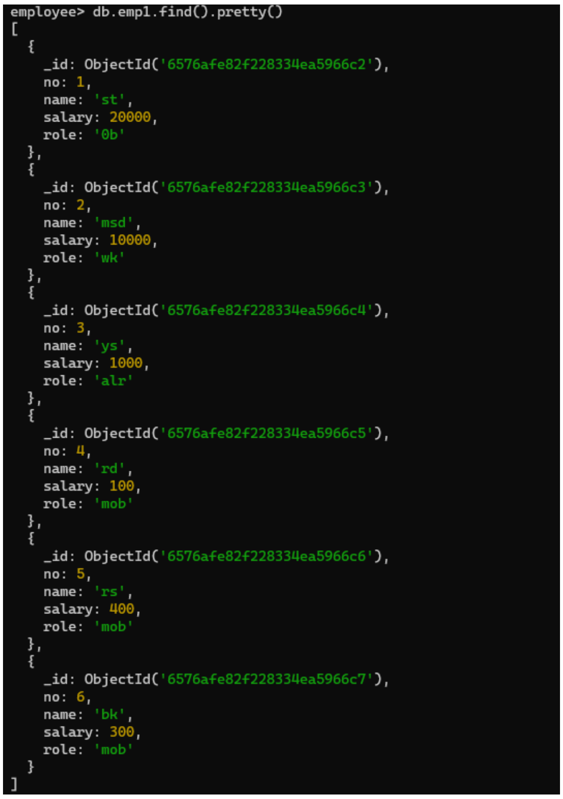
```
emp> db.createCollection("emp1")
{ ok: 1 }
```

Insert Records Into EMP1 Collection:



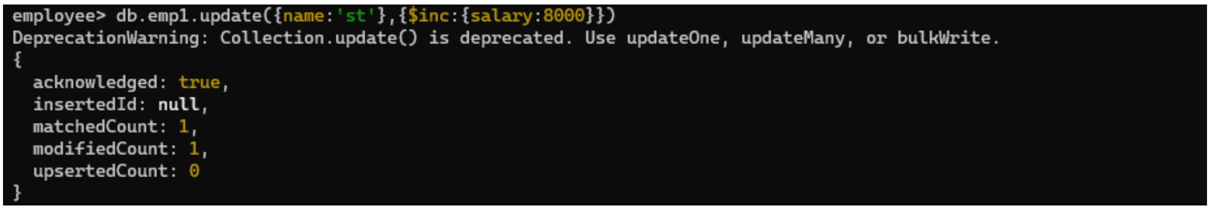
```
emp> use employee
switched to db employee
employee> db.emp1.insert([
... {no:1,name:'st',salary:20000,role:'0b'},{no:2,name:'msd',salary:10000,role:'wk'},{no:3,name:'ys',salary:1000,role:'alr'},{no:4,name:'rd',salary:100,role:'mob'},{no:5,name
:'rs',salary:400,role:'mob'},{no:6,name:'bk',salary:300,role:'mob'}])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('6576afe82f228334ea5966c2'),
    '1': ObjectId('6576afe82f228334ea5966c3'),
    '2': ObjectId('6576afe82f228334ea5966c4'),
    '3': ObjectId('6576afe82f228334ea5966c5'),
    '4': ObjectId('6576afe82f228334ea5966c6'),
    '5': ObjectId('6576afe82f228334ea5966c7')
  }
}
```

Display Data in Proper Format:

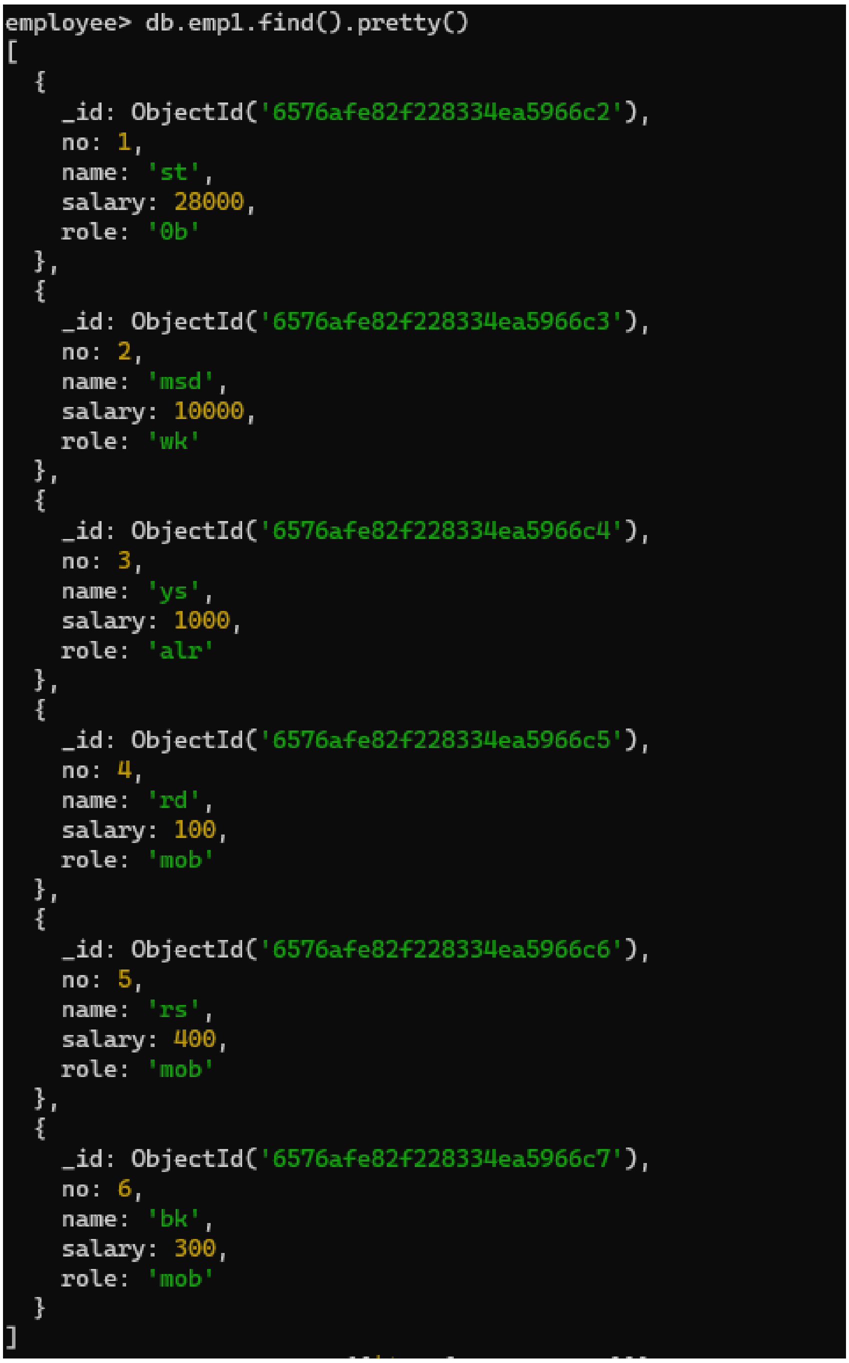


```
employee> db.emp1.find().pretty()
[
  {
    _id: ObjectId('6576afe82f228334ea5966c2'),
    no: 1,
    name: 'st',
    salary: 20000,
    role: '0b'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c3'),
    no: 2,
    name: 'msd',
    salary: 10000,
    role: 'wk'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c4'),
    no: 3,
    name: 'ys',
    salary: 1000,
    role: 'alr'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c5'),
    no: 4,
    name: 'rd',
    salary: 100,
    role: 'mob'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c6'),
    no: 5,
    name: 'rs',
    salary: 400,
    role: 'mob'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c7'),
    no: 6,
    name: 'bk',
    salary: 300,
    role: 'mob'
  }
]
```

Update Salary Of Employee where Name is "ST" by +8000

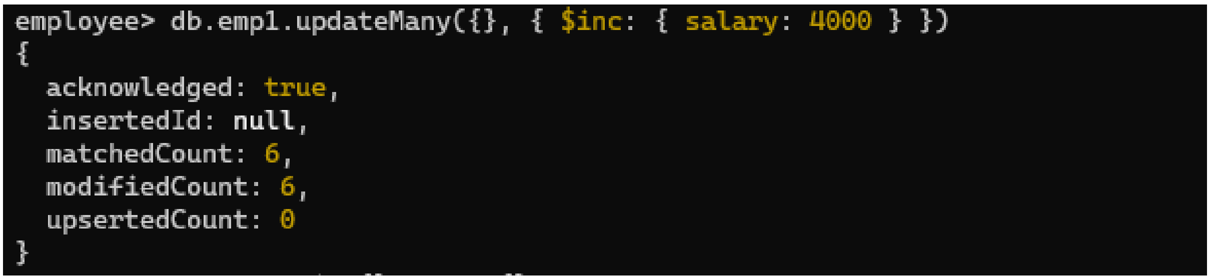


```
employee> db.emp1.update({name:'st'},{$inc:{salary:8000}})
DeprecationWarning: Collection.update() is deprecated. Use updateOne, updateMany, or bulkWrite.
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

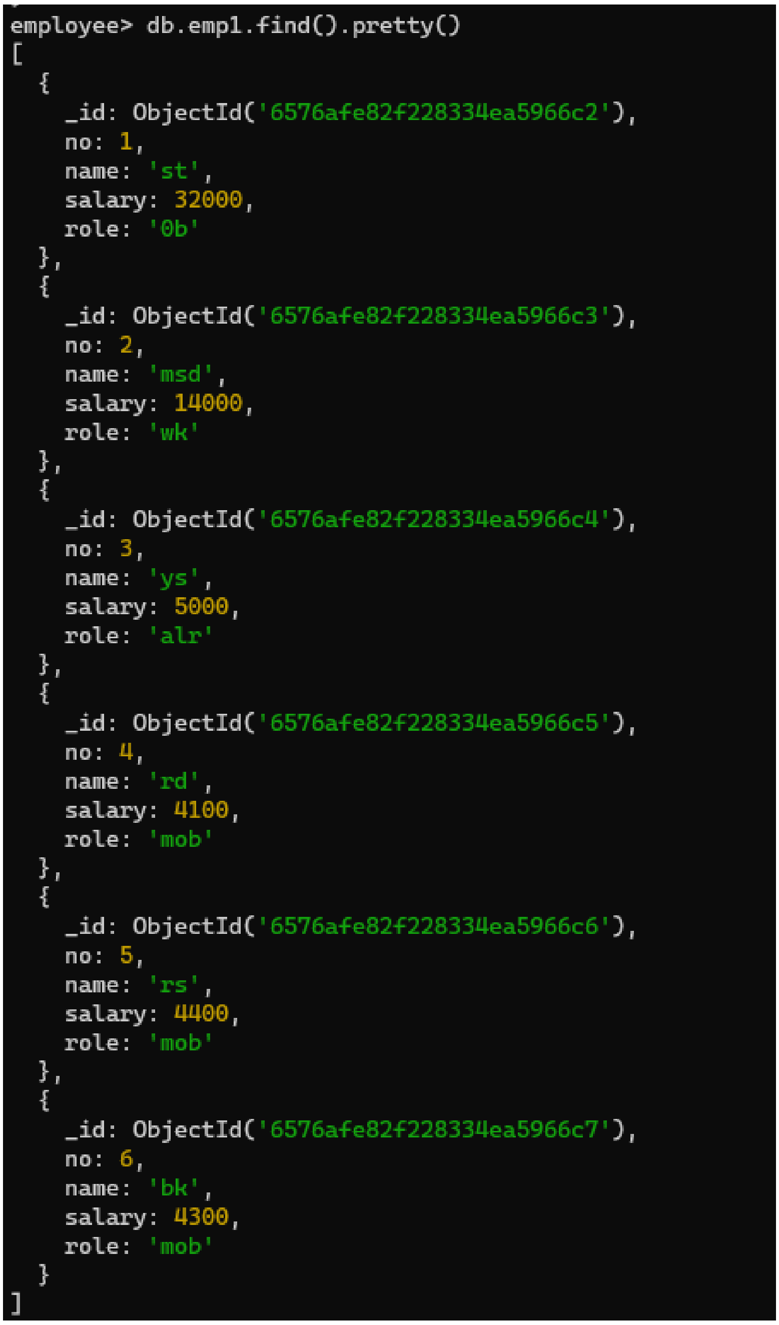


```
employee> db.emp1.find().pretty()
[
  {
    _id: ObjectId('6576afe82f228334ea5966c2'),
    no: 1,
    name: 'st',
    salary: 28000,
    role: '0b'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c3'),
    no: 2,
    name: 'msd',
    salary: 10000,
    role: 'wk'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c4'),
    no: 3,
    name: 'ys',
    salary: 1000,
    role: 'alr'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c5'),
    no: 4,
    name: 'rd',
    salary: 100,
    role: 'mob'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c6'),
    no: 5,
    name: 'rs',
    salary: 400,
    role: 'mob'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c7'),
    no: 6,
    name: 'bk',
    salary: 300,
    role: 'mob'
  }
]
```

Update Salary Of All Employee by giving an increment of +4000 each

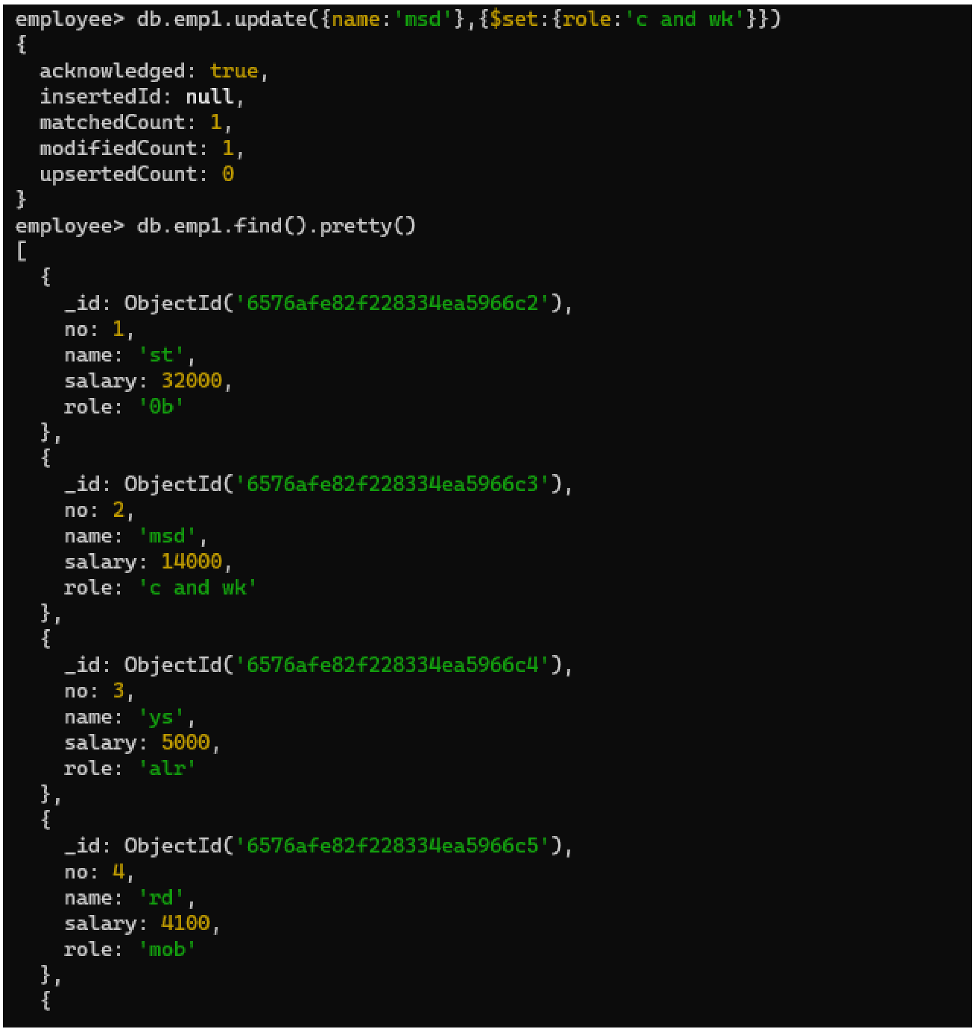


```
employee> db.emp1.updateMany({}, { $inc: { salary: 4000 } })
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 6,
  modifiedCount: 6,
  upsertedCount: 0
}
```



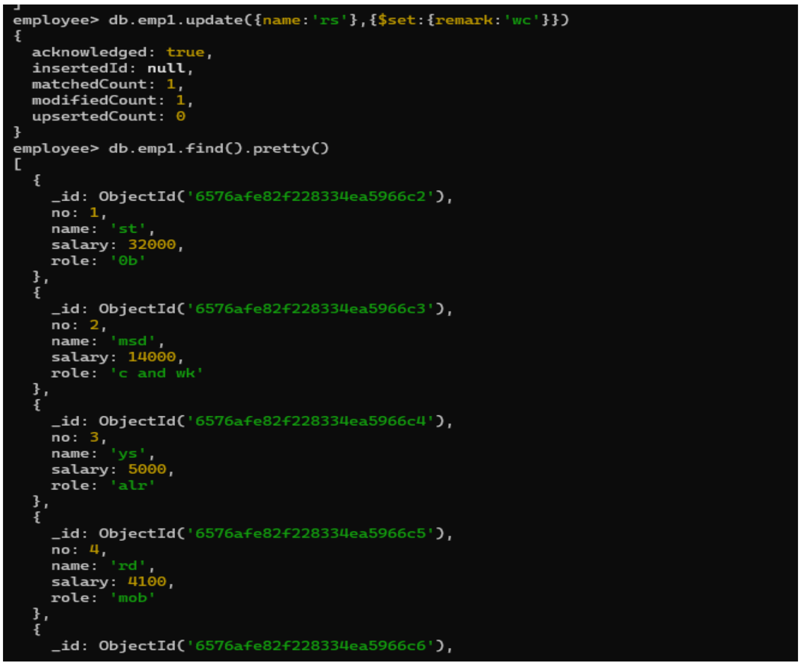
```
employee> db.emp1.find().pretty()
[
  {
    _id: ObjectId('6576afe82f228334ea5966c2'),
    no: 1,
    name: 'st',
    salary: 32000,
    role: '0b'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c3'),
    no: 2,
    name: 'msd',
    salary: 14000,
    role: 'wk'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c4'),
    no: 3,
    name: 'ys',
    salary: 5000,
    role: 'alr'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c5'),
    no: 4,
    name: 'rd',
    salary: 4100,
    role: 'mob'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c6'),
    no: 5,
    name: 'rs',
    salary: 4400,
    role: 'mob'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c7'),
    no: 6,
    name: 'bk',
    salary: 4300,
    role: 'mob'
  }
]
```

Update role of "MSD" as "C and WK"

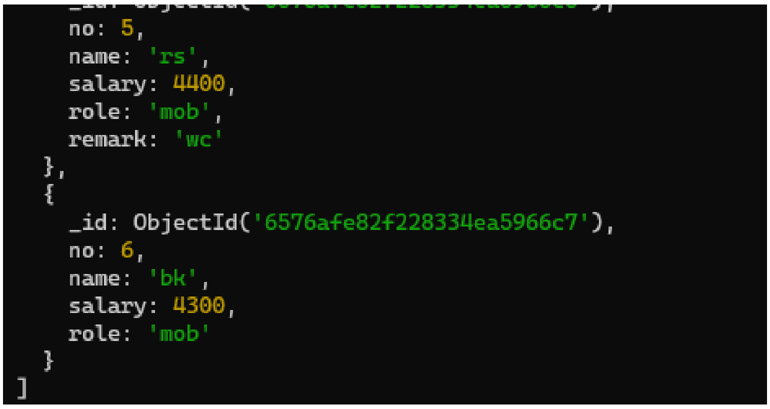


```
employee> db.emp1.update({name:'msd'},{$set:{role:'c and wk'}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
employee> db.emp1.find().pretty()
[
  {
    _id: ObjectId('6576afe82f228334ea5966c2'),
    no: 1,
    name: 'st',
    salary: 32000,
    role: '0b'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c3'),
    no: 2,
    name: 'msd',
    salary: 14000,
    role: 'c and wk'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c4'),
    no: 3,
    name: 'ys',
    salary: 5000,
    role: 'alr'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c5'),
    no: 4,
    name: 'rd',
    salary: 4100,
    role: 'mob'
  },
  {
```

Add a New Field remark to document with name "RS" set Remark as WC

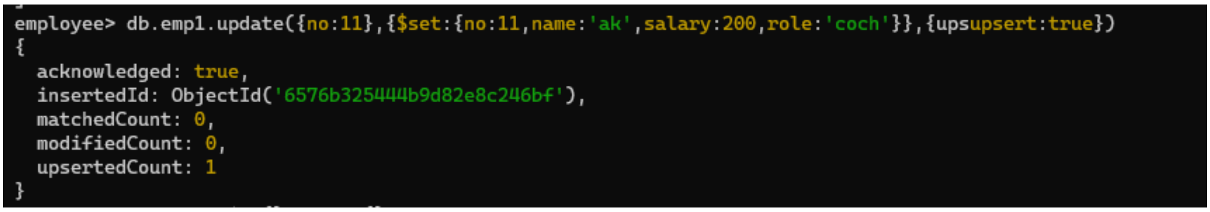


```
employee> db.emp1.update({name:'rs'},{$set:{remark:'wc'}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
employee> db.emp1.find().pretty()
[
  {
    _id: ObjectId('6576afe82f228334ea5966c2'),
    no: 1,
    name: 'st',
    salary: 32000,
    role: '0b'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c3'),
    no: 2,
    name: 'msd',
    salary: 14000,
    role: 'c and wk'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c4'),
    no: 3,
    name: 'ys',
    salary: 5000,
    role: 'alr'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c5'),
    no: 4,
    name: 'rd',
    salary: 4100,
    role: 'mob'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c6'),
```

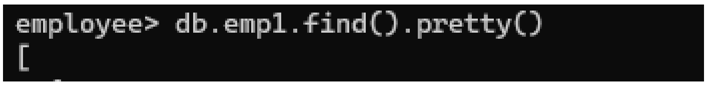


```
    no: 5,
    name: 'rs',
    salary: 4400,
    role: 'mob',
    remark: 'wc'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c7'),
    no: 6,
    name: 'bk',
    salary: 4300,
    role: 'mob'
  }
]
```

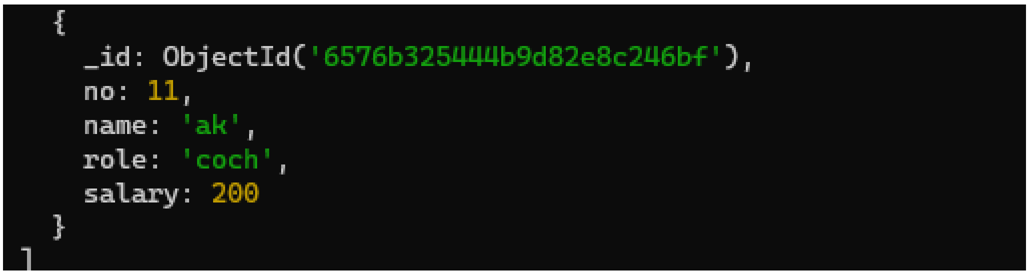
Add a New Field As Number 11,name AK, Salary 200,role coch without using insert statement.But for Doing So You should have a Record Added with number 11.



```
employee> db.emp1.update({no:11},{$set:{no:11,name:'ak',salary:200,role:'coch'}},{upsupsert:true})
{
  acknowledged: true,
  insertedId: ObjectId('6576b325444b9d82e8c246bf'),
  matchedCount: 0,
  modifiedCount: 0,
  upsertedCount: 1
}
```

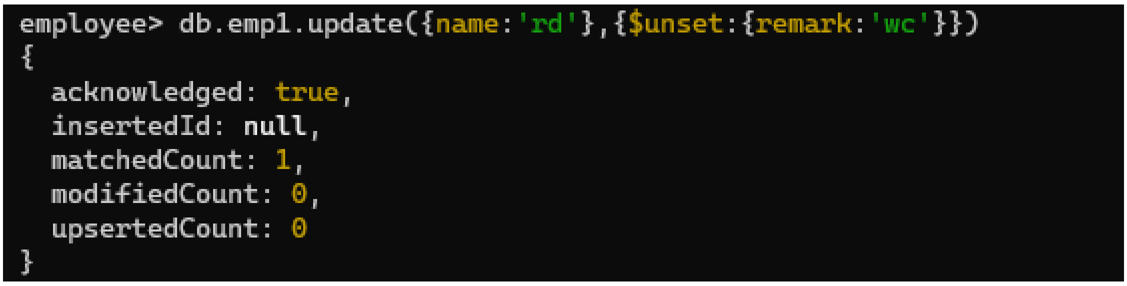


```
employee> db.emp1.find().pretty()
[
```

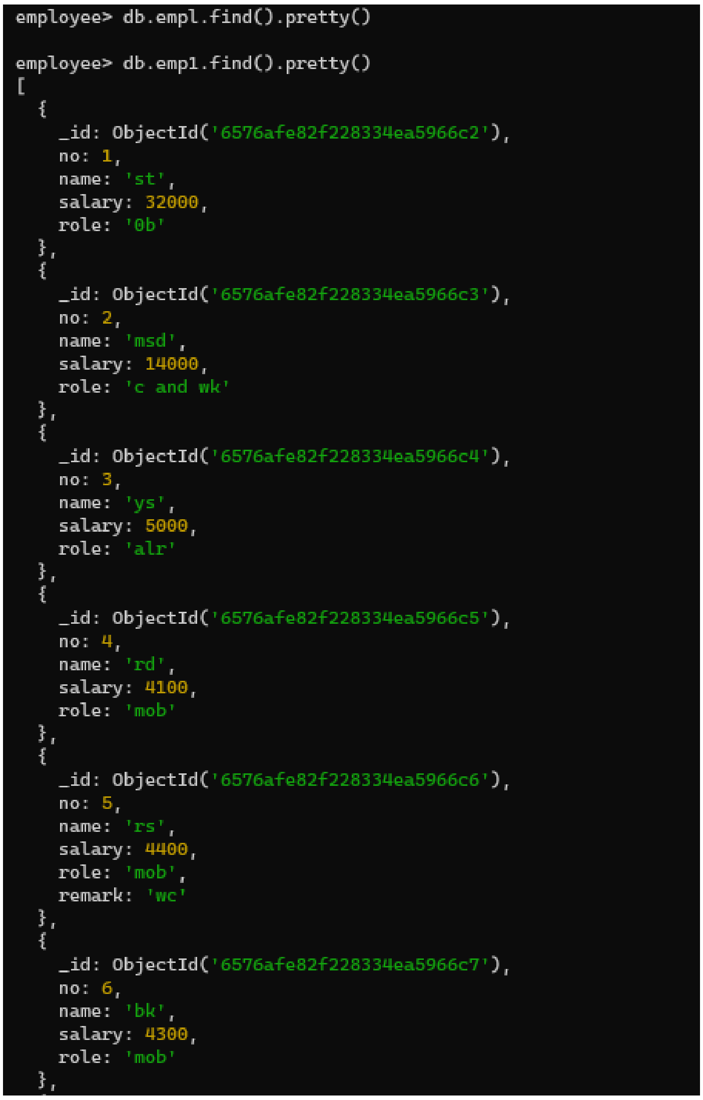


```
  {
    _id: ObjectId('6576b325444b9d82e8c246bf'),
    no: 11,
    name: 'ak',
    role: 'coch',
    salary: 200
  }
]
```

Remove added New Field (REMARK)

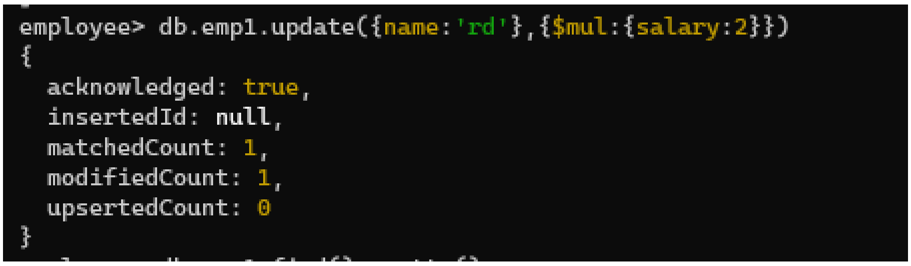


```
employee> db.emp1.update({name:'rd'},{$unset:{remark:'wc'}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 0,
  upsertedCount: 0
}
```

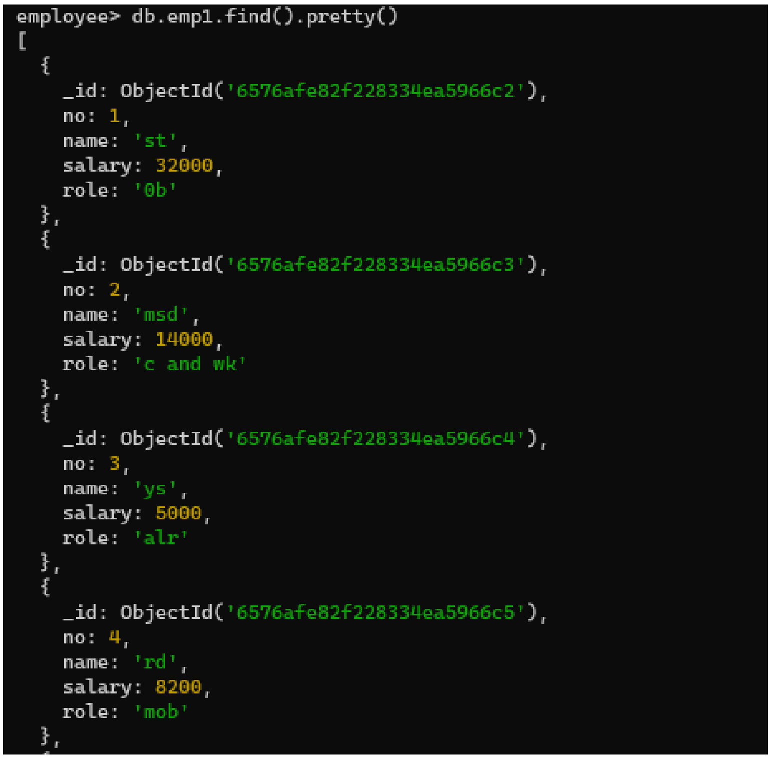


```
employee> db.empl.find().pretty()

employee> db.emp1.find().pretty()
[
  {
    _id: ObjectId('6576afe82f228334ea5966c2'),
    no: 1,
    name: 'st',
    salary: 32000,
    role: '0b'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c3'),
    no: 2,
    name: 'msd',
    salary: 14000,
    role: 'c and wk'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c4'),
    no: 3,
    name: 'ys',
    salary: 5000,
    role: 'alr'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c5'),
    no: 4,
    name: 'rd',
    salary: 4100,
    role: 'mob'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c6'),
    no: 5,
    name: 'rs',
    salary: 4400,
    role: 'mob',
    remark: 'wc'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c7'),
    no: 6,
    name: 'bk',
    salary: 4300,
    role: 'mob'
  },
```

Update the Field "RD" by multiplying with salary by 2

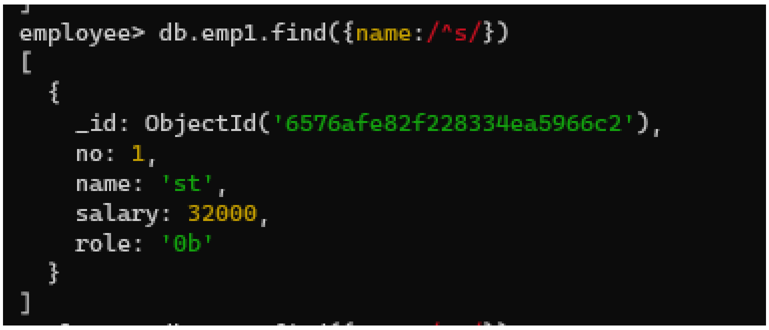


```
employee> db.emp1.update({name:'rd'},{$mul:{salary:2}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```



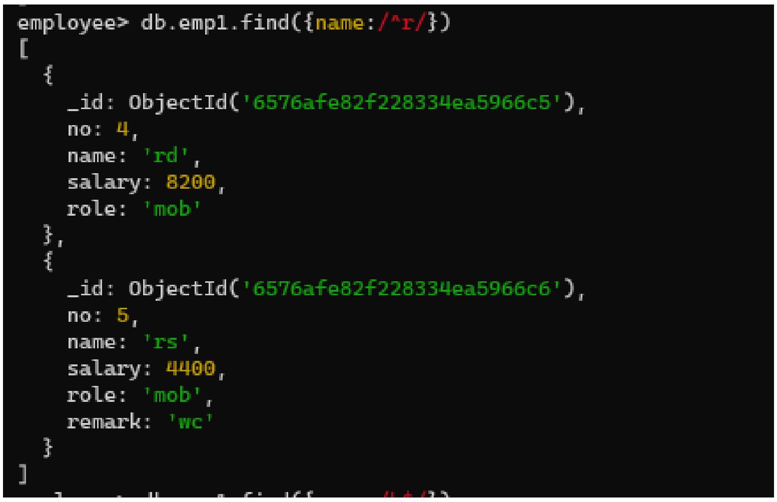
```
employee> db.emp1.find().pretty()
[
  {
    _id: ObjectId('6576afe82f228334ea5966c2'),
    no: 1,
    name: 'st',
    salary: 32000,
    role: '0b'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c3'),
    no: 2,
    name: 'msd',
    salary: 14000,
    role: 'c and wk'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c4'),
    no: 3,
    name: 'ys',
    salary: 5000,
    role: 'alr'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c5'),
    no: 4,
    name: 'rd',
    salary: 8200,
    role: 'mob'
  },
```

To Find Document From the empl collection where name begins with S



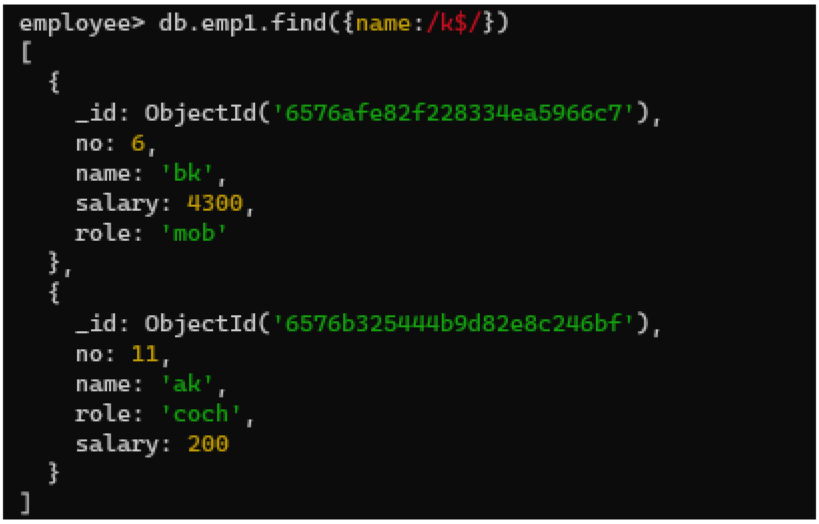
```
employee> db.emp1.find({name:/^s/})
[
  {
    _id: ObjectId('6576afe82f228334ea5966c2'),
    no: 1,
    name: 'st',
    salary: 32000,
    role: '0b'
  }
]
```

To Find Document From the empl collection where name begins with r



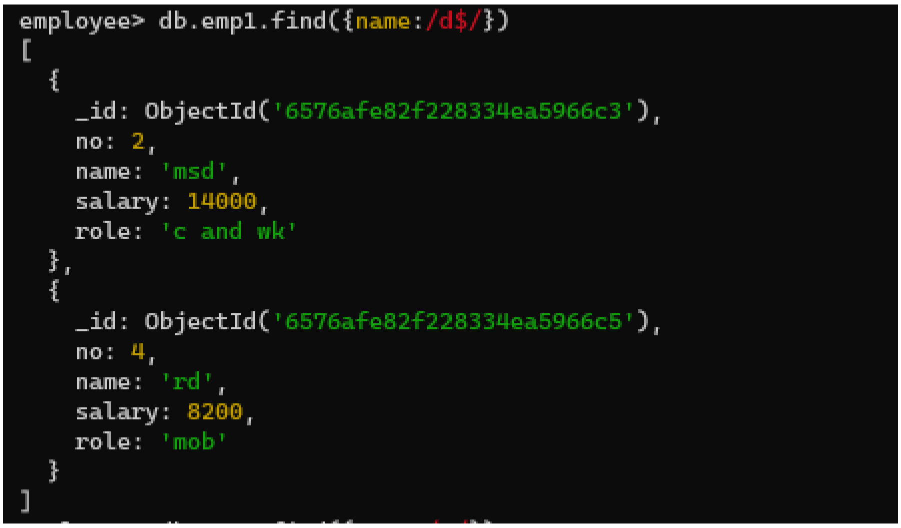
```
employee> db.emp1.find({name:/^r/})
[
  {
    _id: ObjectId('6576afe82f228334ea5966c5'),
    no: 4,
    name: 'rd',
    salary: 8200,
    role: 'mob'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c6'),
    no: 5,
    name: 'rs',
    salary: 4400,
    role: 'mob',
    remark: 'wc'
  }
]
```

To Find Document From the empl collection where name ends with K



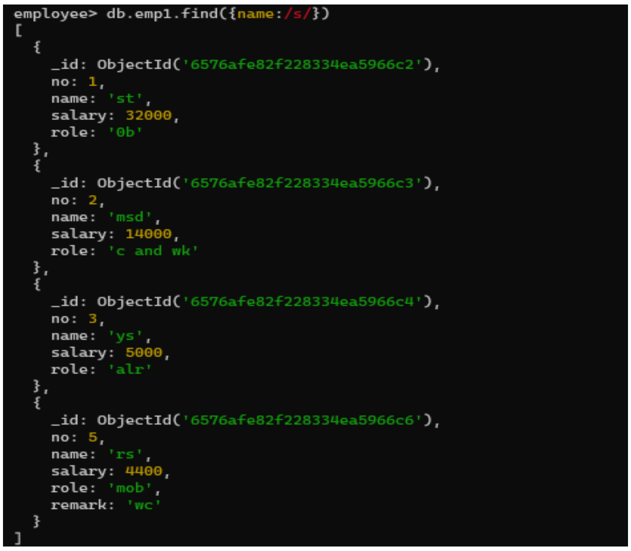
```
employee> db.emp1.find({name:/k$/})
[
  {
    _id: ObjectId('6576afe82f228334ea5966c7'),
    no: 6,
    name: 'bk',
    salary: 4300,
    role: 'mob'
  },
  {
    _id: ObjectId('6576b325444b9d82e8c246bf'),
    no: 11,
    name: 'ak',
    role: 'coch',
    salary: 200
  }
]
```

To Find Document From the empl collection where name ends with D



```
employee> db.emp1.find({name:/d$/})
[
  {
    _id: ObjectId('6576afe82f228334ea5966c3'),
    no: 2,
    name: 'msd',
    salary: 14000,
    role: 'c and wk'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c5'),
    no: 4,
    name: 'rd',
    salary: 8200,
    role: 'mob'
  }
]
```

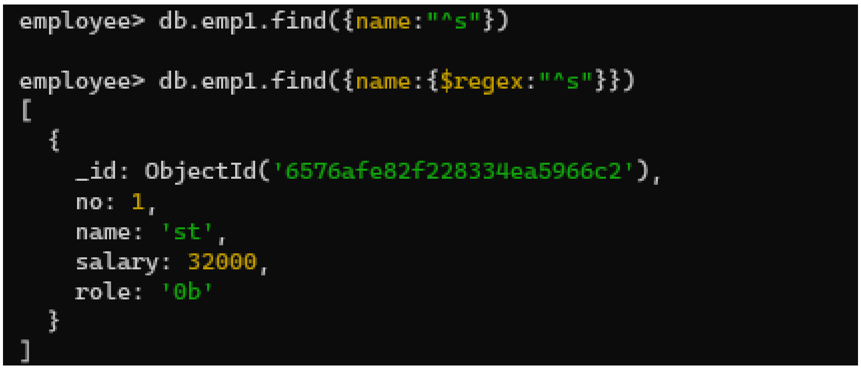
To Find Document From the empl collection where name has S in any position



```
employee> db.emp1.find({name:/s/})
[
  {
    _id: ObjectId('6576afe82f228334ea5966c2'),
    no: 1,
    name: 'st',
    salary: 32000,
    role: '0b'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c3'),
    no: 2,
    name: 'msd',
    salary: 14000,
    role: 'c and wk'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c4'),
    no: 3,
    name: 'ys',
    salary: 5000,
    role: 'alr'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c6'),
    no: 5,
    name: 'rs',
    salary: 4400,
    role: 'mob',
    remark: 'wc'
  }
]
```

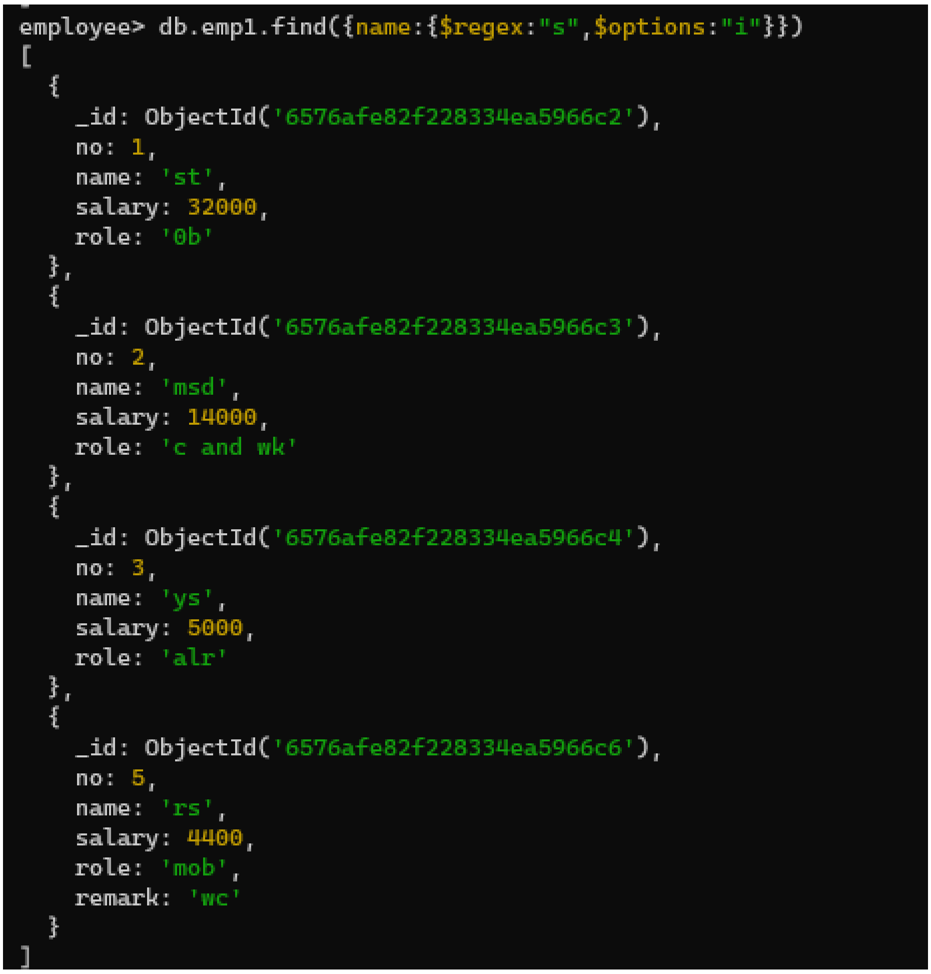
Regular Expression

To Find Document From the empl collection where name begins with S



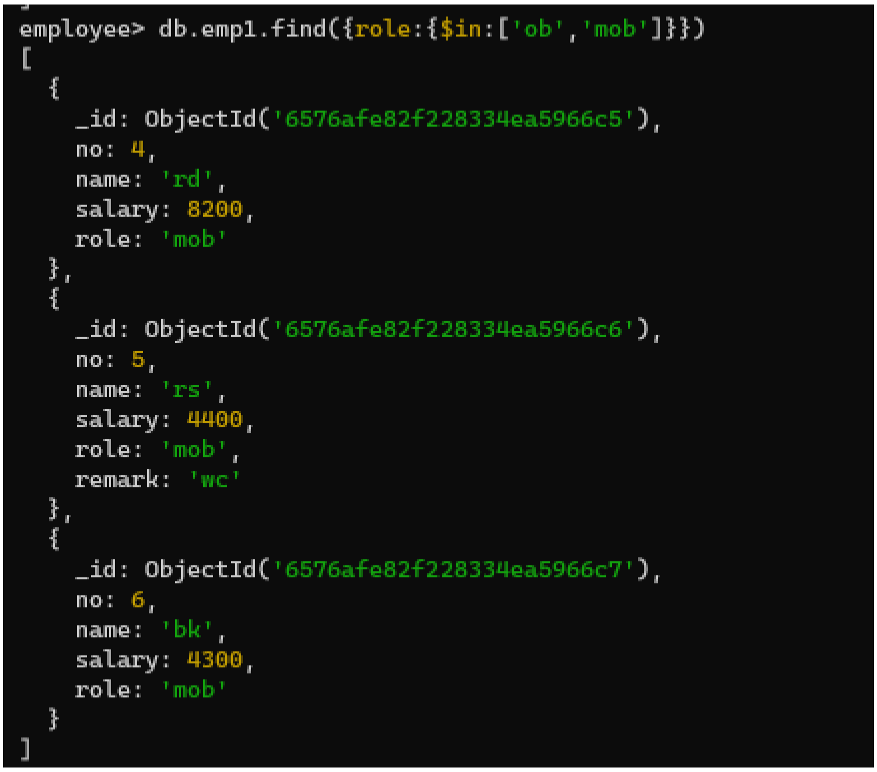
```
employee> db.emp1.find({name:"^s"})

employee> db.emp1.find({name:{$regex:"^s"}})
[
  {
    _id: ObjectId('6576afe82f228334ea5966c2'),
    no: 1,
    name: 'st',
    salary: 32000,
    role: '0b'
  }
]
```

To Find Document From the empl collection where name begins, ends and contains S



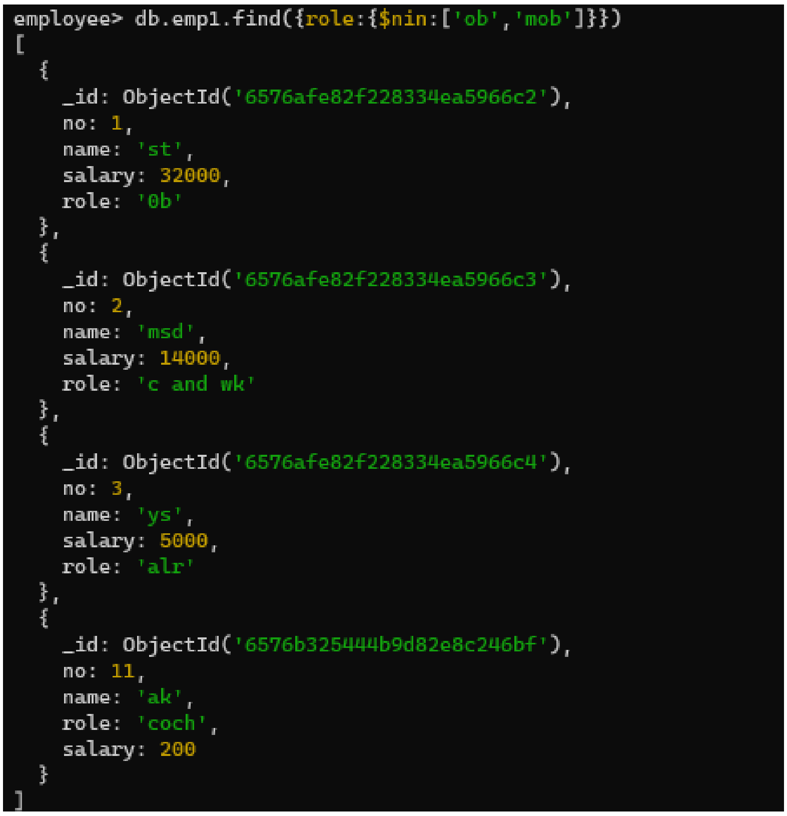
```
employee> db.emp1.find({name:{$regex:"s",$options:"i"}})
[
  {
    _id: ObjectId('6576afe82f228334ea5966c2'),
    no: 1,
    name: 'st',
    salary: 32000,
    role: '0b'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c3'),
    no: 2,
    name: 'msd',
    salary: 14000,
    role: 'c and wk'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c4'),
    no: 3,
    name: 'ys',
    salary: 5000,
    role: 'alr'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c6'),
    no: 5,
    name: 'rs',
    salary: 4400,
    role: 'mob',
    remark: 'wc'
  }
]
```

Display Documents where in empl collection Field have OB,MOB



```
employee> db.emp1.find({role:{$in:['ob','mob']}})
[
  {
    _id: ObjectId('6576afe82f228334ea5966c5'),
    no: 4,
    name: 'rd',
    salary: 8200,
    role: 'mob'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c6'),
    no: 5,
    name: 'rs',
    salary: 4400,
    role: 'mob',
    remark: 'wc'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c7'),
    no: 6,
    name: 'bk',
    salary: 4300,
    role: 'mob'
  }
]
```

Display Documents where in empl collection Field not have OB,MOB



```
employee> db.emp1.find({role:{$nin:['ob','mob']}})
[
  {
    _id: ObjectId('6576afe82f228334ea5966c2'),
    no: 1,
    name: 'st',
    salary: 32000,
    role: '0b'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c3'),
    no: 2,
    name: 'msd',
    salary: 14000,
    role: 'c and wk'
  },
  {
    _id: ObjectId('6576afe82f228334ea5966c4'),
    no: 3,
    name: 'ys',
    salary: 5000,
    role: 'alr'
  },
  {
    _id: ObjectId('6576b325444b9d82e8c246bf'),
    no: 11,
    name: 'ak',
    role: 'coch',
    salary: 200
  }
]
```