

LAB 3

Jiahao Sun

Upload two json file into EC2

```
/cygdrive/d/usc/dsci551
13451@DESKTOP-2DRN4D3 ~
$ cd 'd:/usc/dsci551'

13451@DESKTOP-2DRN4D3 /cygdrive/d/usc/dsci551
$ scp -i dsci551-jiahaosun.pem chats.json ec2-user@ec2-3-141-22-75.us-east-2.com
pute.amazonaws.com:/home/ec2-user/lab
chats.json                                100% 30KB 56.8KB/s 00:00

13451@DESKTOP-2DRN4D3 /cygdrive/d/usc/dsci551
$ scp -i dsci551-jiahaosun.pem roster.json ec2-user@ec2-3-141-22-75.us-east-2.co
mpute.amazonaws.com:/home/ec2-user/lab
roster.json                              100% 4791 16.1KB/s 00:00

13451@DESKTOP-2DRN4D3 /cygdrive/d/usc/dsci551
$ |
```

Load two json file into MongoDB

```
[ec2-user@ip-172-31-20-205 ~]$ cd lab
[ec2-user@ip-172-31-20-205 lab]$ mongoimport --db dsci551 --collection chats --file chats.json
2021-04-02T15:10:57.488+0000 connected to: mongodb://localhost/
2021-04-02T15:10:57.508+0000 301 document(s) imported successfully. 0 document(s) failed to import.
[ec2-user@ip-172-31-20-205 lab]$ mongoimport --db dsci551 --collection roster --file roster.json
2021-04-02T15:14:09.944+0000 connected to: mongodb://localhost/
2021-04-02T15:14:09.955+0000 52 document(s) imported successfully. 0 document(s) failed to import.
[ec2-user@ip-172-31-20-205 lab]$
```

1. Find out the number of people who made at least one chat

```
ec2-user@ip-172-31-20-205~
---
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).

The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---
> use dsci551
switched to db dsci551
> db.chats.find()
{"_id" : ObjectId("60673401f0f9a027cc0aa880"), "Time" : "00:01:44", "Person" : "Yiyang Chen", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa881"), "Time" : "00:01:46", "Person" : "Junbo Sheng", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa882"), "Time" : "00:01:46", "Person" : "Jiayi Lin", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa883"), "Time" : "00:01:47", "Person" : "Baitong Liu", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa884"), "Time" : "00:01:47", "Person" : "Zian Fan", "Message" : "11111" }
{"_id" : ObjectId("60673401f0f9a027cc0aa885"), "Time" : "00:01:48", "Person" : "Leon Luc", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa886"), "Time" : "00:01:48", "Person" : "Zhiqian Wang", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa887"), "Time" : "00:01:49", "Person" : "Jiahui Lu", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa888"), "Time" : "00:01:49", "Person" : "Shiming Chen", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa889"), "Time" : "00:01:49", "Person" : "Zihao Niu", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa88a"), "Time" : "00:01:50", "Person" : "Yue Meng", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa88b"), "Time" : "00:01:50", "Person" : "Qi Xie", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa88c"), "Time" : "00:01:51", "Person" : "Spoorthy Bhat", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa88d"), "Time" : "00:01:52", "Person" : "Dayuan Jian", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa88e"), "Time" : "00:01:52", "Person" : "Minyi Ding", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa88f"), "Time" : "00:01:54", "Person" : "Luke Nelson", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa890"), "Time" : "00:01:54", "Person" : "Dongzhe Jiang", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa891"), "Time" : "00:01:56", "Person" : "Shaozhen Lu", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa892"), "Time" : "00:01:56", "Person" : "Ruochen Cao", "Message" : "1" }
{"_id" : ObjectId("60673401f0f9a027cc0aa893"), "Time" : "00:01:57", "Person" : "Gwen Fang", "Message" : "1" }
Type "it" for more
> db.chats.distinct("Person").length
52
> |
```

2. Find out the names of students (from roster) whose name contains "li" (case insensitive). Output the names in the ascending order.

```
db.roster.find({"Name":/li/i}, {"Name":1, "_id":0}).sort({Name:1})
```

```

ec2-user@ip-172-31-20-205:~$
{
  "Name": "Yadi Zhao"
}
{
  "Name": "Zhixin Xie"
}
> db.chats.aggregate({"$group":{"_id":"$Person",total:{$sum:1}}},{ $sort:{total:-1}},{$limit:5},{ $project:{_id:1}})
{
  "_id": "Vishal Ajaybhai Kapadia"
}
{
  "_id": "Wensheng Wu"
}
{
  "_id": "Tirth Patel"
}
{
  "_id": "Vinjit Regulagedda"
}
{
  "_id": "Madeleine Thompson"
}
> db.roster.find({"Name":"/i/i}).sort({Name:1},{ $project:{_id:0, Name:1}})
{
  "_id": ObjectId("606734c1ac1d10df561daca4"), "Name": "Baitong Liu", "Participating from": "China"
}
{
  "_id": ObjectId("606734c1ac1d10df561daca3"), "Name": "Jiayi Lin", "Participating from": "China"
}
{
  "_id": ObjectId("606734c1ac1d10df561dac9f"), "Name": "Mengxun Li", "Participating from": "China"
}
{
  "_id": ObjectId("606734c1ac1d10df561daca1"), "Name": "Ruoxuan Li", "Participating from": "China"
}
{
  "_id": ObjectId("606734c1ac1d10df561daca0"), "Name": "Wenhao Li", "Participating from": "United States of America"
}
{
  "_id": ObjectId("606734c1ac1d10df561daca2"), "Name": "Wenxuan Li", "Participating from": "China"
}
> db.roster.find({"Name":"/i/i},{ $sort:{Name:1}},{$project:{_id:0, Name:1}})
Error: error: {
  "ok": 0,
  "errmsg": "FieldPath field names may not start with '$'.",
  "code": 16410,
  "codeName": "Location16410"
}
> db.roster.find({"Name":"/i/i},{ $project:{_id:0, Name:1}}).sort({Name:1})
Error: error: {
  "ok": 0,
  "errmsg": "FieldPath field names may not start with '$'.",
  "code": 16410,
  "codeName": "Location16410"
}
}
> db.roster.find({"Name":"/i/i, {"Name":1,"_id":0}}).sort({Name:1})
{
  "Name": "Baitong Liu"
}
{
  "Name": "Jiayi Lin"
}
{
  "Name": "Mengxun Li"
}
{
  "Name": "Ruoxuan Li"
}
{
  "Name": "Wenhao Li"
}
{
  "Name": "Wenxuan Li"
}

```

3. Find out the names of top-5 students ranked by the number of chats they have made.

```

db.chats.aggregate({"$group":{"_id":"$Person",total:{$sum:1}}},{ $sort:{total:-1}},{$limit:5},{ $project:{_id:1}})

```

```

ec2-user@ip-172-31-20-205:~$
Message": "1" }, { "_id": ObjectId("60673401f0f9a027cc0aa89b"), "Time": "00:07:21", "Person": "Harrison Knapp", "Message": "1" }, { "_id": ObjectId("60673401f0f9a027cc0aa901"), "Time": "02:18:31", "Person": "Harrison Knapp", "Message": "1" }, { "_id": ObjectId("60673401f0f9a027cc0aa966"), "Time": "02:51:33", "Person": "Harrison Knapp", "Message": "1" } ] }
{
  "_id": ObjectId("606734c1ac1d10df561dac9e"), "Name": "Maxwell Lee", "Participating from": "United States of America", "talk": [ { "_id": ObjectId("60673401f0f9a027cc0aa8fb"), "Time": "02:18:28", "Person": "Maxwell Lee", "Message": "1" }, { "_id": ObjectId("60673401f0f9a027cc0aa914"), "Time": "02:27:57", "Person": "Maxwell Lee", "Message": "Hfs and map reduce" }, { "_id": ObjectId("60673401f0f9a027cc0aa94e"), "Time": "02:44:08", "Person": "Maxwell Lee", "Message": "name node" }, { "_id": ObjectId("60673401f0f9a027cc0aa952"), "Time": "02:46:00", "Person": "Maxwell Lee", "Message": "2" }, { "_id": ObjectId("60673401f0f9a027cc0aa95d"), "Time": "02:51:28", "Person": "Maxwell Lee", "Message": "1" }, { "_id": ObjectId("60673401f0f9a027cc0aa978"), "Time": "02:57:02", "Person": "Maxwell Lee", "Message": "I think there's a way to mute participants as the host" }, { "_id": ObjectId("60673401f0f9a027cc0aa98d"), "Time": "03:26:00", "Person": "Maxwell Lee", "Message": "1" } ] }
{
  "_id": ObjectId("606734c1ac1d10df561dac9f"), "Name": "Mengxun Li", "Participating from": "China", "talk": [ ] }
Type "it" for more
> db.roster.aggregate({"$lookup":{"from":"chats",localField:"Name",foreignField:"Person",as:"talk"}},{ $match:{talk:{$eq:[]}}},{ $project:{Name:1}},{$sort:{Name:1}})
{
  "_id": ObjectId("606734c1ac1d10df561dacba"), "Name": "Anthony Zhang"
}
{
  "_id": ObjectId("606734c1ac1d10df561dac96"), "Name": "Huanpu Hu"
}
{
  "_id": ObjectId("606734c1ac1d10df561dac9c"), "Name": "Kelly Kim"
}
{
  "_id": ObjectId("606734c1ac1d10df561dac9f"), "Name": "Mengxun Li"
}
{
  "_id": ObjectId("606734c1ac1d10df561dacbd"), "Name": "Yadi Zhao"
}
{
  "_id": ObjectId("606734c1ac1d10df561dacb9"), "Name": "Zhixin Xie"
}
> db.roster.aggregate({"$lookup":{"from":"chats",localField:"Name",foreignField:"Person",as:"talk"}},{ $match:{talk:{$eq:[]}}},{ $project:{Name:1,_id:0}},{$sort:{Name:1}})
{
  "Name": "Anthony Zhang"
}
{
  "Name": "Huanpu Hu"
}
{
  "Name": "Kelly Kim"
}
{
  "Name": "Mengxun Li"
}
{
  "Name": "Yadi Zhao"
}
{
  "Name": "Zhixin Xie"
}
> db.chats.aggregate({"$group":{"_id":"$Person",total:{$sum:1}}},{ $sort:{total:-1}},{$limit:5},{ $project:{_id:1}})
{
  "_id": "Vishal Ajaybhai Kapadia"
}
{
  "_id": "Wensheng Wu"
}
{
  "_id": "Tirth Patel"
}
{
  "_id": "Vinjit Regulagedda"
}
{
  "_id": "Madeleine Thompson"
}

```

4. Find out the names of students who did not make any chats. Return names only, sorted (ascending order)

```

db.roster.aggregate({"$lookup":{"from":"chats",localField:"Name",foreignField:"Person",as:"talk"}},{ $match:{talk:{$eq:[]}}},{ $project:{Name:1,_id:0}},{$sort:{Name:1}})

```

```

ec2-user@ip-172-31-20-205:~$ mongo
> use dsci551
switched to db dsci551
> db.roster.find()
{ "_id" : ObjectId("606734c1ac1d10df561dac8c"), "Name" : "Kelly Kim", "Participating from" : "United States of America", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac9d"), "Name" : "Harrison Knapp", "Participating from" : "United States of America", "talk" : [ { "_id" : ObjectId("60673401f0f9a027cc0aa89a"), "Time" : "00:02:12", "Person" : "Harrison Knapp", "Message" : "1" }, { "_id" : ObjectId("60673401f0f9a027cc0aa89b"), "Time" : "00:07:21", "Person" : "Harrison Knapp", "Message" : "1" }, { "_id" : ObjectId("60673401f0f9a027cc0aa901"), "Time" : "02:18:31", "Person" : "Harrison Knapp", "Message" : "1" }, { "_id" : ObjectId("60673401f0f9a027cc0aa966"), "Time" : "02:51:33", "Person" : "Harrison Knapp", "Message" : "1" } ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac9e"), "Name" : "Maxwell Lee", "Participating from" : "United States of America", "talk" : [ { "_id" : ObjectId("60673401f0f9a027cc0aa8fb"), "Time" : "02:18:28", "Person" : "Maxwell Lee", "Message" : "1" }, { "_id" : ObjectId("60673401f0f9a027cc0aa914"), "Time" : "02:27:57", "Person" : "Maxwell Lee", "Message" : "Hfs and map reduce" }, { "_id" : ObjectId("60673401f0f9a027cc0aa94e"), "Time" : "02:44:08", "Person" : "Maxwell Lee", "Message" : "name node" }, { "_id" : ObjectId("60673401f0f9a027cc0aa952"), "Time" : "02:46:00", "Person" : "Maxwell Lee", "Message" : "2" }, { "_id" : ObjectId("60673401f0f9a027cc0aa95d"), "Time" : "02:51:28", "Person" : "Maxwell Lee", "Message" : "1" }, { "_id" : ObjectId("60673401f0f9a027cc0aa978"), "Time" : "02:57:02", "Person" : "Maxwell Lee", "Message" : "I think there's a way to mute participants as the host" }, { "_id" : ObjectId("60673401f0f9a027cc0aa98d"), "Time" : "03:26:00", "Person" : "Maxwell Lee", "Message" : "1" } ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac9f"), "Name" : "Mengxun Li", "Participating from" : "China", "talk" : [ ] }
Type 'it' for more
> db.roster.aggregate({$lookup:{from:'chats',localField:'Name',foreignField:'Person',as:'talk'}},{match:{talk:{seq:[]}}},{project:{Name:1}},{sort:{Name:1}})
{ "_id" : ObjectId("606734c1ac1d10df561dacba"), "Name" : "Anthony Zhang" }
{ "_id" : ObjectId("606734c1ac1d10df561dac96"), "Name" : "Huanpu Hu" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9c"), "Name" : "Kelly Kim" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9f"), "Name" : "Mengxun Li" }
{ "_id" : ObjectId("606734c1ac1d10df561dacbd"), "Name" : "Yadi Zhao" }
{ "_id" : ObjectId("606734c1ac1d10df561dacb9"), "Name" : "Zhixin Xie" }
> db.roster.aggregate({$lookup:{from:'chats',localField:'Name',foreignField:'Person',as:'talk'}},{match:{talk:{seq:[]}}},{project:{Name:1,_id:0}},{sort:{Name:1}})
{ "Name" : "Anthony Zhang" }
{ "Name" : "Huanpu Hu" }
{ "Name" : "Kelly Kim" }
{ "Name" : "Mengxun Li" }
{ "Name" : "Yadi Zhao" }
{ "Name" : "Zhixin Xie" }

```

5. Add a new student "John Smith", participating from "United States of America" to the roster, with age = 25

`db.roster.insert({"_id":ObjectId(),"Name":"John Smith","Participating from":"United States of America","age":25})`

```

ec2-user@ip-172-31-20-205:~$ mongo
> use dsci551
switched to db dsci551
> db.roster.find()
{ "_id" : ObjectId("606734c1ac1d10df561dac8c"), "Name" : "Spoorthy Bhat", "Participating from" : "United States of America", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac8d"), "Name" : "Ruochen Cao", "Participating from" : "China", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac8e"), "Name" : "Max Chen", "Participating from" : "United States of America", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac8f"), "Name" : "Shiming Chen", "Participating from" : "China", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac90"), "Name" : "Yiyang Chen", "Participating from" : "United States of America", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac91"), "Name" : "Manikanta Chunduru Balaji", "Participating from" : "India", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac92"), "Name" : "Winyi Ding", "Participating from" : "China", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac93"), "Name" : "Zian Fan", "Participating from" : "United States of America", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac94"), "Name" : "Gwen Fang", "Participating from" : "United States of America", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac95"), "Name" : "Qianru Hou", "Participating from" : "China", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac96"), "Name" : "Huanpu Hu", "Participating from" : "China", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac97"), "Name" : "Shayne Hu", "Participating from" : "China", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac98"), "Name" : "Dayuan Jian", "Participating from" : "United States of America", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac99"), "Name" : "Dongzhe Jiang", "Participating from" : "China", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac9a"), "Name" : "Yanru Jiang", "Participating from" : "United States of America", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac9b"), "Name" : "Vishal Ajaybhai Kapadia", "Participating from" : "United States of America", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac9c"), "Name" : "Kelly Kim", "Participating from" : "United States of America", "talk" : [ ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac9d"), "Name" : "Harrison Knapp", "Participating from" : "United States of America", "talk" : [ { "_id" : ObjectId("60673401f0f9a027cc0aa89a"), "Time" : "00:02:12", "Person" : "Harrison Knapp", "Message" : "1" }, { "_id" : ObjectId("60673401f0f9a027cc0aa89b"), "Time" : "00:07:21", "Person" : "Harrison Knapp", "Message" : "1" }, { "_id" : ObjectId("60673401f0f9a027cc0aa901"), "Time" : "02:18:31", "Person" : "Harrison Knapp", "Message" : "1" }, { "_id" : ObjectId("60673401f0f9a027cc0aa966"), "Time" : "02:51:33", "Person" : "Harrison Knapp", "Message" : "1" } ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac9e"), "Name" : "Maxwell Lee", "Participating from" : "United States of America", "talk" : [ { "_id" : ObjectId("60673401f0f9a027cc0aa8fb"), "Time" : "02:18:28", "Person" : "Maxwell Lee", "Message" : "1" }, { "_id" : ObjectId("60673401f0f9a027cc0aa914"), "Time" : "02:27:57", "Person" : "Maxwell Lee", "Message" : "Hfs and map reduce" }, { "_id" : ObjectId("60673401f0f9a027cc0aa94e"), "Time" : "02:44:08", "Person" : "Maxwell Lee", "Message" : "name node" }, { "_id" : ObjectId("60673401f0f9a027cc0aa952"), "Time" : "02:46:00", "Person" : "Maxwell Lee", "Message" : "2" }, { "_id" : ObjectId("60673401f0f9a027cc0aa95d"), "Time" : "02:51:28", "Person" : "Maxwell Lee", "Message" : "1" }, { "_id" : ObjectId("60673401f0f9a027cc0aa978"), "Time" : "02:57:02", "Person" : "Maxwell Lee", "Message" : "I think there's a way to mute participants as the host" }, { "_id" : ObjectId("60673401f0f9a027cc0aa98d"), "Time" : "03:26:00", "Person" : "Maxwell Lee", "Message" : "1" } ] }
{ "_id" : ObjectId("606734c1ac1d10df561dac9f"), "Name" : "Mengxun Li", "Participating from" : "China", "talk" : [ ] }
Type 'it' for more
> db.roster.insert({"_id":ObjectId(),"Name":"John Smith","Participating from":"United States of America","age":25})
WriteResult({ "nInserted" : 1 })

```

6. Add a new student "Mary Smith" with "Age" = 30 (no need to add participation info) and query it by name.

```

ec2-user@ip-172-31-20-205:~
{ "_id" : ObjectId("606734c1ac1d10df561dac8d"), "Name" : "Ruochen Cao", "Participating from" : "China" }
{ "_id" : ObjectId("606734c1ac1d10df561dac8e"), "Name" : "Max Chen", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac8f"), "Name" : "Shiming Chen", "Participating from" : "China" }
{ "_id" : ObjectId("606734c1ac1d10df561dac90"), "Name" : "Yiyang Chen", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac91"), "Name" : "Manikanta Chunduru Balaji", "Participating from" : "India" }
{ "_id" : ObjectId("606734c1ac1d10df561dac92"), "Name" : "Minyi Ding", "Participating from" : "China" }
{ "_id" : ObjectId("606734c1ac1d10df561dac93"), "Name" : "Zian Fan", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac94"), "Name" : "Gwen Fang", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac95"), "Name" : "Qianru Hou", "Participating from" : "China" }
{ "_id" : ObjectId("606734c1ac1d10df561dac96"), "Name" : "Huanpu Hu", "Participating from" : "China" }
{ "_id" : ObjectId("606734c1ac1d10df561dac97"), "Name" : "Shayne Hu", "Participating from" : "China" }
{ "_id" : ObjectId("606734c1ac1d10df561dac98"), "Name" : "Dayuan Jian", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac99"), "Name" : "Dongzhe Jiang", "Participating from" : "China" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9a"), "Name" : "Yanru Jiang", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9b"), "Name" : "Vishal Ajaybhai Kapadia", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9c"), "Name" : "Kelly Kim", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9d"), "Name" : "Harrison Knapp", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9e"), "Name" : "Maxwell Lee", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9f"), "Name" : "Mengxun Li", "Participating from" : "China" }
Type "it" for more
> db.roster.insert({"_id":ObjectId(),"Name":"John Smith","Participating from":"United States of America","age":25})
WriteResult({"nInserted" : 1 })
> db.roster.insert({"_id":ObjectId(),"Name":"Mary Smith","age":30})
WriteResult({"nInserted" : 1 })
> db.roster.find({"Name":"Mary Smith"})
{ "_id" : ObjectId("6067495bcccc4206ae4107b2"), "Name" : "Mary Smith", "age" : 30 }

```

7. Change the name of student John Smith to Jonny Smith and age to 36 and query it by age equal to 36

```

ec2-user@ip-172-31-20-205:~
{ "_id" : ObjectId("606734c1ac1d10df561dac91"), "Name" : "Manikanta Chunduru Balaji", "Participating from" : "India" }
{ "_id" : ObjectId("606734c1ac1d10df561dac92"), "Name" : "Minyi Ding", "Participating from" : "China" }
{ "_id" : ObjectId("606734c1ac1d10df561dac93"), "Name" : "Zian Fan", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac94"), "Name" : "Gwen Fang", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac95"), "Name" : "Qianru Hou", "Participating from" : "China" }
{ "_id" : ObjectId("606734c1ac1d10df561dac96"), "Name" : "Huanpu Hu", "Participating from" : "China" }
{ "_id" : ObjectId("606734c1ac1d10df561dac97"), "Name" : "Shayne Hu", "Participating from" : "China" }
{ "_id" : ObjectId("606734c1ac1d10df561dac98"), "Name" : "Dayuan Jian", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac99"), "Name" : "Dongzhe Jiang", "Participating from" : "China" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9a"), "Name" : "Yanru Jiang", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9b"), "Name" : "Vishal Ajaybhai Kapadia", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9c"), "Name" : "Kelly Kim", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9d"), "Name" : "Harrison Knapp", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9e"), "Name" : "Maxwell Lee", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9f"), "Name" : "Mengxun Li", "Participating from" : "China" }
Type "it" for more
> db.roster.insert({"_id":ObjectId(),"Name":"John Smith","Participating from":"United States of America","age":25})
WriteResult({"nInserted" : 1 })
> db.roster.insert({"_id":ObjectId(),"Name":"Mary Smith","age":30})
WriteResult({"nInserted" : 1 })
>
> db.roster.find({"Name":"Mary Smith"})
{ "_id" : ObjectId("6067495bcccc4206ae4107b2"), "Name" : "Mary Smith", "age" : 30 }
> db.roster.update({"Name":"John Smith"},{$set:{"Name":"Jonny Smith","age":36},{upsert:true})
WriteResult({"nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.roster.find({"age":36})
{ "_id" : ObjectId("6067489bcccc4206ae4107b1"), "Name" : "Jonny Smith", "Participating from" : "United States of America", "age" : 36 }
>

```

8. Add a new attribute "Program" and set its value to "ADS" for all students whose names contain Smith (case sensitive). And then query these students

```

ec2-user@ip-172-31-20-205:~$ mongo
{ "_id" : ObjectId("606734c1ac1d10df561dac9a"), "Name" : "Yanru Jiang", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9b"), "Name" : "Vishal Ajaybhai Kapadia", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9c"), "Name" : "Kelly Kim", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9d"), "Name" : "Harrison Knapp", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9e"), "Name" : "Maxwell Lee", "Participating from" : "United States of America" }
{ "_id" : ObjectId("606734c1ac1d10df561dac9f"), "Name" : "Mengxun Li", "Participating from" : "China" }
Type "it" for more
> db.roster.insert({"_id":ObjectId(),"Name":"John Smith","Participating from":"United States of America","age":25})
WriteResult({"nInserted" : 1 })
> db.roster.insert({"_id":ObjectId(),"Name":"Mary Smith","age":30})
WriteResult({"nInserted" : 1 })
>
> db.roster.find({"Name":"Mary Smith"})
{ "_id" : ObjectId("6067495bcccc4206ae4107b2"), "Name" : "Mary Smith", "age" : 30 }
> db.roster.update({"Name":"John Smith"},{$set:{"Name":"Jonny Smith","age":36},{upsert:true})
WriteResult({"nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.roster.find({"age":36})
{ "_id" : ObjectId("6067489bcccc4206ae4107b1"), "Name" : "Jonny Smith", "Participating from" : "United States of America", "age" : 36 }
> db.roster.find({"Name":"/Smith/})
{ "_id" : ObjectId("6067489bcccc4206ae4107b1"), "Name" : "Jonny Smith", "Participating from" : "United States of America", "age" : 36 }
{ "_id" : ObjectId("6067495bcccc4206ae4107b2"), "Name" : "Mary Smith", "age" : 30 }
> db.roster.find({"Name":"/Smith})
uncaught exception: SyntaxError: unterminated regular expression literal :
@(:shell):1:23
> db.roster.update({"Name":"/Smith/},{set:{"Program":"ADS"}},{multi:true})
WriteResult({"nMatched" : 2, "nUpserted" : 0, "nModified" : 2 })
> db.roster.find({"Program":"ADS"})
{ "_id" : ObjectId("6067489bcccc4206ae4107b1"), "Name" : "Jonny Smith", "Participating from" : "United States of America", "age" : 36, "Program" : "ADS" }
{ "_id" : ObjectId("6067495bcccc4206ae4107b2"), "Name" : "Mary Smith", "age" : 30, "Program" : "ADS" }

```