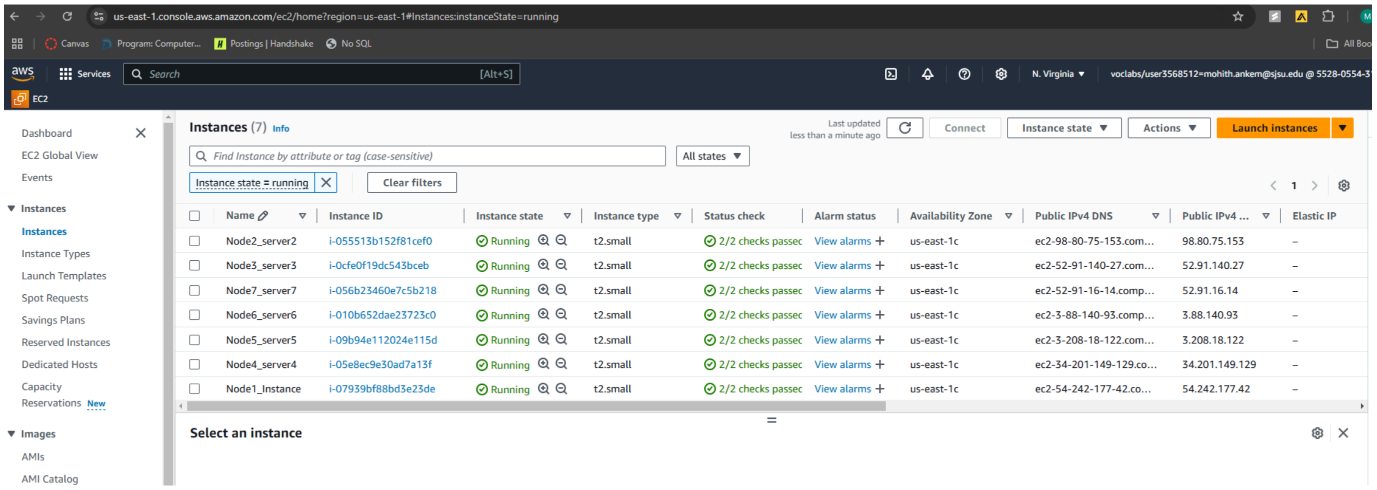
1. <u>Set up nodes in AWS. Determine the number of nodes based on your deployment plan.</u>

Ans:There are a total of 7 instances; 3 for config servers, 1 for mongos router and the other 3 for sharding.

Config Servers : Node1, Node2 and Node3

Mongos Server: Node4

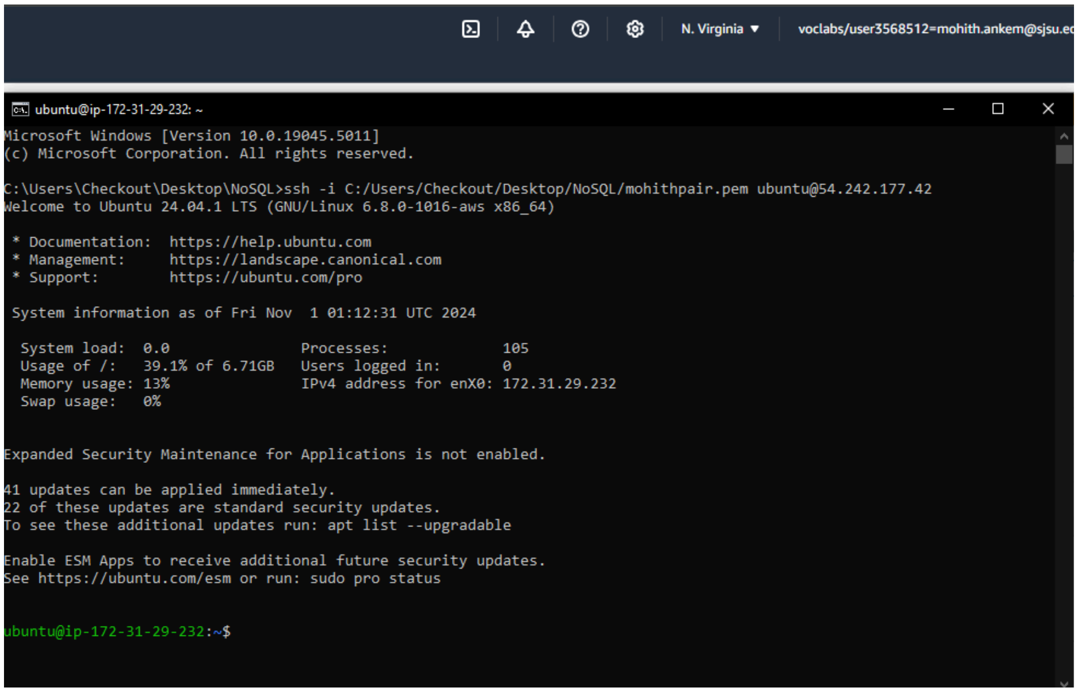
Shard Servers: Node5, Node6 and Node7



Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
Node2_server2	i-055513b152f81cef0	Running	t2.small	2/2 checks passec	View alarms +	us-east-1c	ec2-98-80-75-153.com...	98.80.75.153	–
Node3_server3	i-0cfe0f19dc543bceb	Running	t2.small	2/2 checks passec	View alarms +	us-east-1c	ec2-52-91-140-27.com...	52.91.140.27	–
Node7_server7	i-056b23460e7c5b218	Running	t2.small	2/2 checks passec	View alarms +	us-east-1c	ec2-52-91-16-14.comp...	52.91.16.14	–
Node6_server6	i-010b652dae23723c0	Running	t2.small	2/2 checks passec	View alarms +	us-east-1c	ec2-3-88-140-93.comp...	3.88.140.93	–
Node5_server5	i-09b94e112024e115d	Running	t2.small	2/2 checks passec	View alarms +	us-east-1c	ec2-3-208-18-122.com...	3.208.18.122	–
Node4_server4	i-05e8ec9e30ad7a13f	Running	t2.small	2/2 checks passec	View alarms +	us-east-1c	ec2-34-201-149-129.co...	34.201.149.129	–
Node1_Instance	i-07939bf88bd3e23de	Running	t2.small	2/2 checks passec	View alarms +	us-east-1c	ec2-54-242-177-42.co...	54.242.177.42	–

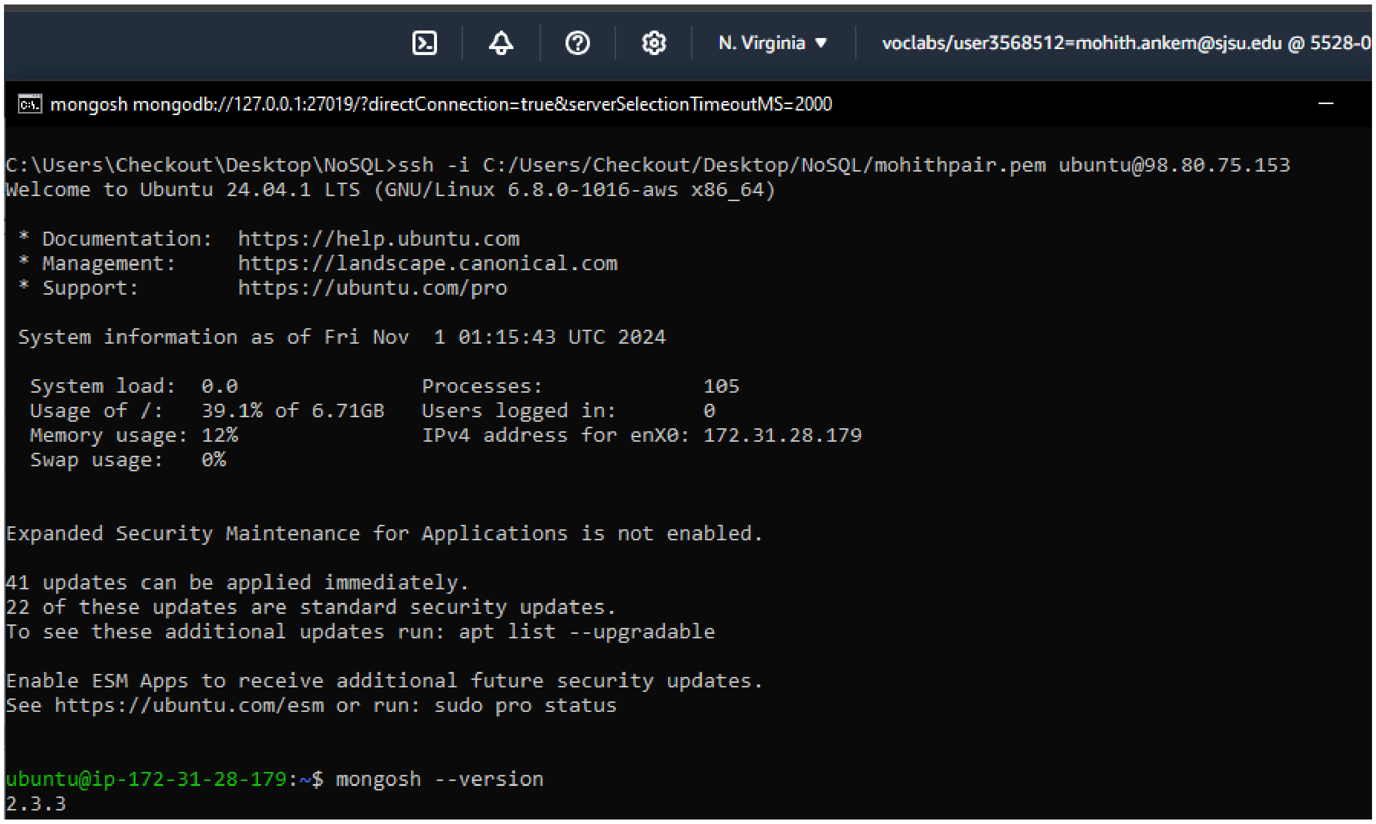
2. <u>Access these instances (nodes) through SSH</u>

Instance 1:


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.e
ubuntu@ip-172-31-29-232: ~

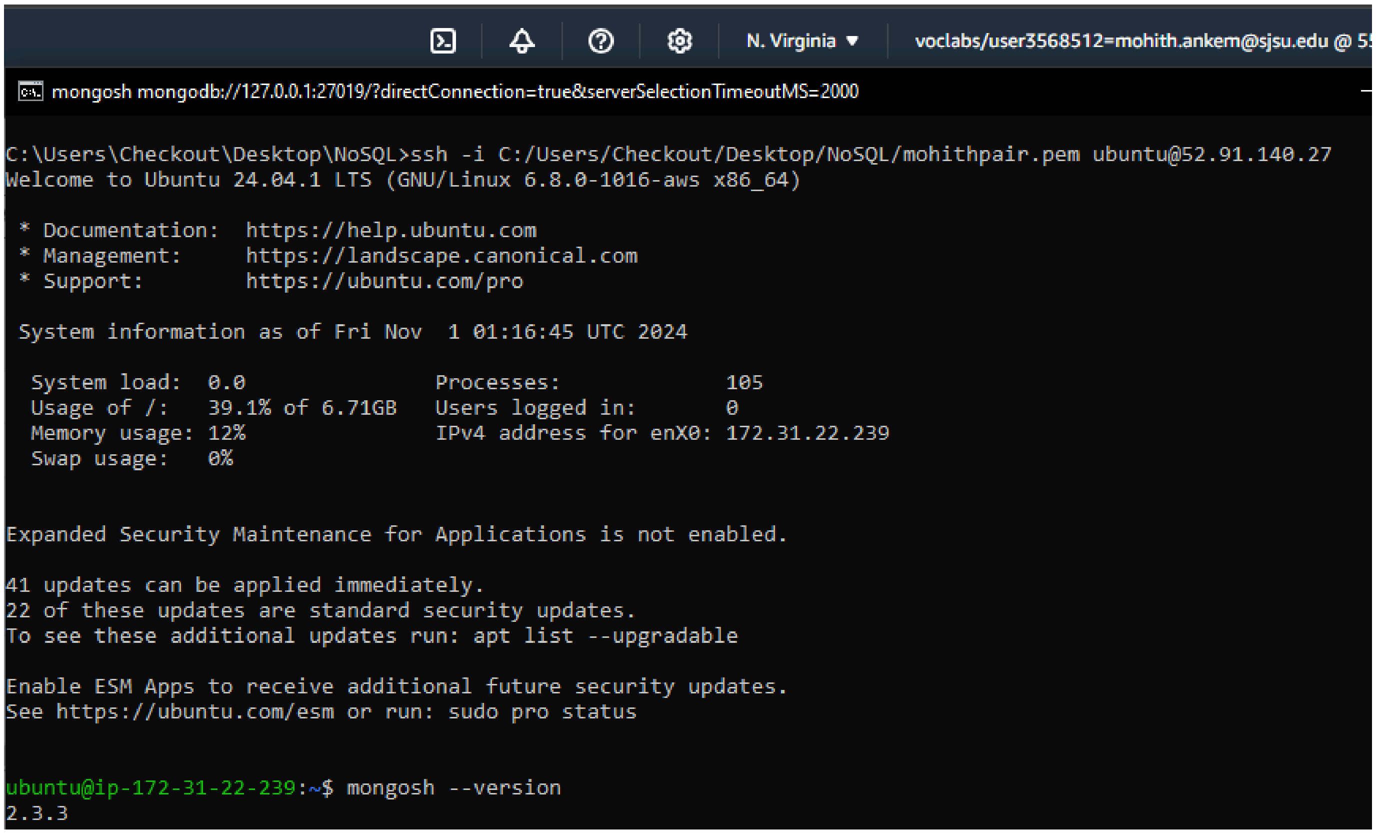
```
Microsoft Windows [Version 10.0.19045.5011]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Checkout\Desktop\NoSQL>ssh -i C:/Users/Checkout/Desktop/NoSQL/mohithpair.pem ubuntu@54.242.177.42
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Fri Nov  1 01:12:31 UTC 2024

  System load:  0.0               Processes:             105
  Usage of /:   39.1% of 6.71GB   Users logged in:       0
  Memory usage: 13%               IPv4 address for enX0: 172.31.29.232
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

41 updates can be applied immediately.
22 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


ubuntu@ip-172-31-29-232:~$
```

Instance 2:


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0
mongosh mongodb://127.0.0.1:27019/?directConnection=true&serverSelectionTimeoutMS=2000

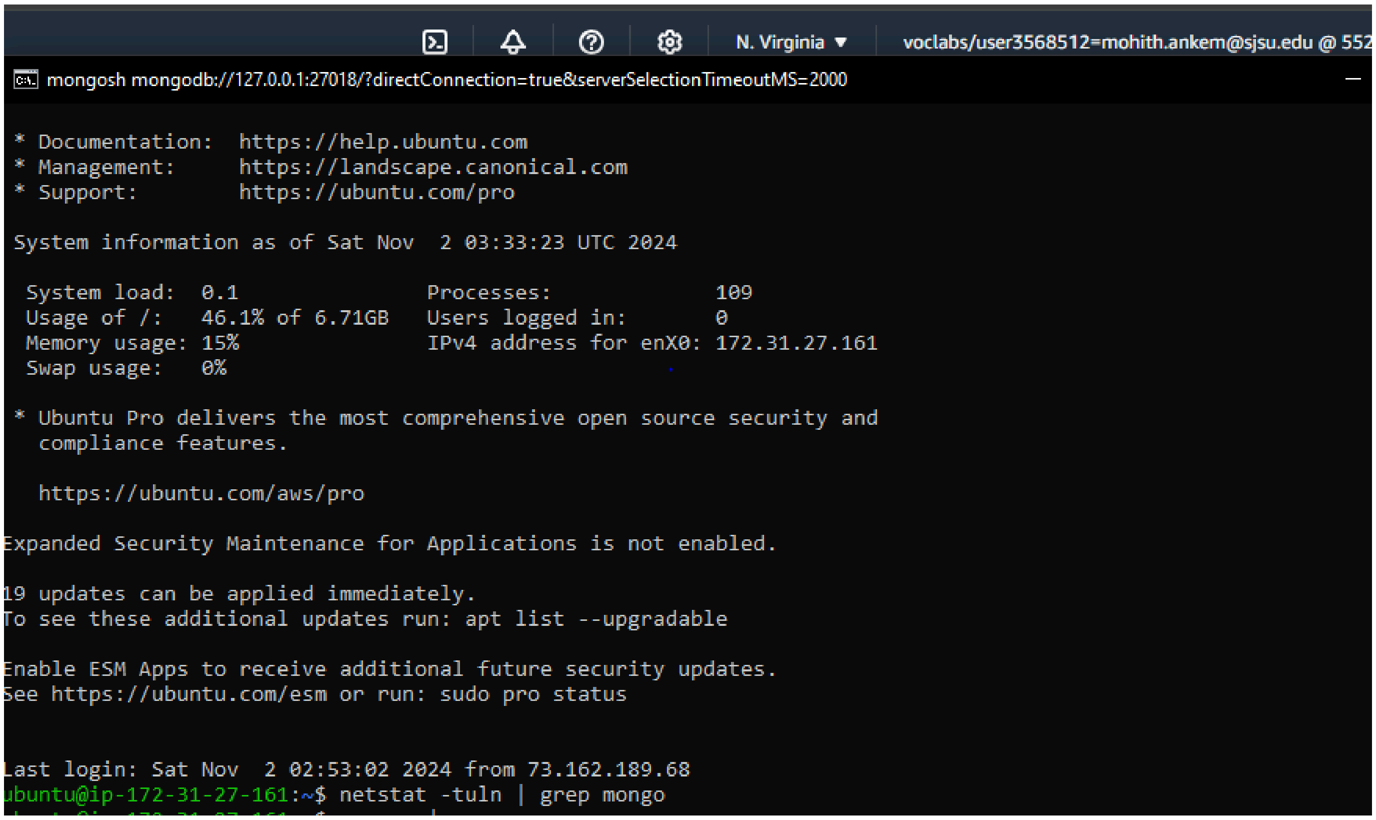
```
C:\Users\Checkout\Desktop\NoSQL>ssh -i C:/Users/Checkout/Desktop/NoSQL/mohithpair.pem ubuntu@98.80.75.153
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Fri Nov  1 01:15:43 UTC 2024

  System load:  0.0               Processes:             105
  Usage of /:   39.1% of 6.71GB   Users logged in:       0
  Memory usage: 12%               IPv4 address for enX0: 172.31.28.179
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

41 updates can be applied immediately.
22 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


ubuntu@ip-172-31-28-179:~$ mongosh --version
2.3.3
```

Instance 3:


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5
mongosh mongodb://127.0.0.1:27019/?directConnection=true&serverSelectionTimeoutMS=2000

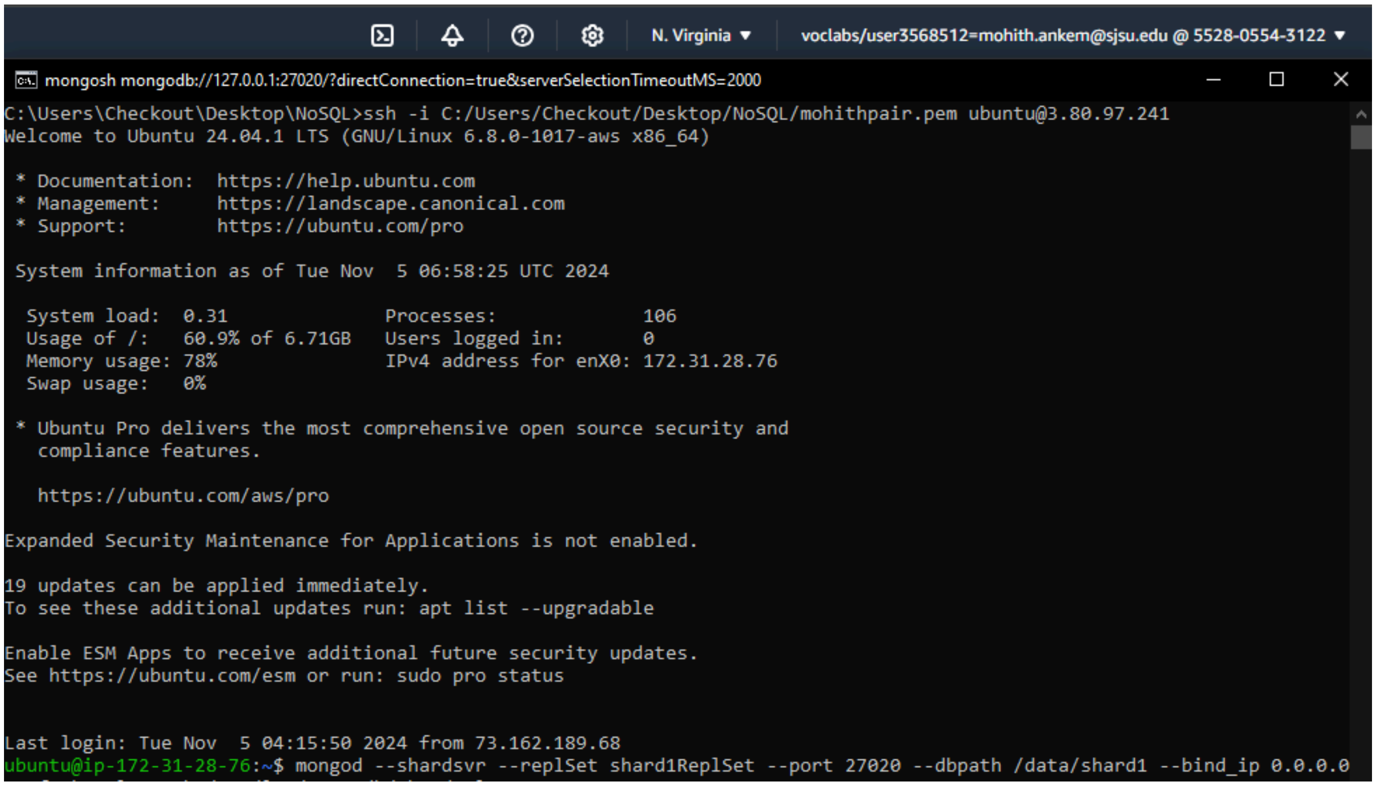
```
C:\Users\Checkout\Desktop\NoSQL>ssh -i C:/Users/Checkout/Desktop/NoSQL/mohithpair.pem ubuntu@52.91.140.27
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Fri Nov  1 01:16:45 UTC 2024

  System load:  0.0               Processes:             105
  Usage of /:   39.1% of 6.71GB   Users logged in:       0
  Memory usage: 12%               IPv4 address for enX0: 172.31.22.239
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

41 updates can be applied immediately.
22 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


ubuntu@ip-172-31-22-239:~$ mongosh --version
2.3.3
```

Instance 4:



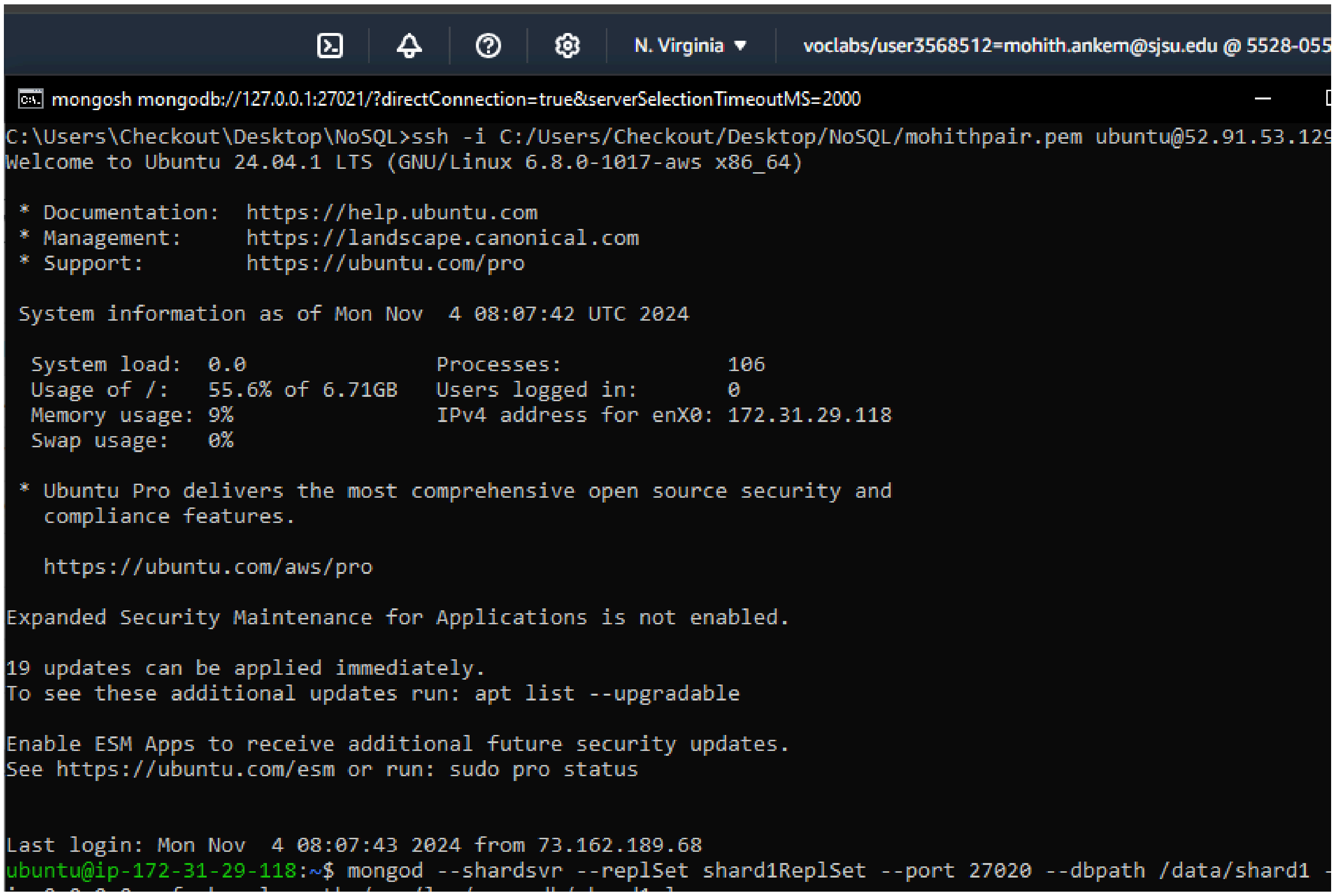
```
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Sat Nov  2 03:33:23 UTC 2024

  System load:  0.1               Processes:             109
  Usage of /:   46.1% of 6.71GB   Users logged in:       0
  Memory usage: 15%               IPv4 address for enX0: 172.31.27.161
  Swap usage:   0%

 * Ubuntu Pro delivers the most comprehensive open source security and
   compliance features.

   https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

19 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


Last login: Sat Nov  2 02:53:02 2024 from 73.162.189.68
ubuntu@ip-172-31-27-161:~$ netstat -tuln | grep mongo
```

Instance 5:


N. Virginia ▼
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122 ▼

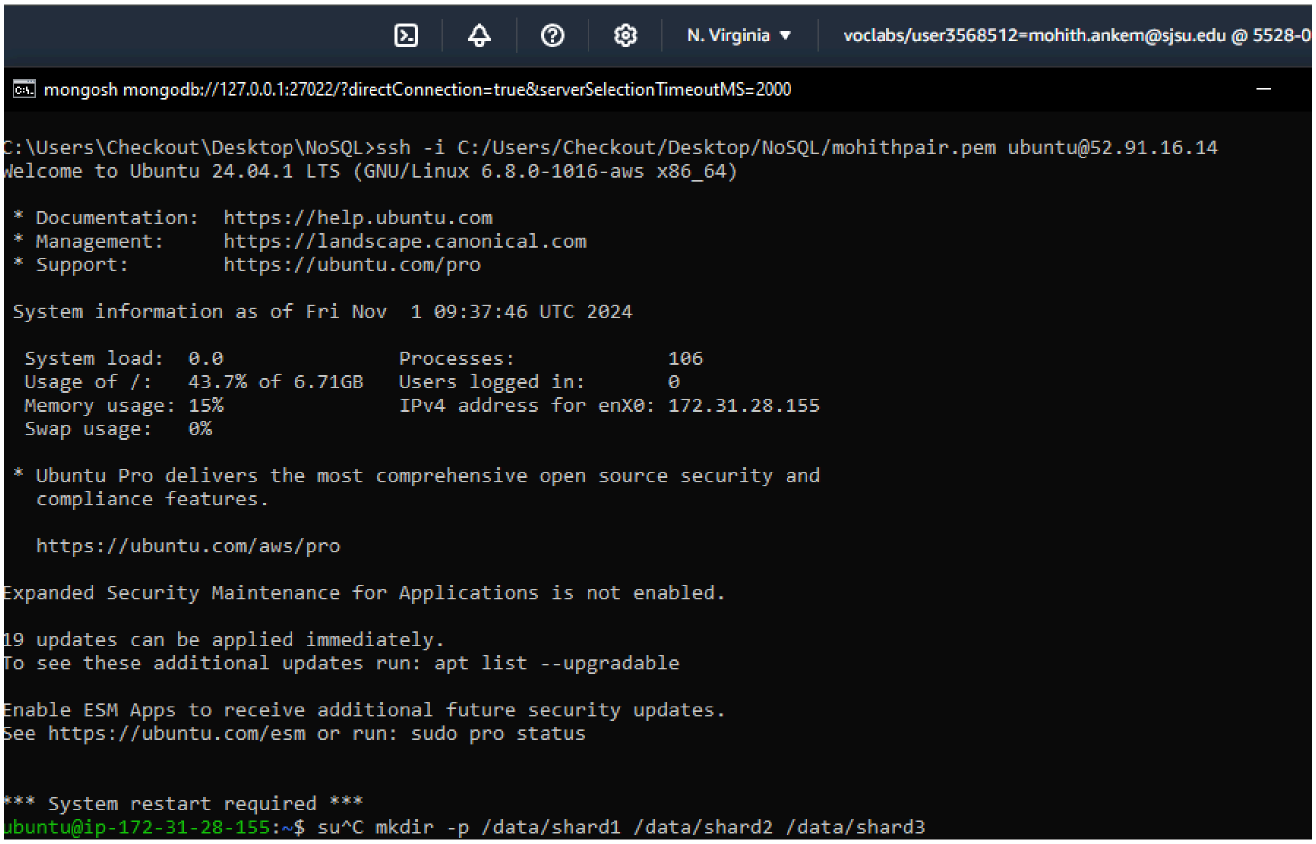
```
mongosh mongodb://127.0.0.1:27020/?directConnection=true&serverSelectionTimeoutMS=2000
C:\Users\Checkout\Desktop\NoSQL>ssh -i C:/Users/Checkout/Desktop/NoSQL/mohithpair.pem ubuntu@3.80.97.241
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1017-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Tue Nov  5 06:58:25 UTC 2024

  System load:  0.31              Processes:             106
  Usage of /:   60.9% of 6.71GB   Users logged in:       0
  Memory usage: 78%               IPv4 address for enX0: 172.31.28.76
  Swap usage:   0%

 * Ubuntu Pro delivers the most comprehensive open source security and
   compliance features.

   https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

19 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


Last login: Tue Nov  5 04:15:50 2024 from 73.162.189.68
ubuntu@ip-172-31-28-76:~$ mongod --shardsvr --replSet shard1ReplSet --port 27020 --dbpath /data/shard1 --bind_ip 0.0.0.0
```

Instance 6:



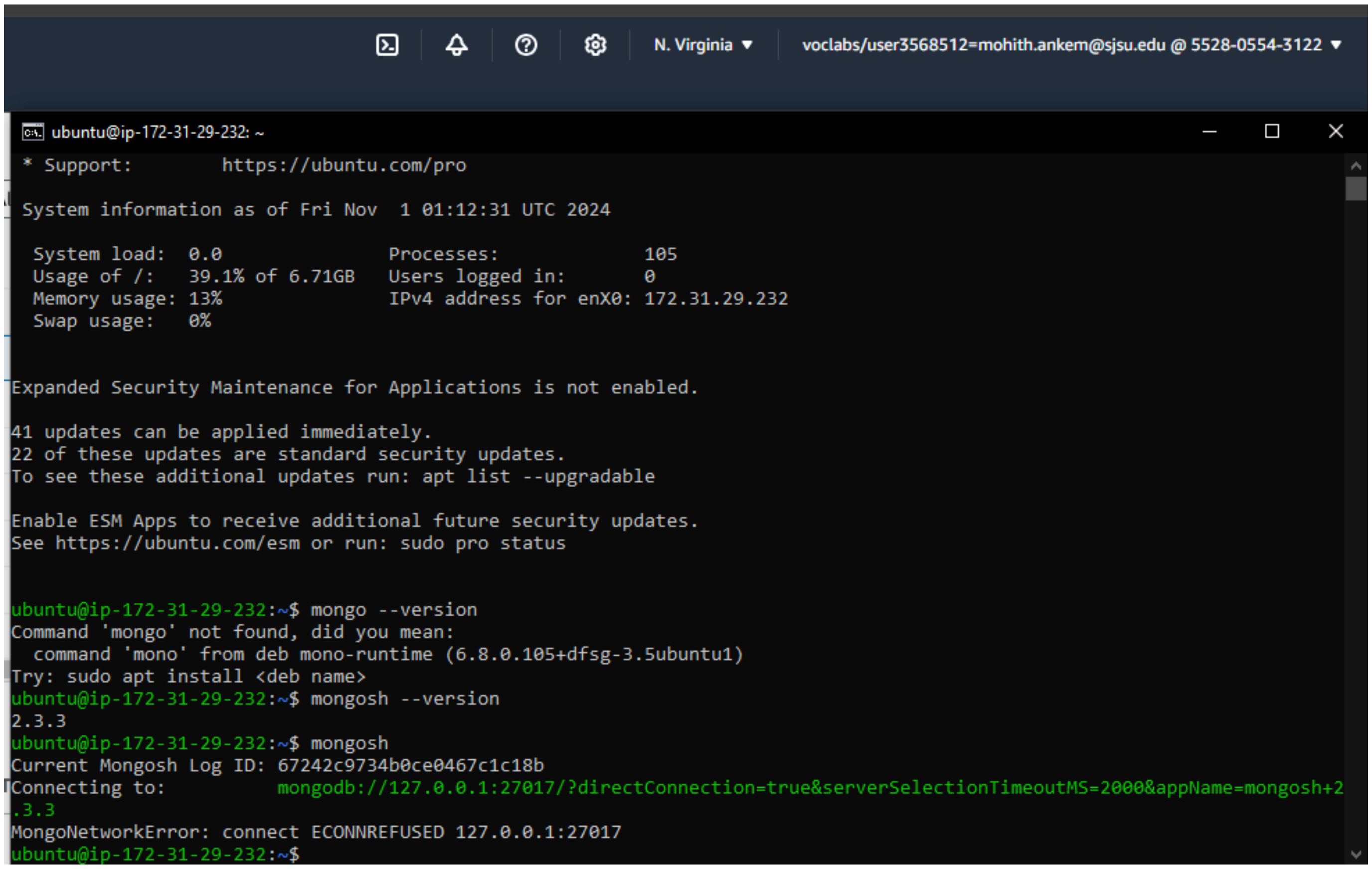
```
mongosh mongodb://127.0.0.1:27021/?directConnection=true&serverSelectionTimeoutMS=2000
C:\Users\Checkout\Desktop\NoSQL>ssh -i C:/Users/Checkout/Desktop/NoSQL/mohithpair.pem ubuntu@52.91.53.129
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1017-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Mon Nov  4 08:07:42 UTC 2024

  System load:  0.0               Processes:             106
  Usage of /:   55.6% of 6.71GB   Users logged in:       0
  Memory usage: 9%                IPv4 address for enX0: 172.31.29.118
  Swap usage:   0%

 * Ubuntu Pro delivers the most comprehensive open source security and
   compliance features.

   https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

19 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


Last login: Mon Nov  4 08:07:43 2024 from 73.162.189.68
ubuntu@ip-172-31-29-118:~$ mongod --shardsvr --replSet shard1ReplSet --port 27020 --dbpath /data/shard1 -
```

Instance 7:

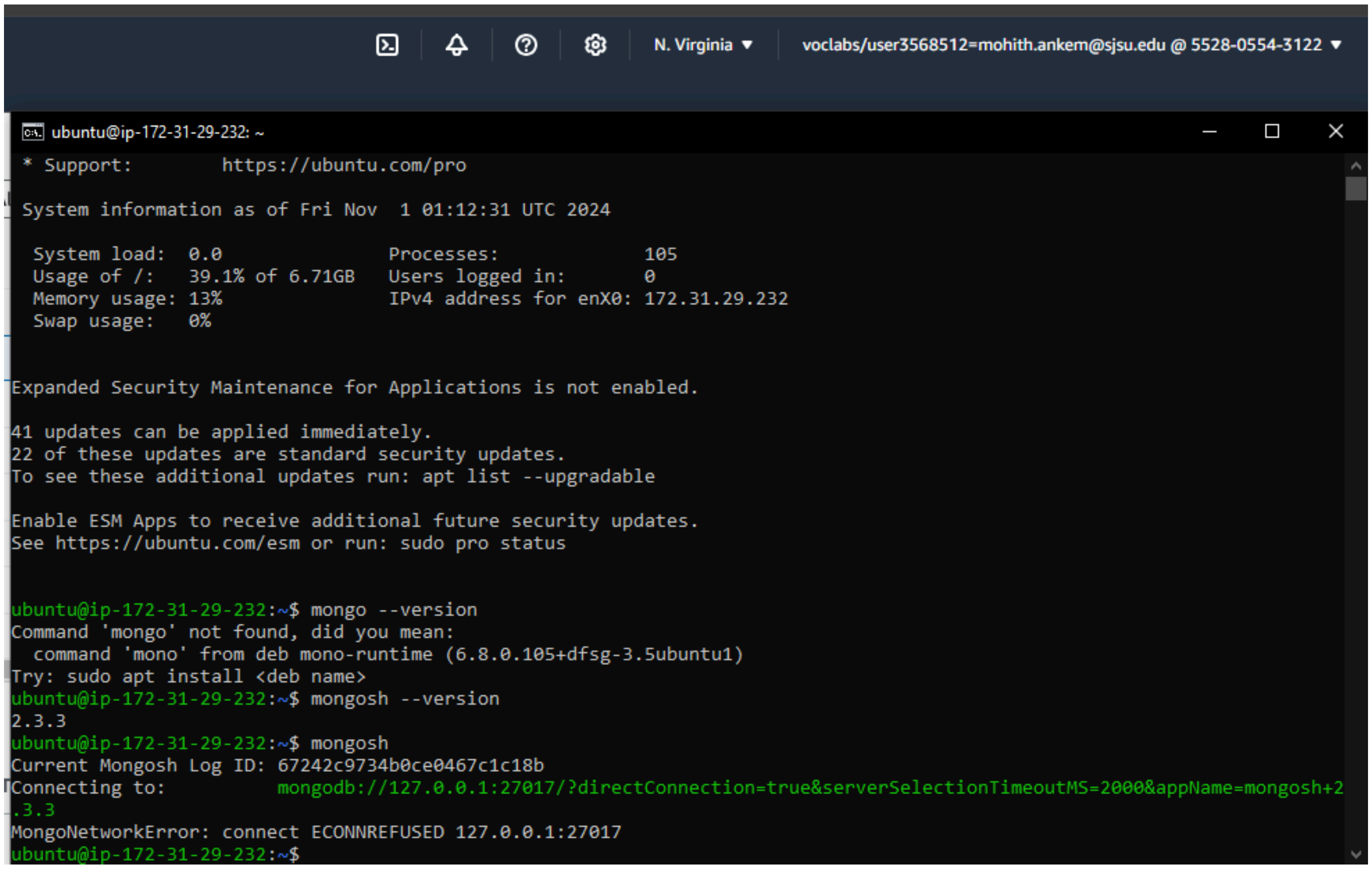


```
mongosh mongodb://127.0.0.1:27022/?directConnection=true&serverSelectionTimeoutMS=2000

C:\Users\Checkout\Desktop\NoSQL>ssh -i C:/Users/Checkout/Desktop/NoSQL/mohithpair.pem ubuntu@52.91.16.14
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Fri Nov  1 09:37:46 UTC 2024

  System load:  0.0               Processes:             106
  Usage of /:   43.7% of 6.71GB   Users logged in:       0
  Memory usage: 15%               IPv4 address for enX0: 172.31.28.155
  Swap usage:   0%

 * Ubuntu Pro delivers the most comprehensive open source security and
   compliance features.

   https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

19 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


*** System restart required ***
ubuntu@ip-172-31-28-155:~$ su^C mkdir -p /data/shard1 /data/shard2 /data/shard3
```

3. Install MongoDB in each node (i.e. instance)


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
ubuntu@ip-172-31-29-232: ~

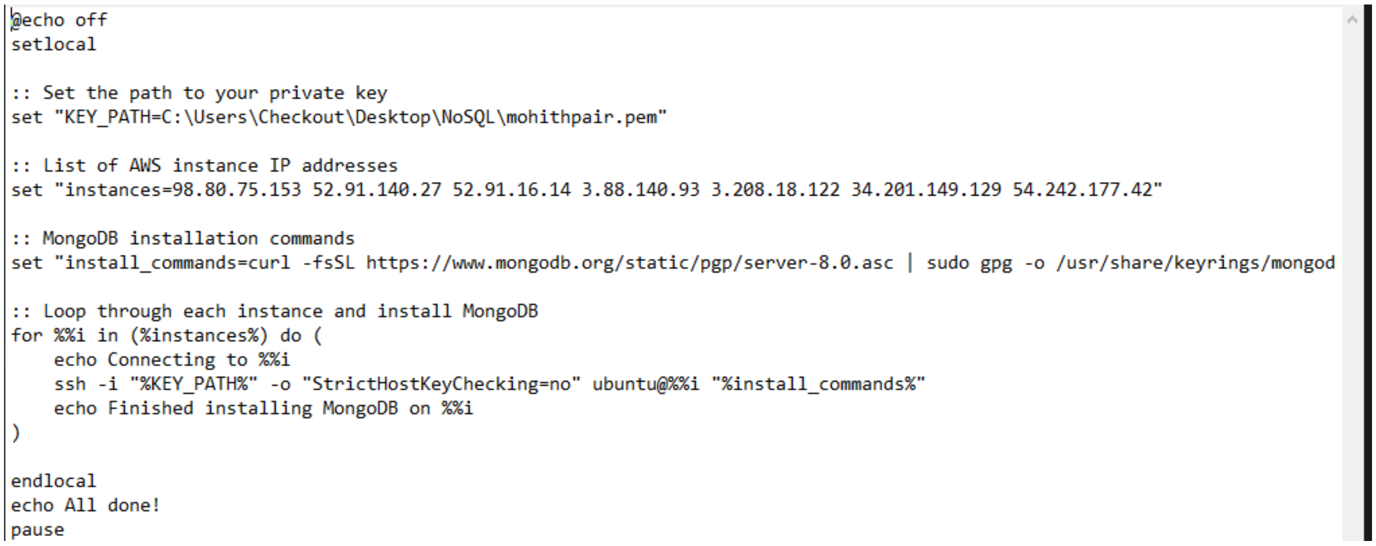
```
 * Support:        https://ubuntu.com/pro

 System information as of Fri Nov  1 01:12:31 UTC 2024

  System load:  0.0               Processes:             105
  Usage of /:   39.1% of 6.71GB   Users logged in:       0
  Memory usage: 13%               IPv4 address for enX0: 172.31.29.232
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

41 updates can be applied immediately.
22 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


ubuntu@ip-172-31-29-232:~$ mongo --version
Command 'mongo' not found, did you mean:
  command 'mono' from deb mono-runtime (6.8.0.105+dfsg-3.5ubuntu1)
Try: sudo apt install <deb name>
ubuntu@ip-172-31-29-232:~$ mongosh --version
2.3.3
ubuntu@ip-172-31-29-232:~$ mongosh
Current Mongosh Log ID: 67242c9734b0ce0467c1c18b
Connecting to:          mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2
.3.3
MongoNetworkError: connect ECONNREFUSED 127.0.0.1:27017
ubuntu@ip-172-31-29-232:~$
```


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
ubuntu@ip-172-31-29-232: ~

```
 * Support:        https://ubuntu.com/pro

 System information as of Fri Nov  1 01:12:31 UTC 2024

  System load:  0.0               Processes:             105
  Usage of /:   39.1% of 6.71GB   Users logged in:       0
  Memory usage: 13%               IPv4 address for enX0: 172.31.29.232
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

41 updates can be applied immediately.
22 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


ubuntu@ip-172-31-29-232:~$ mongo --version
Command 'mongo' not found, did you mean:
  command 'mono' from deb mono-runtime (6.8.0.105+dfsg-3.5ubuntu1)
Try: sudo apt install <deb name>
ubuntu@ip-172-31-29-232:~$ mongosh --version
2.3.3
ubuntu@ip-172-31-29-232:~$ mongosh
Current Mongosh Log ID: 67242c9734b0ce0467c1c18b
Connecting to:          mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2
.3.3
MongoNetworkError: connect ECONNREFUSED 127.0.0.1:27017
ubuntu@ip-172-31-29-232:~$
```

The Above two are proof that mongo is installed in the respective server.

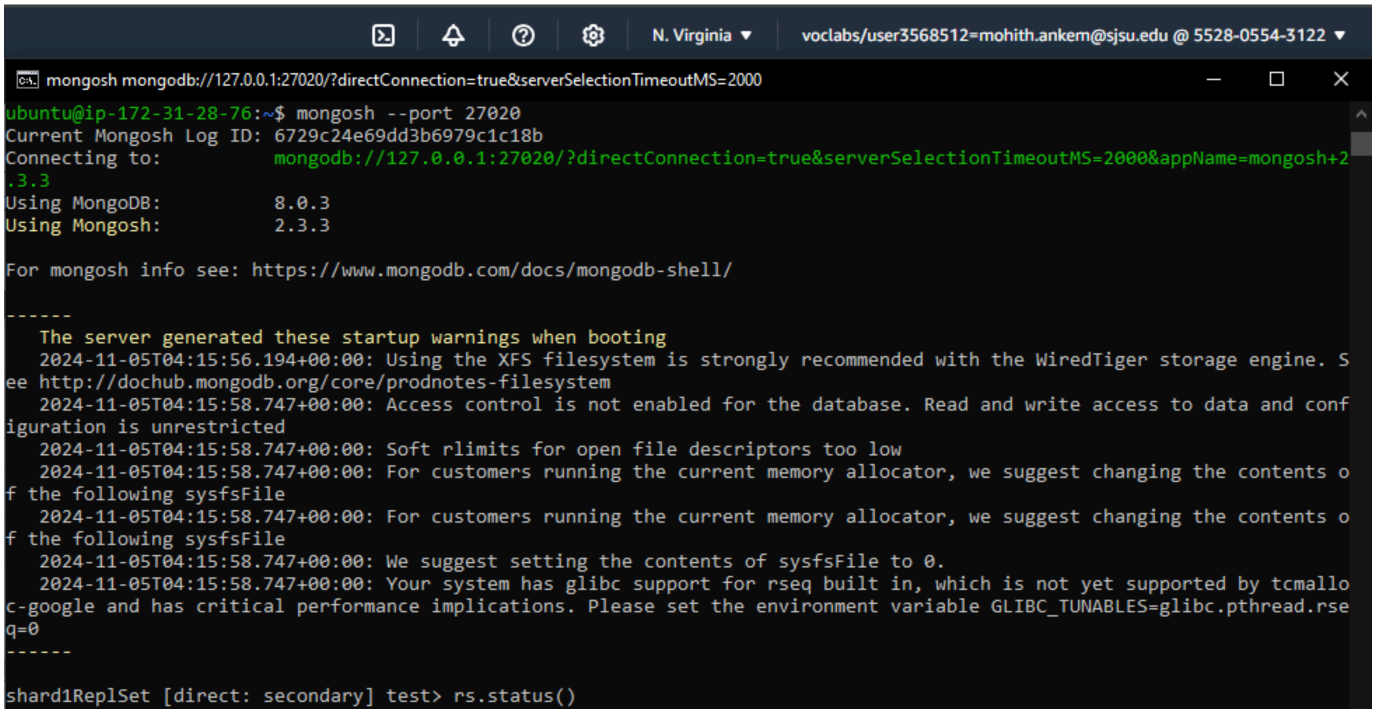
This is done using the install_mongo.bat script shown below:



```
@echo off
setlocal

:: Set the path to your private key
set "KEY_PATH=C:\Users\Checkout\Desktop\NoSQL\mohithpair.pem"

:: List of AWS instance IP addresses
set "instances=98.80.75.153 52.91.140.27 52.91.16.14 3.88.140.93 3.208.18.122 34.201.149.129 54.242.177.42"

:: MongoDB installation commands
set "install_commands=curl -fsSL https://www.mongodb.org/static/pgp/server-8.0.asc | sudo gpg -o /usr/share/keyrings/mongod

:: Loop through each instance and install MongoDB
for %%i in (%instances%) do (
    echo Connecting to %%i
    ssh -i "%KEY_PATH%" -o "StrictHostKeyChecking=no" ubuntu@%%i "%install_commands%"
    echo Finished installing MongoDB on %%i
)

endlocal
echo All done!
pause
```

Using the respective port numbers all the servers are connected to mongos using the command

Mongosh –port 27019



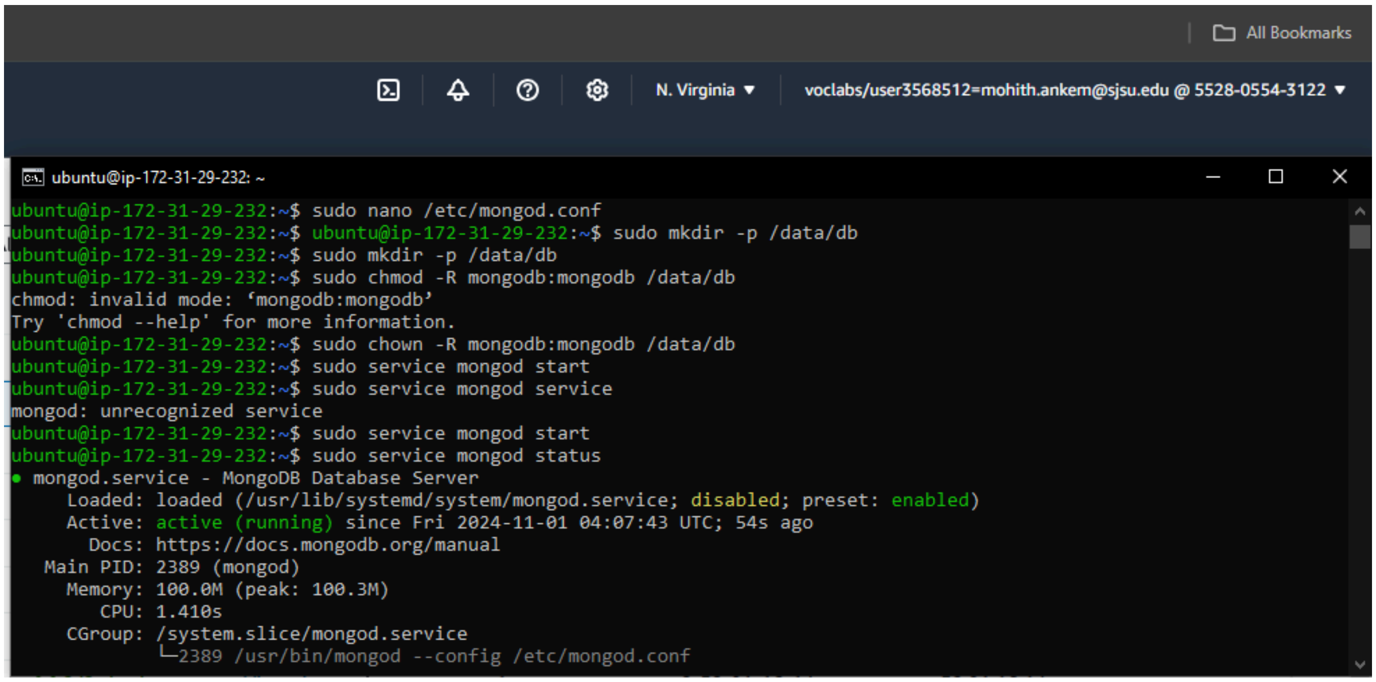
```
mongosh mongodb://127.0.0.1:27020/?directConnection=true&serverSelectionTimeoutMS=2000
ubuntu@ip-172-31-28-76:~$ mongosh --port 27020
Current Mongosh Log ID: 6729c24e69dd3b6979c1c18b
Connecting to:          mongodb://127.0.0.1:27020/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2
.3.3
Using MongoDB:          8.0.3
Using Mongosh:          2.3.3

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/

------
   The server generated these startup warnings when booting
   2024-11-05T04:15:56.194+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. S
ee http://dochub.mongodb.org/core/prodnotes-filesystem
   2024-11-05T04:15:58.747+00:00: Access control is not enabled for the database. Read and write access to data and conf
iguration is unrestricted
   2024-11-05T04:15:58.747+00:00: Soft rlimits for open file descriptors too low
   2024-11-05T04:15:58.747+00:00: For customers running the current memory allocator, we suggest changing the contents o
f the following sysfsFile
   2024-11-05T04:15:58.747+00:00: For customers running the current memory allocator, we suggest changing the contents o
f the following sysfsFile
   2024-11-05T04:15:58.747+00:00: We suggest setting the contents of sysfsFile to 0.
   2024-11-05T04:15:58.747+00:00: Your system has glibc support for rseq built in, which is not yet supported by tcmallo
c-google and has critical performance implications. Please set the environment variable GLIBC_TUNABLES=glibc.pthread.rse
q=0
------

shard1ReplSet [direct: secondary] test> rs.status()
```

4. <u>Create a directory to store database in each node</u>

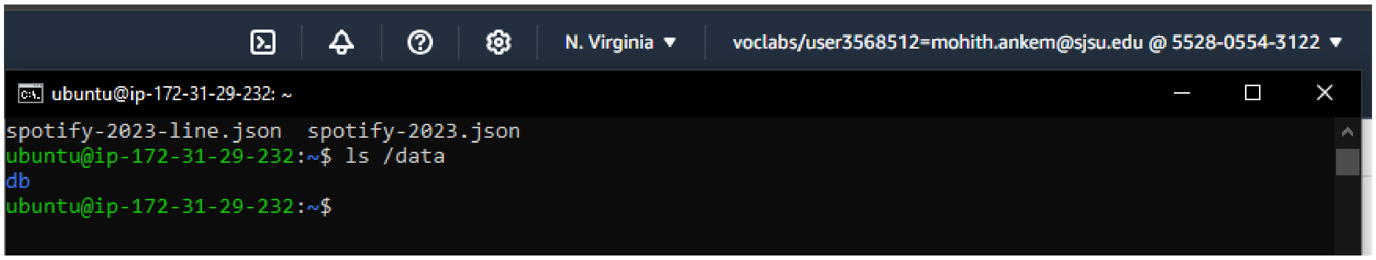
The Commands that are used to make directory in the servers to store database in each node are as follows:

- Sudo mkdir -p /data/db
- Sudo chown -R mongodb:mongodb /data/db

The Screenshot is attached below to make the directories which are done in all servers.


All Bookmarks
N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
ubuntu@ip-172-31-29-232: ~

```
ubuntu@ip-172-31-29-232:~$ sudo nano /etc/mongod.conf
ubuntu@ip-172-31-29-232:~$ ubuntu@ip-172-31-29-232:~$ sudo mkdir -p /data/db
ubuntu@ip-172-31-29-232:~$ sudo mkdir -p /data/db
ubuntu@ip-172-31-29-232:~$ sudo chmod -R mongodb:mongodb /data/db
chmod: invalid mode: ‘mongodb:mongodb’
Try 'chmod --help' for more information.
ubuntu@ip-172-31-29-232:~$ sudo chown -R mongodb:mongodb /data/db
ubuntu@ip-172-31-29-232:~$ sudo service mongod start
ubuntu@ip-172-31-29-232:~$ sudo service mongod service
mongod: unrecognized service
ubuntu@ip-172-31-29-232:~$ sudo service mongod start
ubuntu@ip-172-31-29-232:~$ sudo service mongod status
● mongod.service - MongoDB Database Server
     Loaded: loaded (/usr/lib/systemd/system/mongod.service; disabled; preset: enabled)
     Active: active (running) since Fri 2024-11-01 04:07:43 UTC; 54s ago
       Docs: https://docs.mongodb.org/manual
   Main PID: 2389 (mongod)
     Memory: 100.0M (peak: 100.3M)
        CPU: 1.410s
     CGroup: /system.slice/mongod.service
             └─2389 /usr/bin/mongod --config /etc/mongod.conf
```


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
ubuntu@ip-172-31-29-232: ~

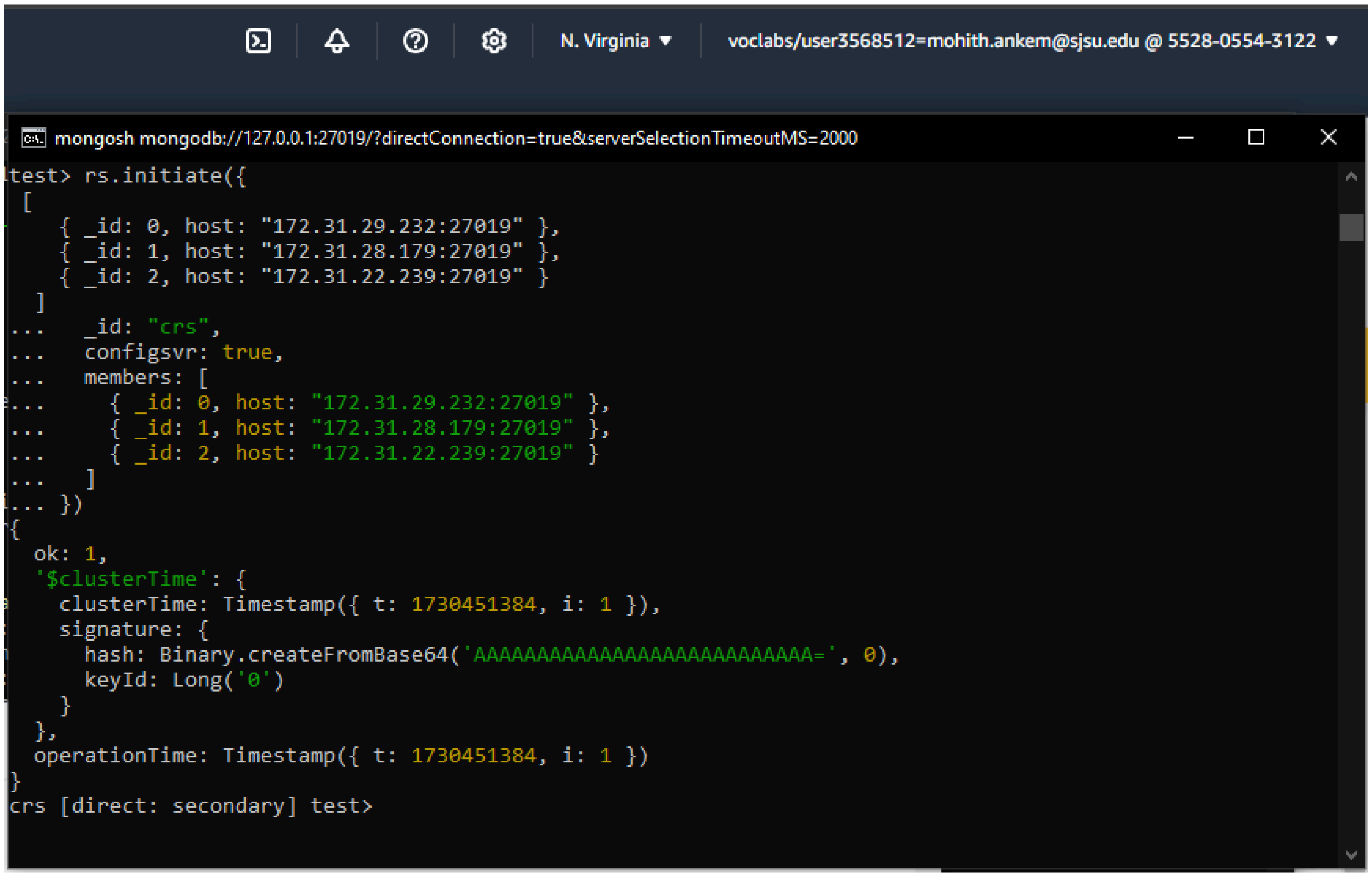
```
spotify-2023-line.json  spotify-2023.json
ubuntu@ip-172-31-29-232:~$ ls /data
db
ubuntu@ip-172-31-29-232:~$
```

5. Specify Public and Private IP Addresses of AWS instances used in your solution.

Instance Name	Public IP	Private IP
Node1_Instance	54.197.4.176	172.31.29.232
Node2_server2	3.87.107.88	172.31.28.179
Node3_server3	18.212.98.236	172.31.22.239
Node4_server4	23.22.61.21	172.31.27.161
Node5_server5	3.80.97.241	172.31.28.76
Node6_server6	3.95.243.227	172.31.29.118
Node7_server7	54.226.141.42	172.31.28.155

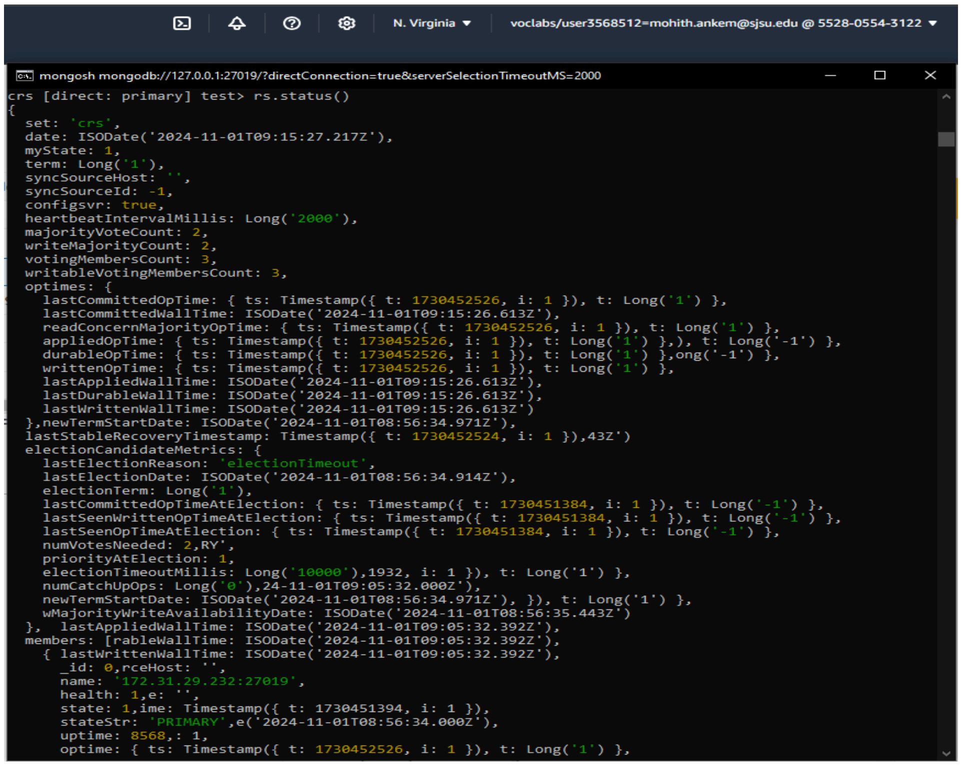
6. Set up and launch three config servers in a replica set.

I have used the rs.initiate() command to initialize the replica set for the config servers.

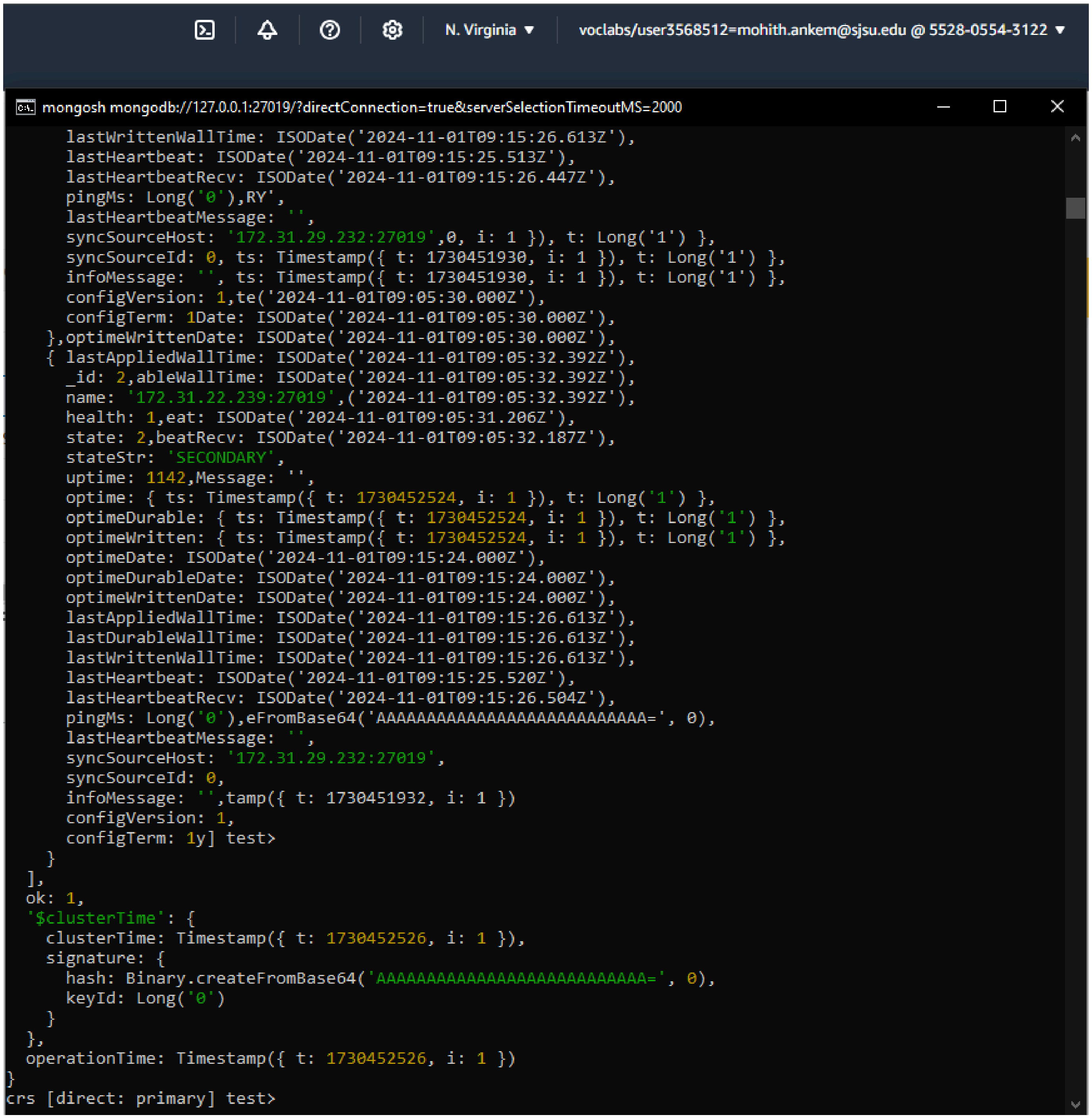

N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
mongosh mongodb://127.0.0.1:27019/?directConnection=true&serverSelectionTimeoutMS=2000

```
test> rs.initiate({
 [
   { _id: 0, host: "172.31.29.232:27019" },
   { _id: 1, host: "172.31.28.179:27019" },
   { _id: 2, host: "172.31.22.239:27019" }
  ]
...   _id: "crs",
...   configsvr: true,
...   members: [
...     { _id: 0, host: "172.31.29.232:27019" },
...     { _id: 1, host: "172.31.28.179:27019" },
...     { _id: 2, host: "172.31.22.239:27019" }
...   ]
... })
{
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1730451384, i: 1 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: Long('0')
    }
  },
  operationTime: Timestamp({ t: 1730451384, i: 1 })
}
crs [direct: secondary] test>
```

rs.status():


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
mongosh mongodb://127.0.0.1:27019/?directConnection=true&serverSelectionTimeoutMS=2000

```
crs [direct: primary] test> rs.status()
{
  set: 'crs',
  date: ISODate('2024-11-01T09:15:27.217Z'),
  myState: 1,
  term: Long('1'),
  syncSourceHost: '',
  syncSourceId: -1,
  configsvr: true,
  heartbeatIntervalMillis: Long('2000'),
  majorityVoteCount: 2,
  writeMajorityCount: 2,
  votingMembersCount: 3,
  writableVotingMembersCount: 3,
  optimes: {
    lastCommittedOpTime: { ts: Timestamp({ t: 1730452526, i: 1 }), t: Long('1') },
    lastCommittedWallTime: ISODate('2024-11-01T09:15:26.613Z'),
    readConcernMajorityOpTime: { ts: Timestamp({ t: 1730452526, i: 1 }), t: Long('1') },
    appliedOpTime: { ts: Timestamp({ t: 1730452526, i: 1 }), t: Long('1') },), t: Long('-1') },
    durableOpTime: { ts: Timestamp({ t: 1730452526, i: 1 }), t: Long('1') },ong('-1') },
    writtenOpTime: { ts: Timestamp({ t: 1730452526, i: 1 }), t: Long('1') },
    lastAppliedWallTime: ISODate('2024-11-01T09:15:26.613Z'),
    lastDurableWallTime: ISODate('2024-11-01T09:15:26.613Z'),
    lastWrittenWallTime: ISODate('2024-11-01T09:15:26.613Z')
  },newTermStartDate: ISODate('2024-11-01T08:56:34.971Z'),
  lastStableRecoveryTimestamp: Timestamp({ t: 1730452524, i: 1 }),43Z')
  electionCandidateMetrics: {
    lastElectionReason: 'electionTimeout',
    lastElectionDate: ISODate('2024-11-01T08:56:34.914Z'),
    electionTerm: Long('1'),
    lastCommittedOpTimeAtElection: { ts: Timestamp({ t: 1730451384, i: 1 }), t: Long('-1') },
    lastSeenWrittenOpTimeAtElection: { ts: Timestamp({ t: 1730451384, i: 1 }), t: Long('-1') },
    lastSeenOpTimeAtElection: { ts: Timestamp({ t: 1730451384, i: 1 }), t: Long('-1') },
    numVotesNeeded: 2,RY',
    priorityAtElection: 1,
    electionTimeoutMillis: Long('10000'),1932, i: 1 }), t: Long('1') },
    numCatchUpOps: Long('0'),24-11-01T09:05:32.000Z'),
    newTermStartDate: ISODate('2024-11-01T08:56:34.971Z'), }), t: Long('1') },
    wMajorityWriteAvailabilityDate: ISODate('2024-11-01T08:56:35.443Z')
  },  lastAppliedWallTime: ISODate('2024-11-01T09:05:32.392Z'),
  members: [rableWallTime: ISODate('2024-11-01T09:05:32.392Z'),
    { lastWrittenWallTime: ISODate('2024-11-01T09:05:32.392Z'),
      _id: 0,rceHost: '',
      name: '172.31.29.232:27019',
      health: 1,e: '',
      state: 1,ime: Timestamp({ t: 1730451394, i: 1 }),
      stateStr: 'PRIMARY',e('2024-11-01T08:56:34.000Z'),
      uptime: 8568,: 1,
      optime: { ts: Timestamp({ t: 1730452526, i: 1 }), t: Long('1') },
```

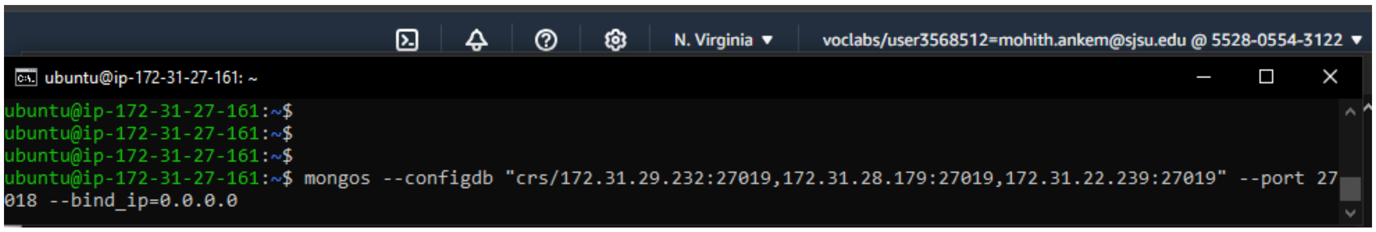

N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
mongosh mongodb://127.0.0.1:27019/?directConnection=true&serverSelectionTimeoutMS=2000

```
      lastWrittenWallTime: ISODate('2024-11-01T09:15:26.613Z'),
      lastHeartbeat: ISODate('2024-11-01T09:15:25.513Z'),
      lastHeartbeatRecv: ISODate('2024-11-01T09:15:26.447Z'),
      pingMs: Long('0'),RY',
      lastHeartbeatMessage: '',
      syncSourceHost: '172.31.29.232:27019',0, i: 1 }), t: Long('1') },
      syncSourceId: 0, ts: Timestamp({ t: 1730451930, i: 1 }), t: Long('1') },
      infoMessage: '', ts: Timestamp({ t: 1730451930, i: 1 }), t: Long('1') },
      configVersion: 1,te('2024-11-01T09:05:30.000Z'),
      configTerm: 1Date: ISODate('2024-11-01T09:05:30.000Z'),
    },optimeWrittenDate: ISODate('2024-11-01T09:05:30.000Z'),
    { lastAppliedWallTime: ISODate('2024-11-01T09:05:32.392Z'),
      _id: 2,ableWallTime: ISODate('2024-11-01T09:05:32.392Z'),
      name: '172.31.22.239:27019',('2024-11-01T09:05:32.392Z'),
      health: 1,eat: ISODate('2024-11-01T09:05:31.206Z'),
      state: 2,beatRecv: ISODate('2024-11-01T09:05:32.187Z'),
      stateStr: 'SECONDARY',
      uptime: 1142,Message: '',
      optime: { ts: Timestamp({ t: 1730452524, i: 1 }), t: Long('1') },
      optimeDurable: { ts: Timestamp({ t: 1730452524, i: 1 }), t: Long('1') },
      optimeWritten: { ts: Timestamp({ t: 1730452524, i: 1 }), t: Long('1') },
      optimeDate: ISODate('2024-11-01T09:15:24.000Z'),
      optimeDurableDate: ISODate('2024-11-01T09:15:24.000Z'),
      optimeWrittenDate: ISODate('2024-11-01T09:15:24.000Z'),
      lastAppliedWallTime: ISODate('2024-11-01T09:15:26.613Z'),
      lastDurableWallTime: ISODate('2024-11-01T09:15:26.613Z'),
      lastWrittenWallTime: ISODate('2024-11-01T09:15:26.613Z'),
      lastHeartbeat: ISODate('2024-11-01T09:15:25.520Z'),
      lastHeartbeatRecv: ISODate('2024-11-01T09:15:26.504Z'),
      pingMs: Long('0'),eFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      lastHeartbeatMessage: '',
      syncSourceHost: '172.31.29.232:27019',
      syncSourceId: 0,
      infoMessage: '',tamp({ t: 1730451932, i: 1 })
      configVersion: 1,
      configTerm: 1y] test>
    }
  ],
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1730452526, i: 1 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: Long('0')
    }
  },
  operationTime: Timestamp({ t: 1730452526, i: 1 })
}
crs [direct: primary] test>
```

7. Connect mongos to each config server.

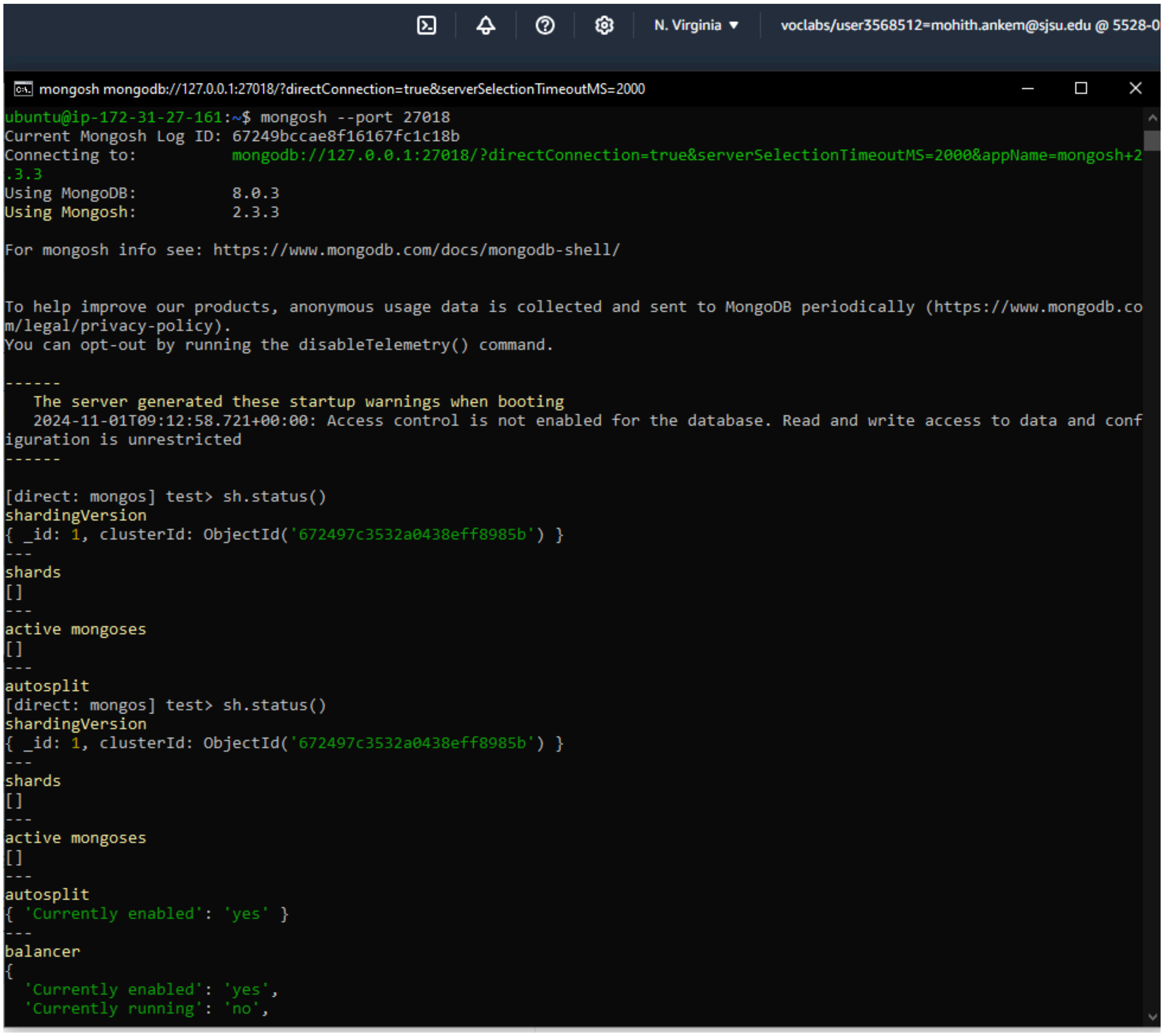
The command to run mongos server is: **mongos –config /etc/mongos.conf**

Using this command I modified the sharding configuration which will setup mongos server to each config server.


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
ubuntu@ip-172-31-27-161: ~

```
ubuntu@ip-172-31-27-161:~$
ubuntu@ip-172-31-27-161:~$
ubuntu@ip-172-31-27-161:~$
ubuntu@ip-172-31-27-161:~$ mongos --configdb "crs/172.31.29.232:27019,172.31.28.179:27019,172.31.22.239:27019" --port 27
018 --bind_ip=0.0.0.0
```

Using the above command in the command prompt , I kept the server running and opened another terminal to access the monos server and sh.status() after executing the command is attached below.


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
ubuntu@ip-172-31-27-161:~$ mongosh --port 27018
Current Mongosh Log ID: 67249bccae8f16167fc1c18b
Connecting to:          mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2
.3.3
Using MongoDB:          8.0.3
Using Mongosh:          2.3.3

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/


To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.co
m/legal/privacy-policy).
You can opt-out by running the disableTelemetry() command.

------
   The server generated these startup warnings when booting
   2024-11-01T09:12:58.721+00:00: Access control is not enabled for the database. Read and write access to data and conf
iguration is unrestricted
------

[direct: mongos] test> sh.status()
shardingVersion
{ _id: 1, clusterId: ObjectId('672497c3532a0438eff8985b') }
---
shards
[]
---
active mongoses
[]
---
autosplit
[direct: mongos] test> sh.status()
shardingVersion
{ _id: 1, clusterId: ObjectId('672497c3532a0438eff8985b') }
---
shards
[]
---
active mongoses
[]
---
autosplit
{ 'Currently enabled': 'yes' }
---
balancer
{
  'Currently enabled': 'yes',
  'Currently running': 'no',
```

8. Set up and launch each of the three shards. Make sure to include the result of sh.status() before adding the shards, which will be done in the next task.

On all of the shard servers we run the following commands:

sudo mkdir -p /data/shard1 /data/shard2 /data/shard3

sudo chown ubuntu:ubuntu /data/shard1 /data/shard2 /data/shard3 /var/log/mongodb

mongod --shardsvr --replSet shard1ReplSet --port 27020 --dbpath /data/shard1 --bind_ip 0.0.0.0 --fork --logpath /var/log/mongodb/shard1.log

mongod --shardsvr --replSet shard2ReplSet --port 27021 --dbpath /data/shard2 --bind_ip 0.0.0.0 --fork --logpath /var/log/mongodb/shard2.log

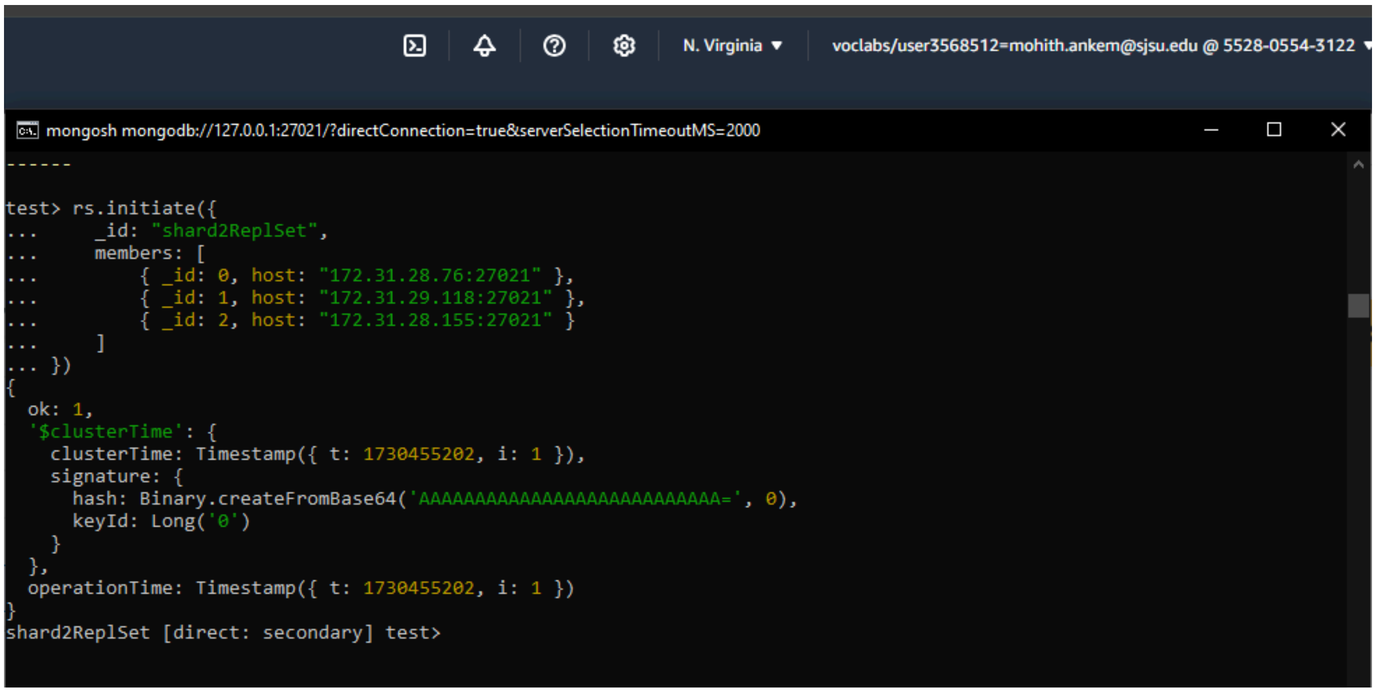
mongod --shardsvr --replSet shard3ReplSet --port 27022 --dbpath /data/shard3 --bind_ip 0.0.0.0 --fork --logpath /var/log/mongodb/shard3.log

After running these 3 commands on all three shard servers replica sets are created for each shard.Then we connect to mongosh to see the status.

Shard 1:

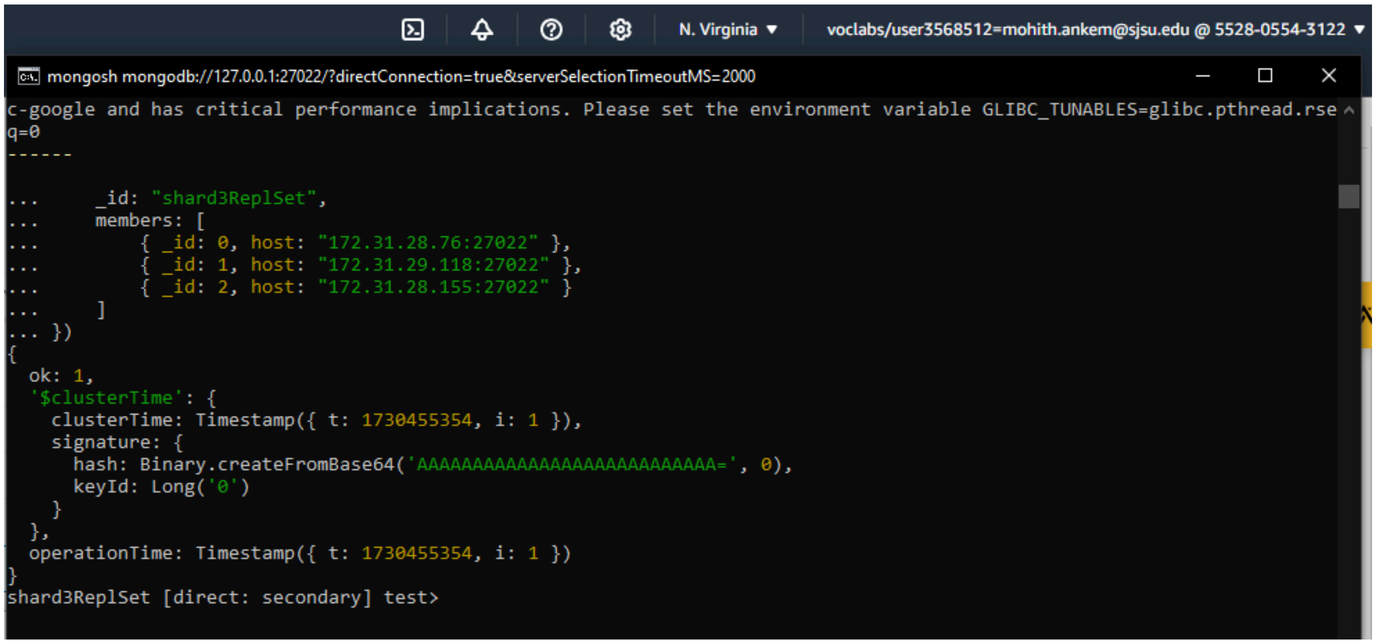
```
rs.initiate({

    _id: "shard1ReplSet",

    members: [

        { _id: 0, host: "172.31.28.36:27020" },

        { _id: 1, host: "172.31.29.118:27020" },

        { _id: 2, host: "172.31.28.155:27020" }

    ]

})
```

Shard2-Initiate:


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
mongosh mongodb://127.0.0.1:27021/?directConnection=true&serverSelectionTimeoutMS=2000

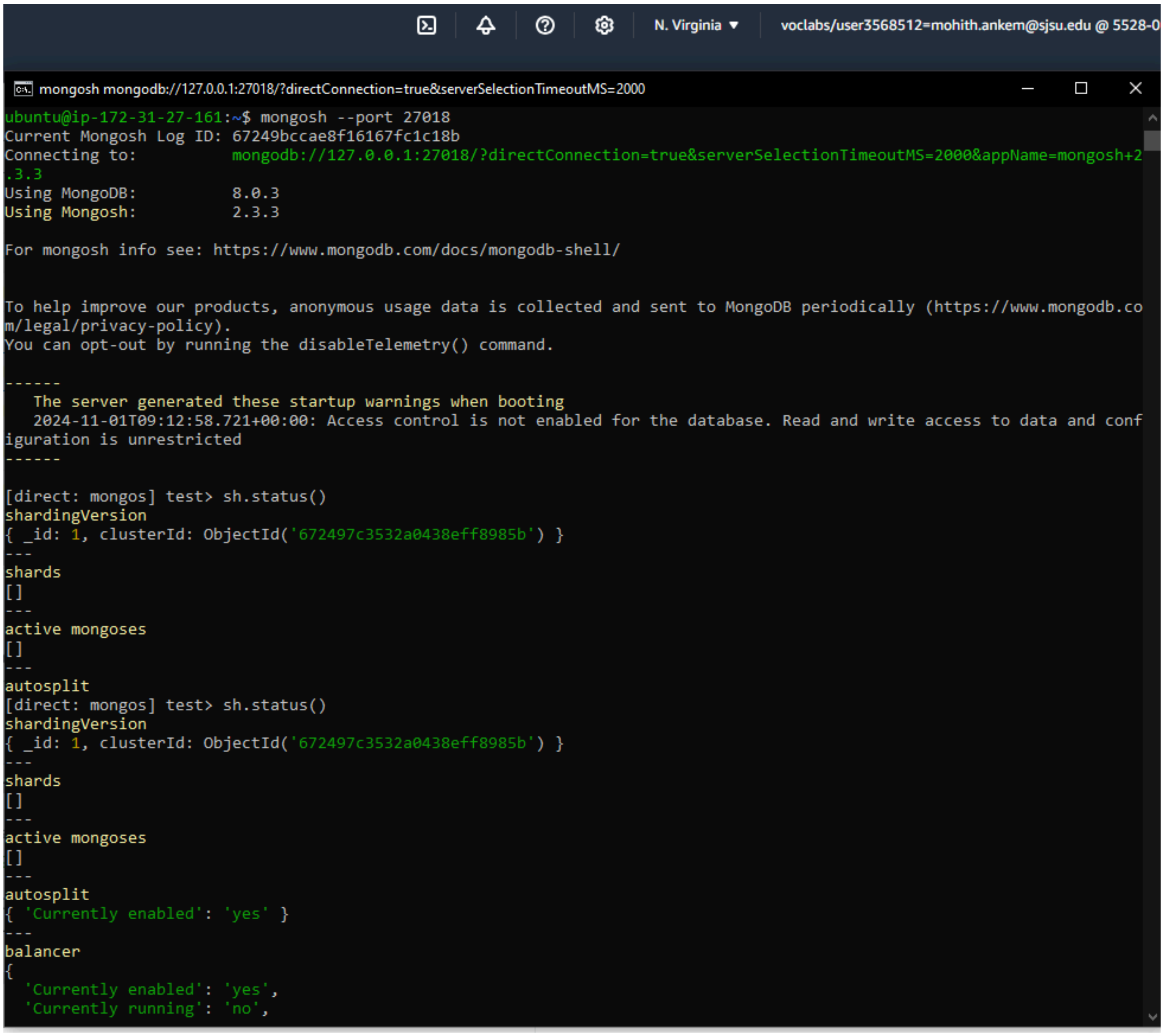
```
------

test> rs.initiate({
...     _id: "shard2ReplSet",
...     members: [
...         { _id: 0, host: "172.31.28.76:27021" },
...         { _id: 1, host: "172.31.29.118:27021" },
...         { _id: 2, host: "172.31.28.155:27021" }
...     ]
... })
{
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1730455202, i: 1 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: Long('0')
    }
  },
  operationTime: Timestamp({ t: 1730455202, i: 1 })
}
shard2ReplSet [direct: secondary] test>
```

Shard 3-Initiate:


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
mongosh mongodb://127.0.0.1:27022/?directConnection=true&serverSelectionTimeoutMS=2000

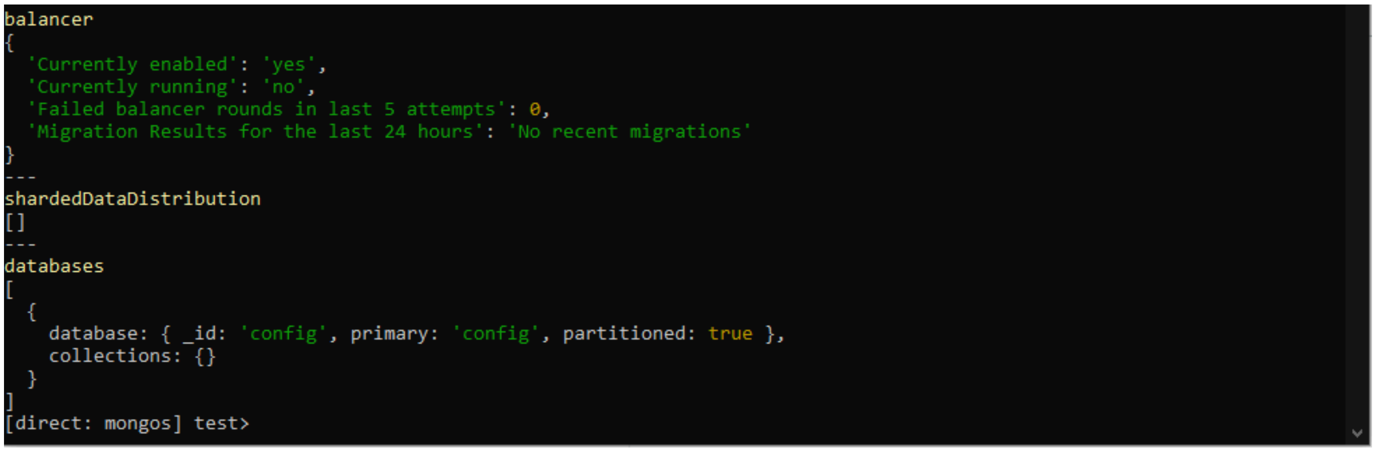
```
c-google and has critical performance implications. Please set the environment variable GLIBC_TUNABLES=glibc.pthread.rse
q=0
------

...     _id: "shard3ReplSet",
...     members: [
...         { _id: 0, host: "172.31.28.76:27022" },
...         { _id: 1, host: "172.31.29.118:27022" },
...         { _id: 2, host: "172.31.28.155:27022" }
...     ]
... })
{
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1730455354, i: 1 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: Long('0')
    }
  },
  operationTime: Timestamp({ t: 1730455354, i: 1 })
}
shard3ReplSet [direct: secondary] test>
```

sh.status() before adding shards:



N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

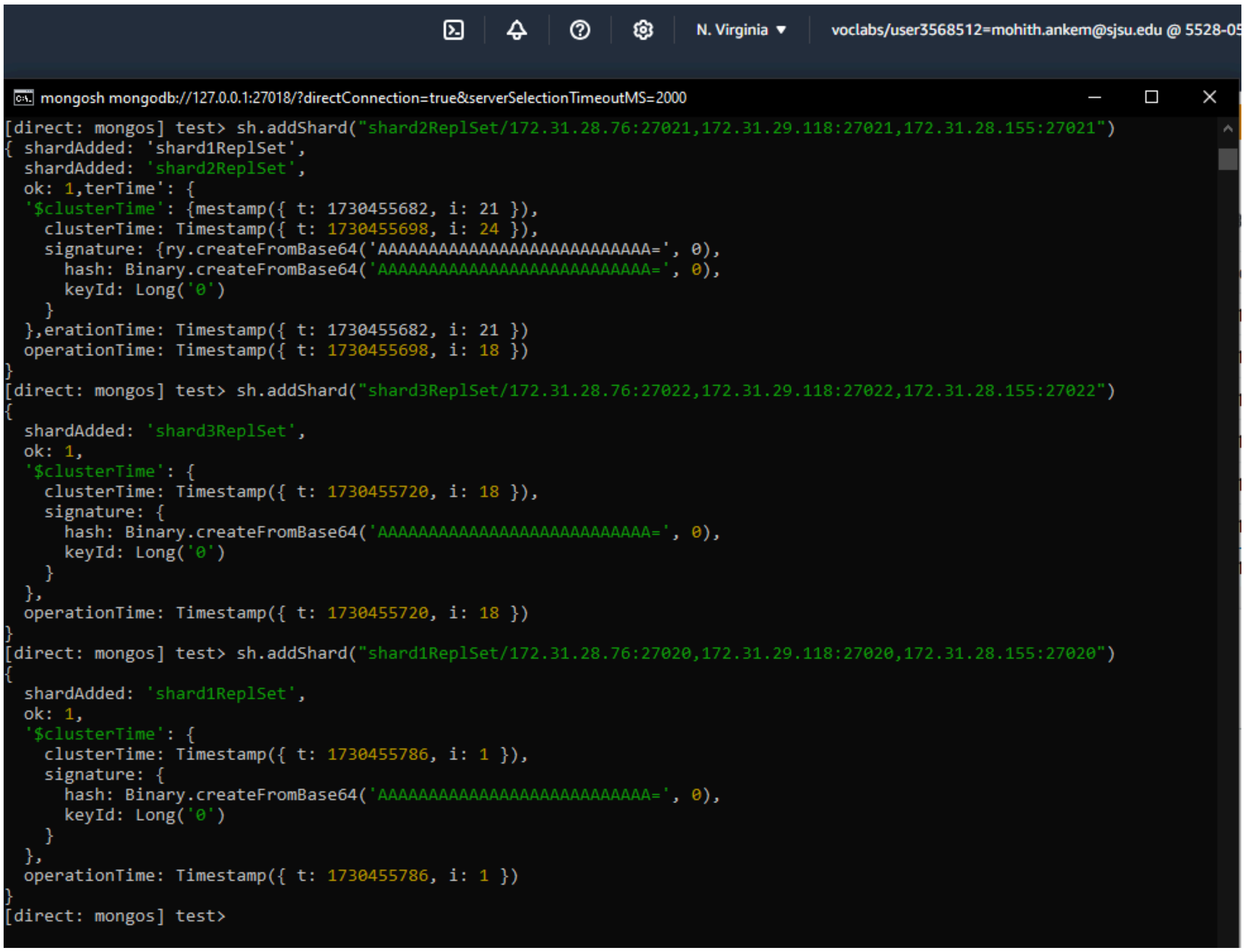
```
ubuntu@ip-172-31-27-161:~$ mongosh --port 27018
Current Mongosh Log ID: 67249bccae8f16167fc1c18b
Connecting to:          mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2
.3.3
Using MongoDB:          8.0.3
Using Mongosh:          2.3.3

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/


To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.co
m/legal/privacy-policy).
You can opt-out by running the disableTelemetry() command.

------
   The server generated these startup warnings when booting
   2024-11-01T09:12:58.721+00:00: Access control is not enabled for the database. Read and write access to data and conf
iguration is unrestricted
------

[direct: mongos] test> sh.status()
shardingVersion
{ _id: 1, clusterId: ObjectId('672497c3532a0438eff8985b') }
---
shards
[]
---
active mongoses
[]
---
autosplit
[direct: mongos] test> sh.status()
shardingVersion
{ _id: 1, clusterId: ObjectId('672497c3532a0438eff8985b') }
---
shards
[]
---
active mongoses
[]
---
autosplit
{ 'Currently enabled': 'yes' }
---
balancer
{
  'Currently enabled': 'yes',
  'Currently running': 'no',
```



```
balancer
{
  'Currently enabled': 'yes',
  'Currently running': 'no',
  'Failed balancer rounds in last 5 attempts': 0,
  'Migration Results for the last 24 hours': 'No recent migrations'
}
---
shardedDataDistribution
[]
---
databases
[
  {
    database: { _id: 'config', primary: 'config', partitioned: true },
    collections: {}
  }
]
[direct: mongos] test>
```

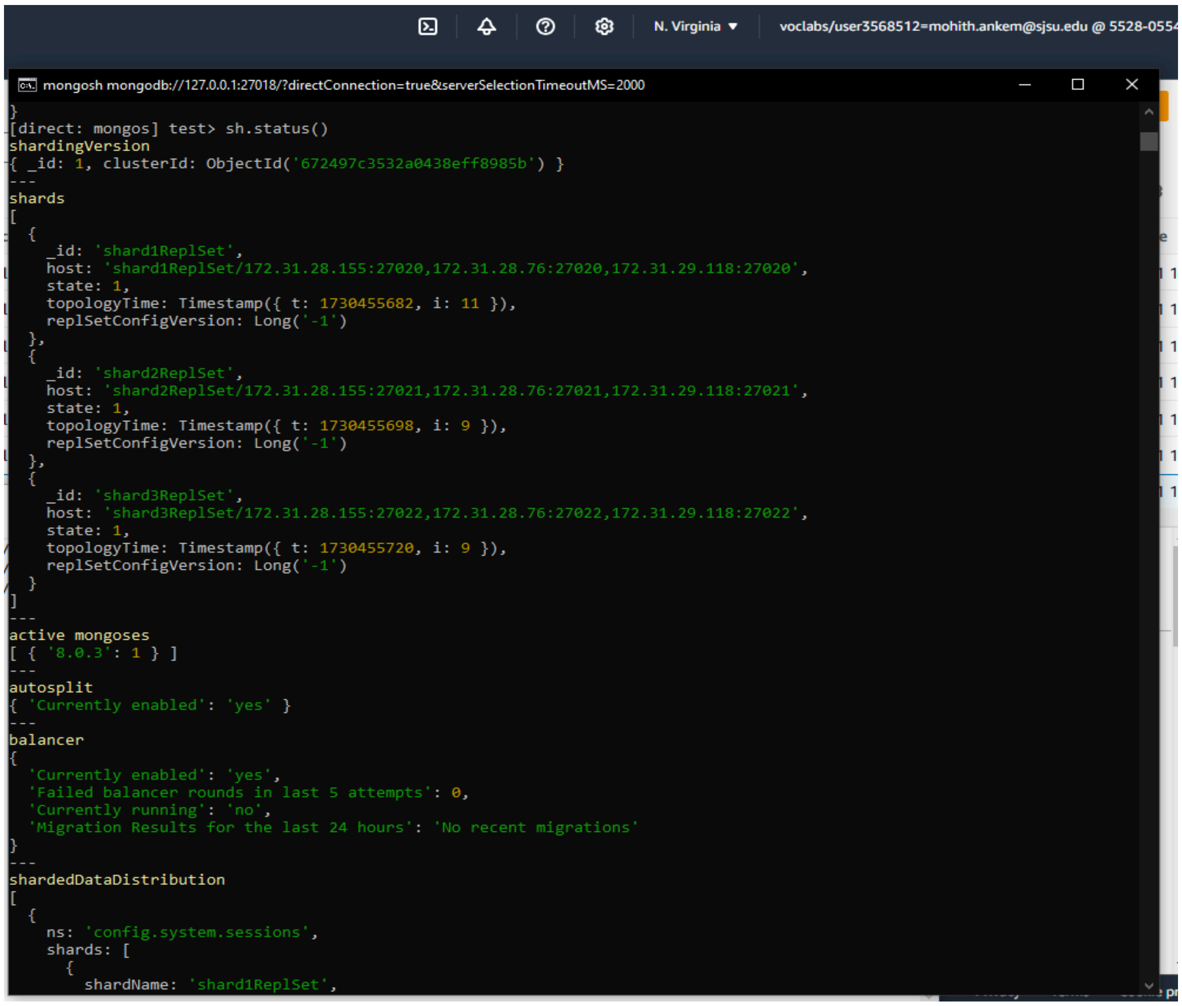
9. Add the shards and ensure you include the result of sh.status() after adding them.

sh.addShard() Command is used to add the shard servers.



```
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000
[direct: mongos] test> sh.addShard("shard2ReplSet/172.31.28.76:27021,172.31.29.118:27021,172.31.28.155:27021")
{ shardAdded: 'shard1ReplSet',
  shardAdded: 'shard2ReplSet',
  ok: 1,terTime': {
  '$clusterTime': {mestamp({ t: 1730455682, i: 21 }),
    clusterTime: Timestamp({ t: 1730455698, i: 24 }),
    signature: {ry.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: Long('0')
    }
  },erationTime: Timestamp({ t: 1730455682, i: 21 })
  operationTime: Timestamp({ t: 1730455698, i: 18 })
}
[direct: mongos] test> sh.addShard("shard3ReplSet/172.31.28.76:27022,172.31.29.118:27022,172.31.28.155:27022")
{
  shardAdded: 'shard3ReplSet',
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1730455720, i: 18 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: Long('0')
    }
  },
  operationTime: Timestamp({ t: 1730455720, i: 18 })
}
[direct: mongos] test> sh.addShard("shard1ReplSet/172.31.28.76:27020,172.31.29.118:27020,172.31.28.155:27020")
{
  shardAdded: 'shard1ReplSet',
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1730455786, i: 1 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: Long('0')
    }
  },
  operationTime: Timestamp({ t: 1730455786, i: 1 })
}
[direct: mongos] test>
```

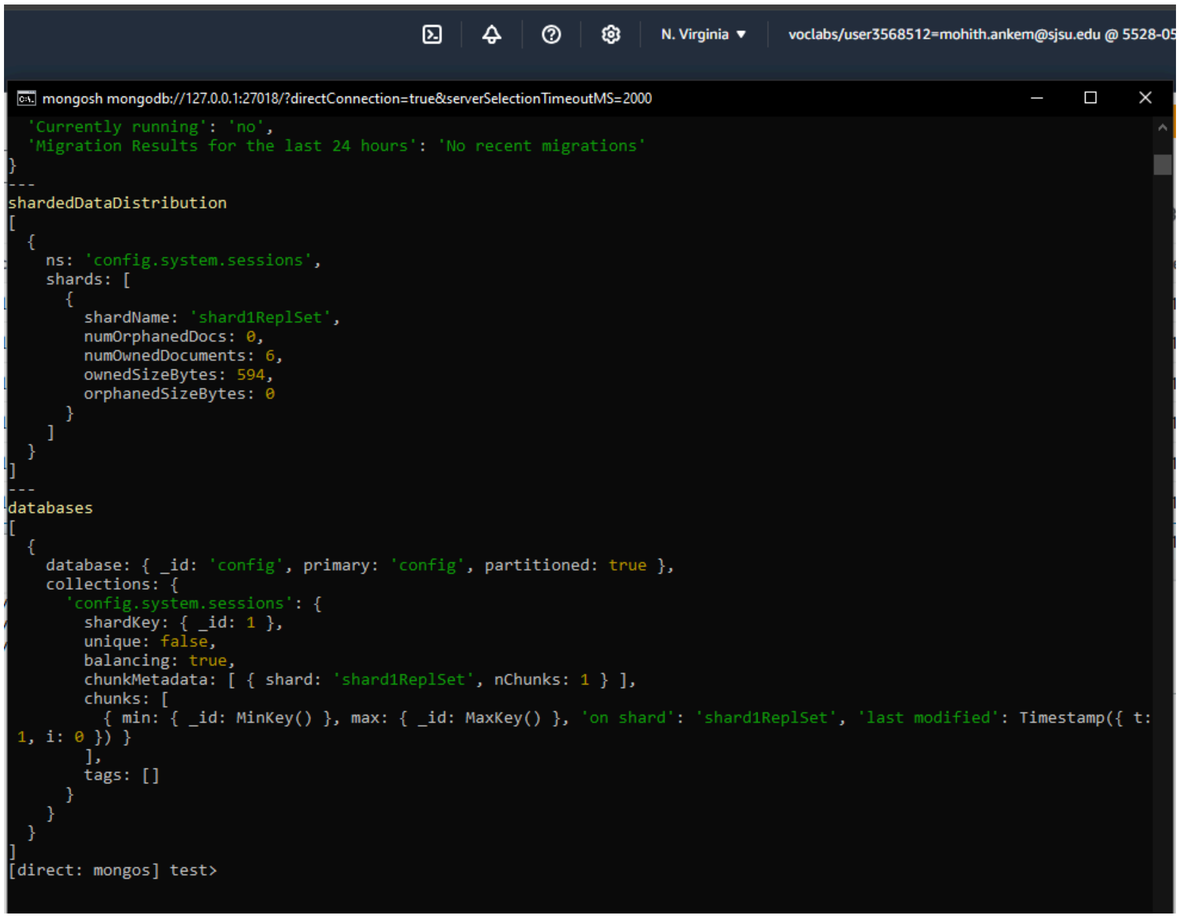
sh.status() after adding the shards:



N. Virginia ▾ voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-055

mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
}
[direct: mongos] test> sh.status()
shardingVersion
{ _id: 1, clusterId: ObjectId('672497c3532a0438eff8985b') }
---
shards
[
  {
    _id: 'shard1ReplSet',
    host: 'shard1ReplSet/172.31.28.155:27020,172.31.28.76:27020,172.31.29.118:27020',
    state: 1,
    topologyTime: Timestamp({ t: 1730455682, i: 11 }),
    replSetConfigVersion: Long('-1')
  },
  {
    _id: 'shard2ReplSet',
    host: 'shard2ReplSet/172.31.28.155:27021,172.31.28.76:27021,172.31.29.118:27021',
    state: 1,
    topologyTime: Timestamp({ t: 1730455698, i: 9 }),
    replSetConfigVersion: Long('-1')
  },
  {
    _id: 'shard3ReplSet',
    host: 'shard3ReplSet/172.31.28.155:27022,172.31.28.76:27022,172.31.29.118:27022',
    state: 1,
    topologyTime: Timestamp({ t: 1730455720, i: 9 }),
    replSetConfigVersion: Long('-1')
  }
]
---
active mongoses
[ { '8.0.3': 1 } ]
---
autosplit
{ 'Currently enabled': 'yes' }
---
balancer
{
  'Currently enabled': 'yes',
  'Failed balancer rounds in last 5 attempts': 0,
  'Currently running': 'no',
  'Migration Results for the last 24 hours': 'No recent migrations'
}
---
shardedDataDistribution
[
  {
    ns: 'config.system.sessions',
    shards: [
      {
        shardName: 'shard1ReplSet',
```



N. Virginia ▾ voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-05

mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
  'Currently running': 'no',
  'Migration Results for the last 24 hours': 'No recent migrations'
}
---
shardedDataDistribution
[
  {
    ns: 'config.system.sessions',
    shards: [
      {
        shardName: 'shard1ReplSet',
        numOrphanedDocs: 0,
        numOwnedDocuments: 6,
        ownedSizeBytes: 594,
        orphanedSizeBytes: 0
      }
    ]
  }
]
---
databases
[
  {
    database: { _id: 'config', primary: 'config', partitioned: true },
    collections: {
      'config.system.sessions': {
        shardKey: { _id: 1 },
        unique: false,
        balancing: true,
        chunkMetadata: [ { shard: 'shard1ReplSet', nChunks: 1 } ],
        chunks: [
          { min: { _id: MinKey() }, max: { _id: MaxKey() }, 'on shard': 'shard1ReplSet', 'last modified': Timestamp({ t:
 1, i: 0 }) }
        ],
        tags: []
      }
    }
  }
]
[direct: mongos] test>
```

10. Enable the shards and explain the nature of the shard key (ascending, random, or location-based) as well as the sharding strategy (range-based or hash-based) used in your deployment.

A. Enabling Shards:

We enabled sharding on the database testdb with the command:

db.runCommand({ enableSharding: "testdb" });

After enabling sharding on testdb, we shard the grading collection with the specified shard key using the command:

sh.shardCollection("testdb.grading", { student_id: "hashed" });

B. Shard Key:

The chosen shard key for the grading collection is **student_id**, with a **hashed sharding strategy.**

1. Nature of Shard Key:

Hashed keys are random in nature. They work by taking the value of the specified field (student_id) and hashing it to distribute documents across shards uniformly.

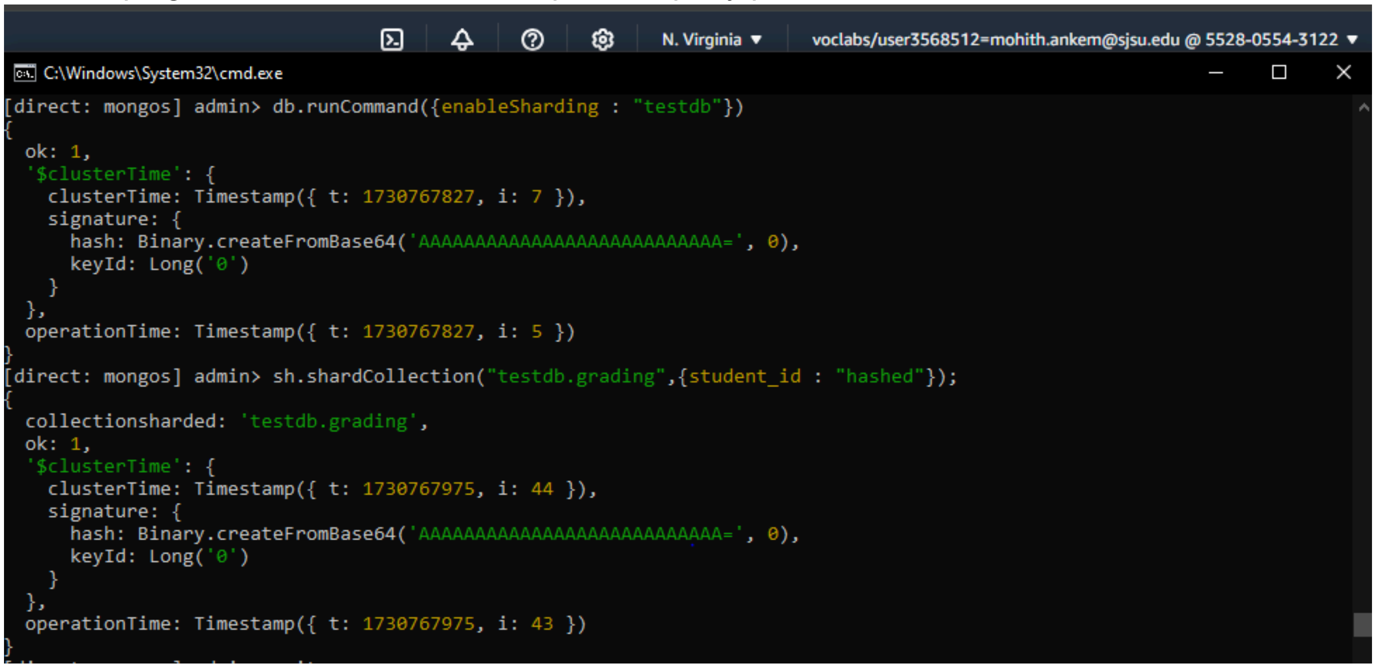
C. Sharding Strategy:

We used a hash-based sharding strategy by specifying **{ "student_id": "hashed" }** as the shard key.

Reason for Choosing Hash-Based Sharding:

In this deployment, hash-based sharding is suitable because it provides consistent and balanced data distribution. Since student_id has a high cardinality (many unique values), hashing each value distributes the documents randomly across the shards.

This approach ensures that each shard receives an approximately equal amount of data, helping to balance the load and optimize query performance.


N. Virginia ▾
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122 ▾
C:\Windows\System32\cmd.exe

```
[direct: mongos] admin> db.runCommand({enableSharding : "testdb"})
{
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1730767827, i: 7 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: Long('0')
    }
  },
  operationTime: Timestamp({ t: 1730767827, i: 5 })
}
[direct: mongos] admin> sh.shardCollection("testdb.grading",{student_id : "hashed"});
{
  collectionsharded: 'testdb.grading',
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1730767975, i: 44 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: Long('0')
    }
  },
  operationTime: Timestamp({ t: 1730767975, i: 43 })
}
```

11. Populate the cluster with data using a public dataset. Explain your collection and include the code to populate the data, along with the result of sh.status() after the data is populated. Specify the URL for the dataset. (Refer to Task 12 to choose an appropriate dataset for executing the given queries. You are allowed to clean and reduce the public dataset of your choice to populate a reasonable amount of data to be distributed across the shards. You may determine what constitutes a reasonable amount.) You are not permitted to use zips.json provided in the prior assignment. If the public dataset you choose is not large enough, you are allowed to synthesize additional data to scale it.

- Dataset:

  The dataset used is based on a modified version of a public dataset, grades.json, which contains student grading information. Each document in the collection represents the scores of a student across different types of assessments (exam, quiz, and homework) for a specific class.

  Dataset URL: The dataset can be found on Kaggle website

  https://www.kaggle.com/datasets/shrashtisinghal/mongo-db-datsets?select=grades.json

  In the given link we can find grades.json in the set of mongoDB Datasets.

- Collection Structure:

  The collection, grading, includes the following fields:

  _id: A unique identifier for each document.

  student_id: An integer representing the student ID.

  scores: An array of subdocuments where each entry includes:

  type: The type of assessment (e.g., "exam", "quiz", "homework").

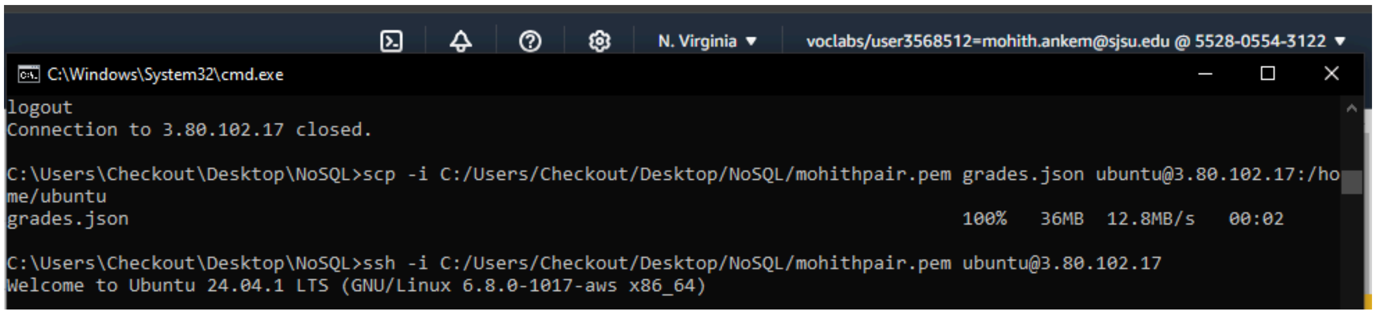
  score: The score received on that assessment.

  class_id: An integer representing the class or course ID.

  This data structure allows us to query based on student_id, class_id, or within the array field scores using operators like $elemMatch and aggregation.

- Shard Key and Sharding Strategy:

  We used student_id as the shard key with a hashed sharding strategy, ensuring a balanced distribution of documents across shards. The hashed strategy helps achieve even distribution, as the student_id values have sufficient cardinality (uniqueness).


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
C:\Windows\System32\cmd.exe

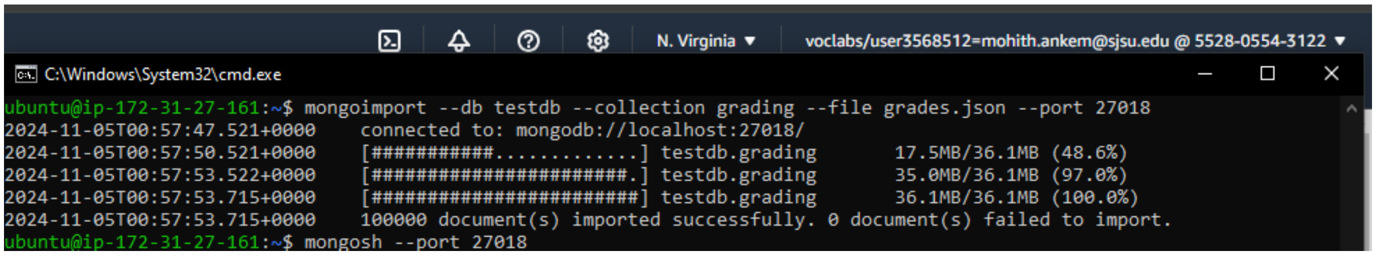
```
logout
Connection to 3.80.102.17 closed.

C:\Users\Checkout\Desktop\NoSQL>scp -i C:/Users/Checkout/Desktop/NoSQL/mohithpair.pem grades.json ubuntu@3.80.102.17:/ho
me/ubuntu
grades.json                                                                          100%   36MB  12.8MB/s   00:02

C:\Users\Checkout\Desktop\NoSQL>ssh -i C:/Users/Checkout/Desktop/NoSQL/mohithpair.pem ubuntu@3.80.102.17
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1017-aws x86_64)
```

Using the

scp -i C:/Users/Desktop/NoSQL/mohithpair.pem grades.json ubuntu@publicIP:/home/ubuntu

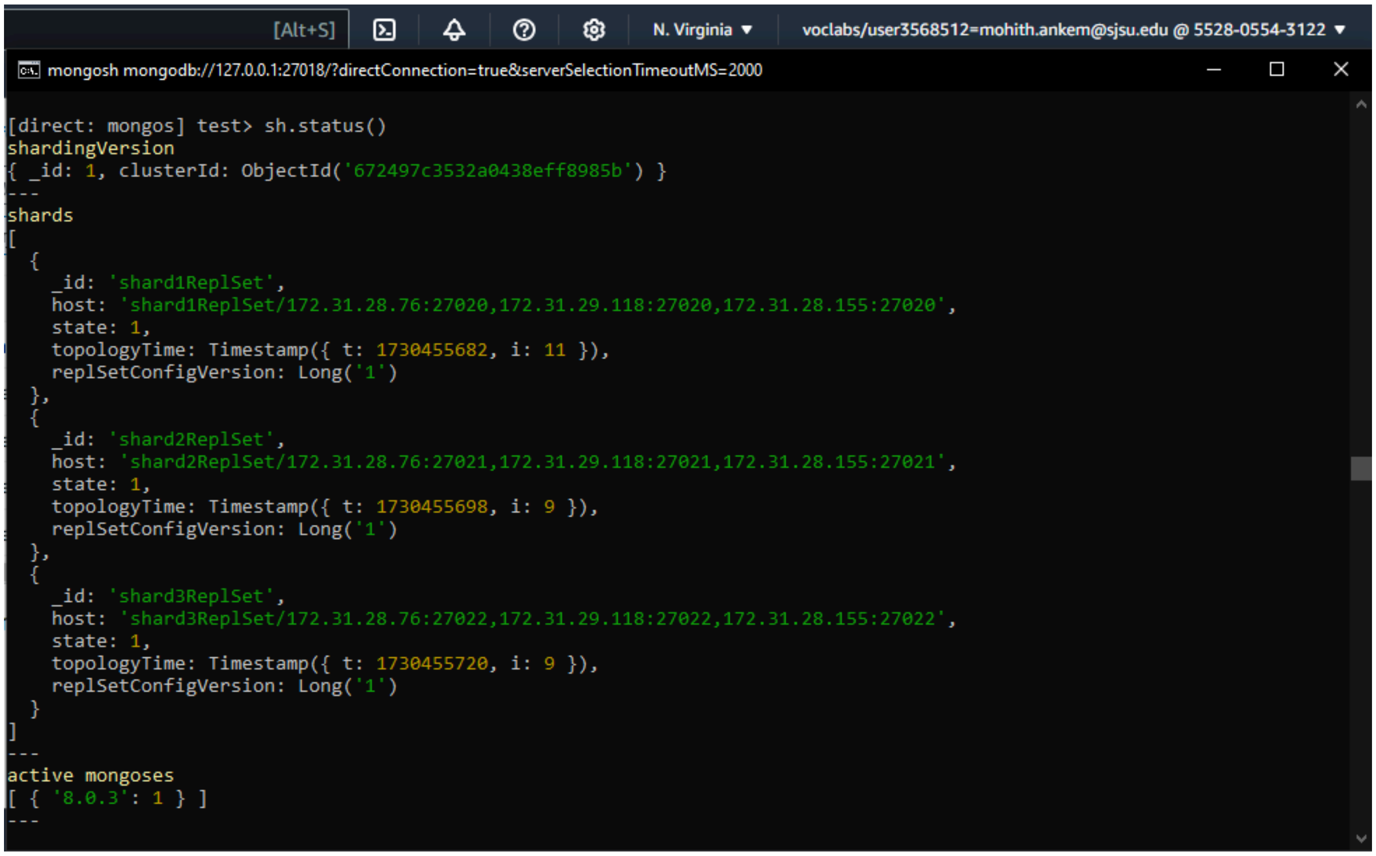
We uploaded the file from the local machine to the server.

Then after enabling the shard database and sharding strategy as explained in the above question we import the dataset into mongos using **mongoimport** command.

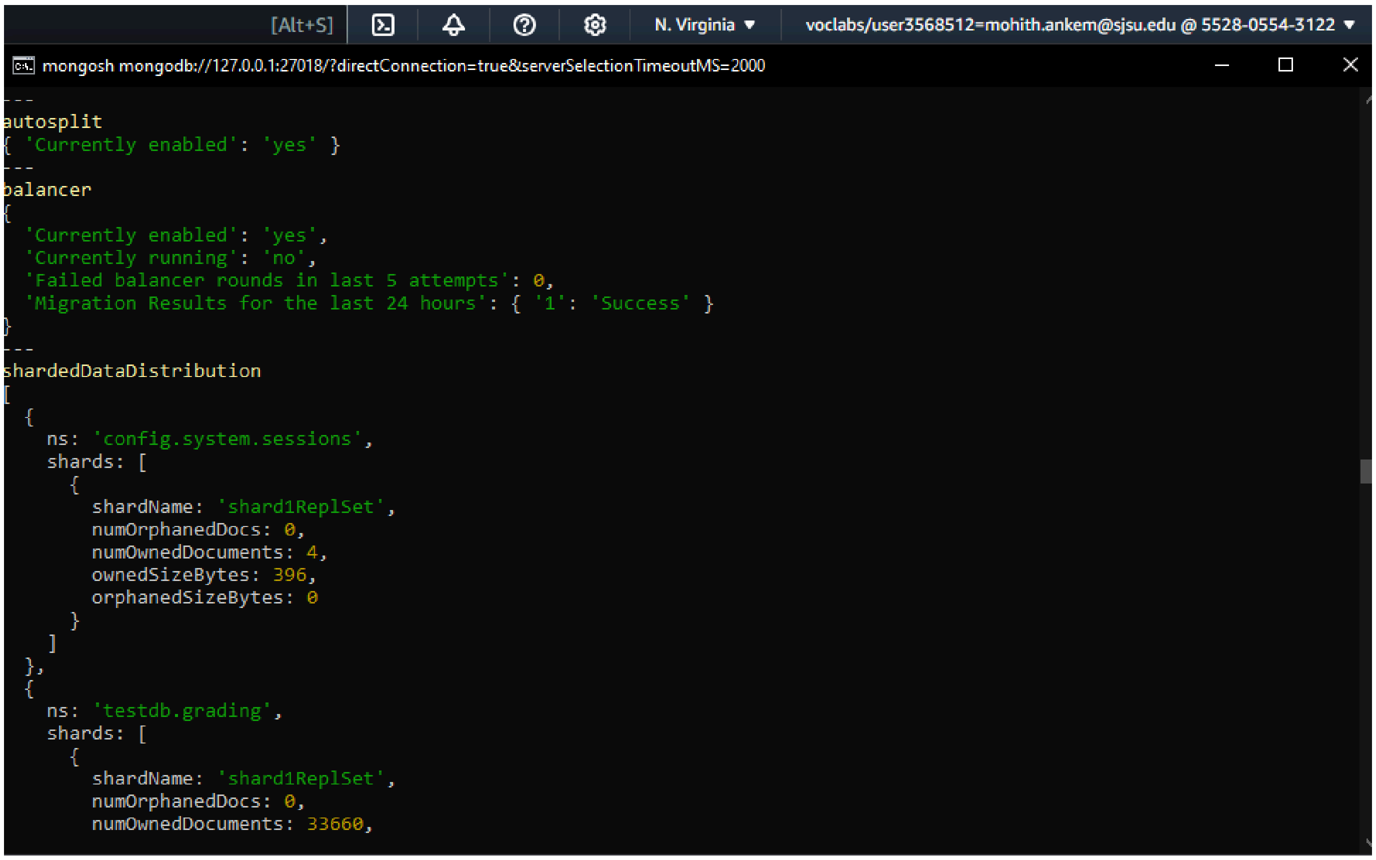

N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
C:\Windows\System32\cmd.exe

```
ubuntu@ip-172-31-27-161:~$ mongoimport --db testdb --collection grading --file grades.json --port 27018
2024-11-05T00:57:47.521+0000    connected to: mongodb://localhost:27018/
2024-11-05T00:57:50.521+0000    [###########.............] testdb.grading       17.5MB/36.1MB (48.6%)
2024-11-05T00:57:53.522+0000    [#######################.] testdb.grading       35.0MB/36.1MB (97.0%)
2024-11-05T00:57:53.715+0000    [########################] testdb.grading       36.1MB/36.1MB (100.0%)
2024-11-05T00:57:53.715+0000    100000 document(s) imported successfully. 0 document(s) failed to import.
ubuntu@ip-172-31-27-161:~$ mongosh --port 27018
```

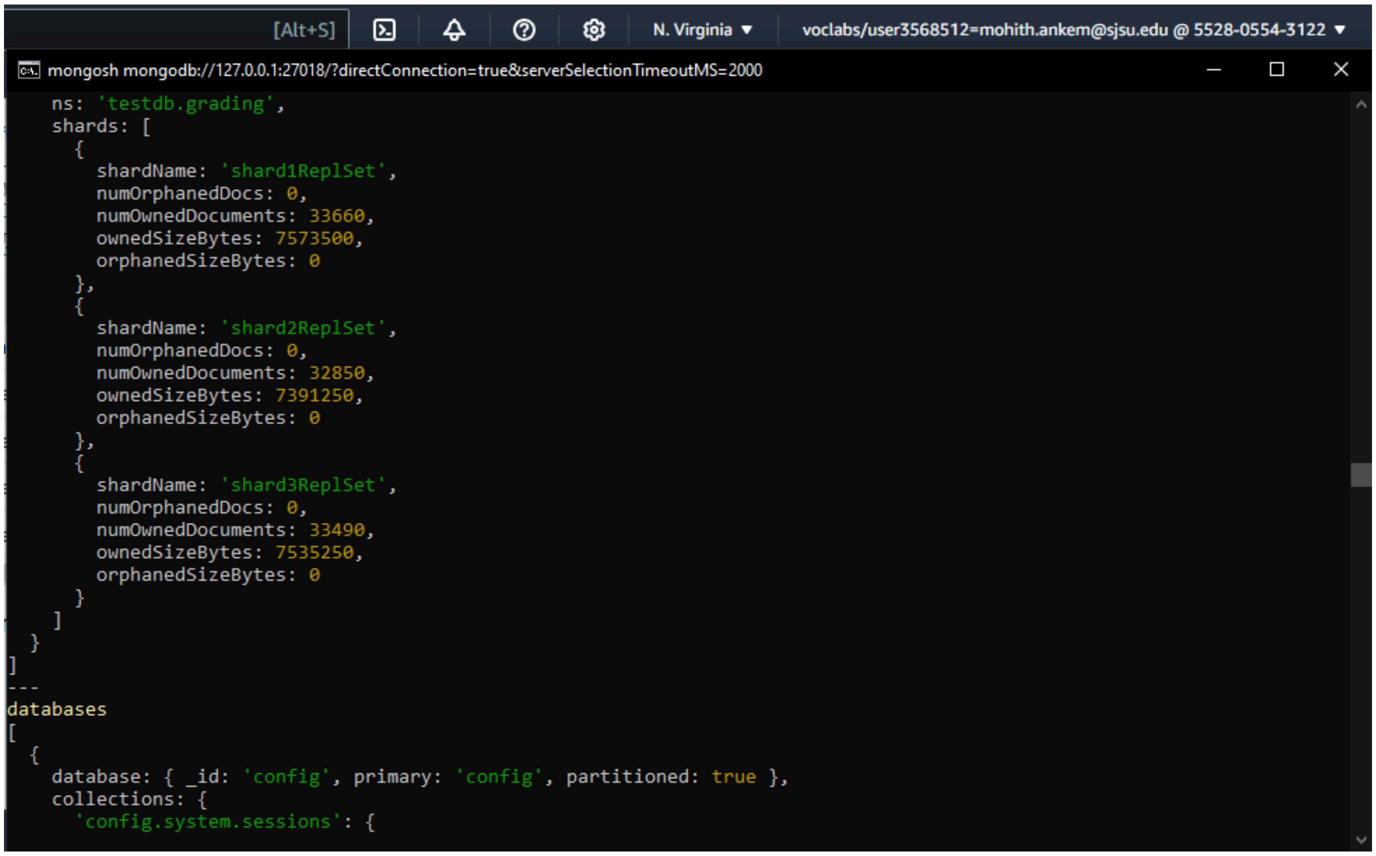
sh.status() after data population:


[Alt+S]
N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

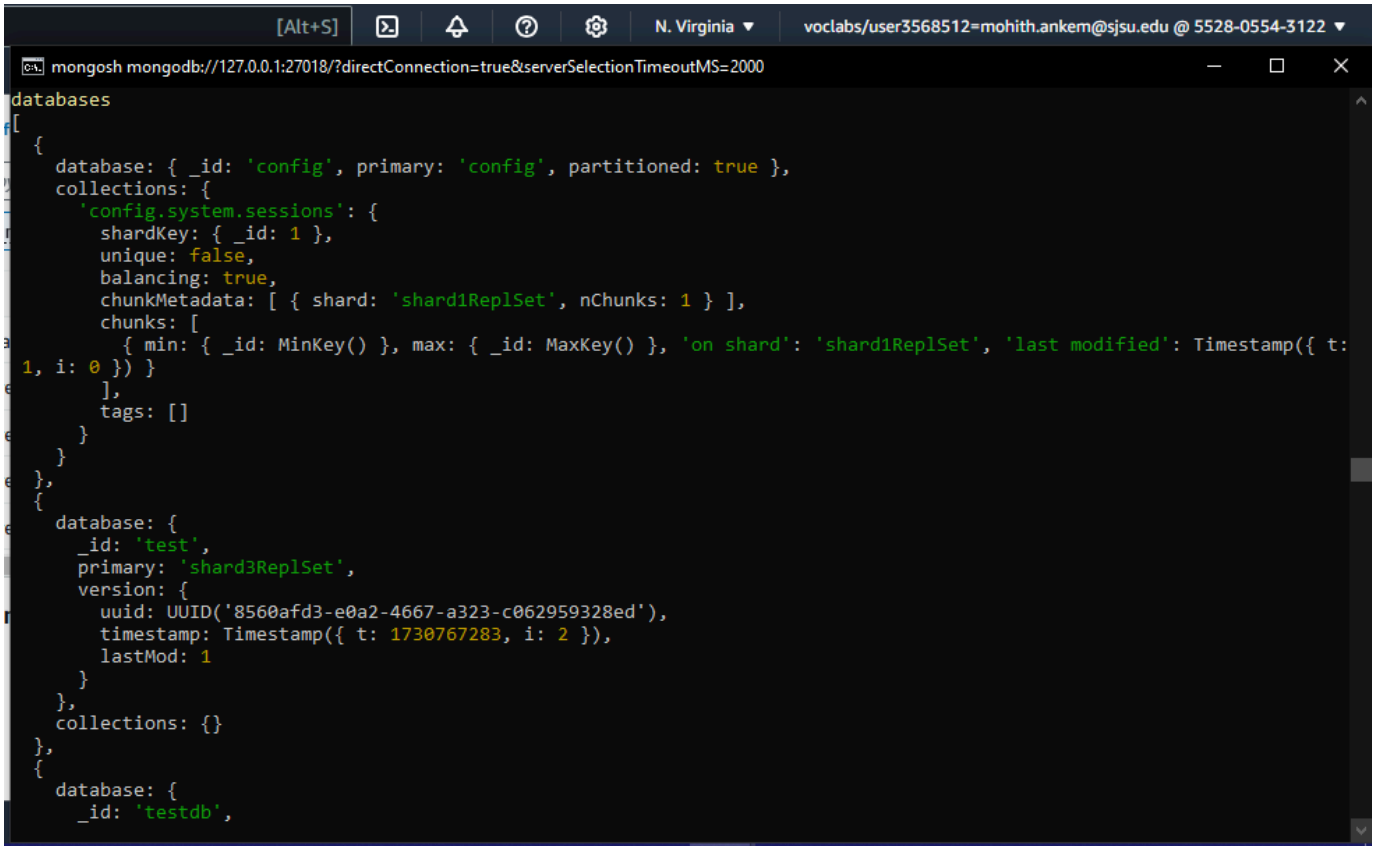
```
[direct: mongos] test> sh.status()
shardingVersion
{ _id: 1, clusterId: ObjectId('672497c3532a0438eff8985b') }
---
shards
[
  {
    _id: 'shard1ReplSet',
    host: 'shard1ReplSet/172.31.28.76:27020,172.31.29.118:27020,172.31.28.155:27020',
    state: 1,
    topologyTime: Timestamp({ t: 1730455682, i: 11 }),
    replSetConfigVersion: Long('1')
  },
  {
    _id: 'shard2ReplSet',
    host: 'shard2ReplSet/172.31.28.76:27021,172.31.29.118:27021,172.31.28.155:27021',
    state: 1,
    topologyTime: Timestamp({ t: 1730455698, i: 9 }),
    replSetConfigVersion: Long('1')
  },
  {
    _id: 'shard3ReplSet',
    host: 'shard3ReplSet/172.31.28.76:27022,172.31.29.118:27022,172.31.28.155:27022',
    state: 1,
    topologyTime: Timestamp({ t: 1730455720, i: 9 }),
    replSetConfigVersion: Long('1')
  }
]
---
active mongoses
[ { '8.0.3': 1 } ]
---
```



```
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000
---
autosplit
{ 'Currently enabled': 'yes' }
---
balancer
{
  'Currently enabled': 'yes',
  'Currently running': 'no',
  'Failed balancer rounds in last 5 attempts': 0,
  'Migration Results for the last 24 hours': { '1': 'Success' }
}
---
shardedDataDistribution
[
  {
    ns: 'config.system.sessions',
    shards: [
      {
        shardName: 'shard1ReplSet',
        numOrphanedDocs: 0,
        numOwnedDocuments: 4,
        ownedSizeBytes: 396,
        orphanedSizeBytes: 0
      }
    ]
  },
  {
    ns: 'testdb.grading',
    shards: [
      {
        shardName: 'shard1ReplSet',
        numOrphanedDocs: 0,
        numOwnedDocuments: 33660,
```

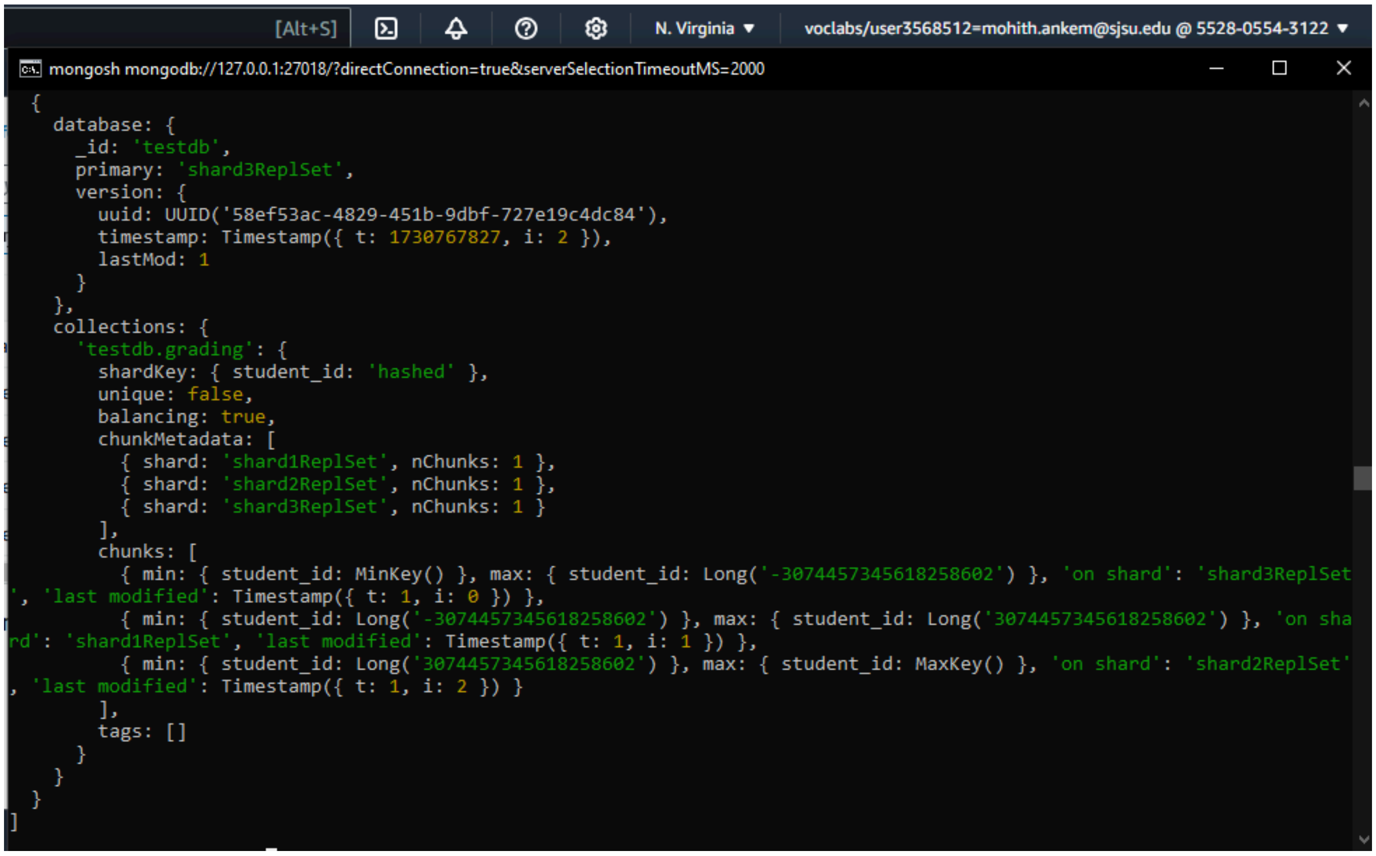


```
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000
    ns: 'testdb.grading',
    shards: [
      {
        shardName: 'shard1ReplSet',
        numOrphanedDocs: 0,
        numOwnedDocuments: 33660,
        ownedSizeBytes: 7573500,
        orphanedSizeBytes: 0
      },
      {
        shardName: 'shard2ReplSet',
        numOrphanedDocs: 0,
        numOwnedDocuments: 32850,
        ownedSizeBytes: 7391250,
        orphanedSizeBytes: 0
      },
      {
        shardName: 'shard3ReplSet',
        numOrphanedDocs: 0,
        numOwnedDocuments: 33490,
        ownedSizeBytes: 7535250,
        orphanedSizeBytes: 0
      }
    ]
  }
]
---
databases
[
  {
    database: { _id: 'config', primary: 'config', partitioned: true },
    collections: {
      'config.system.sessions': {
```



[Alt+S]
N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
databases
[
  {
    database: { _id: 'config', primary: 'config', partitioned: true },
    collections: {
      'config.system.sessions': {
        shardKey: { _id: 1 },
        unique: false,
        balancing: true,
        chunkMetadata: [ { shard: 'shard1ReplSet', nChunks: 1 } ],
        chunks: [
          { min: { _id: MinKey() }, max: { _id: MaxKey() }, 'on shard': 'shard1ReplSet', 'last modified': Timestamp({ t:
 1, i: 0 }) }
        ],
        tags: []
      }
    }
  },
  {
    database: {
      _id: 'test',
      primary: 'shard3ReplSet',
      version: {
        uuid: UUID('8560afd3-e0a2-4667-a323-c062959328ed'),
        timestamp: Timestamp({ t: 1730767283, i: 2 }),
        lastMod: 1
      }
    },
    collections: {}
  },
  {
    database: {
      _id: 'testdb',
```



[Alt+S]
N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
  {
    database: {
      _id: 'testdb',
      primary: 'shard3ReplSet',
      version: {
        uuid: UUID('58ef53ac-4829-451b-9dbf-727e19c4dc84'),
        timestamp: Timestamp({ t: 1730767827, i: 2 }),
        lastMod: 1
      }
    },
    collections: {
      'testdb.grading': {
        shardKey: { student_id: 'hashed' },
        unique: false,
        balancing: true,
        chunkMetadata: [
          { shard: 'shard1ReplSet', nChunks: 1 },
          { shard: 'shard2ReplSet', nChunks: 1 },
          { shard: 'shard3ReplSet', nChunks: 1 }
        ],
        chunks: [
          { min: { student_id: MinKey() }, max: { student_id: Long('-3074457345618258602') }, 'on shard': 'shard3ReplSet
', 'last modified': Timestamp({ t: 1, i: 0 }) },
          { min: { student_id: Long('-3074457345618258602') }, max: { student_id: Long('3074457345618258602') }, 'on sha
rd': 'shard1ReplSet', 'last modified': Timestamp({ t: 1, i: 1 }) },
          { min: { student_id: Long('3074457345618258602') }, max: { student_id: MaxKey() }, 'on shard': 'shard2ReplSet'
, 'last modified': Timestamp({ t: 1, i: 2 }) }
        ],
        tags: []
      }
    }
  }
]
```

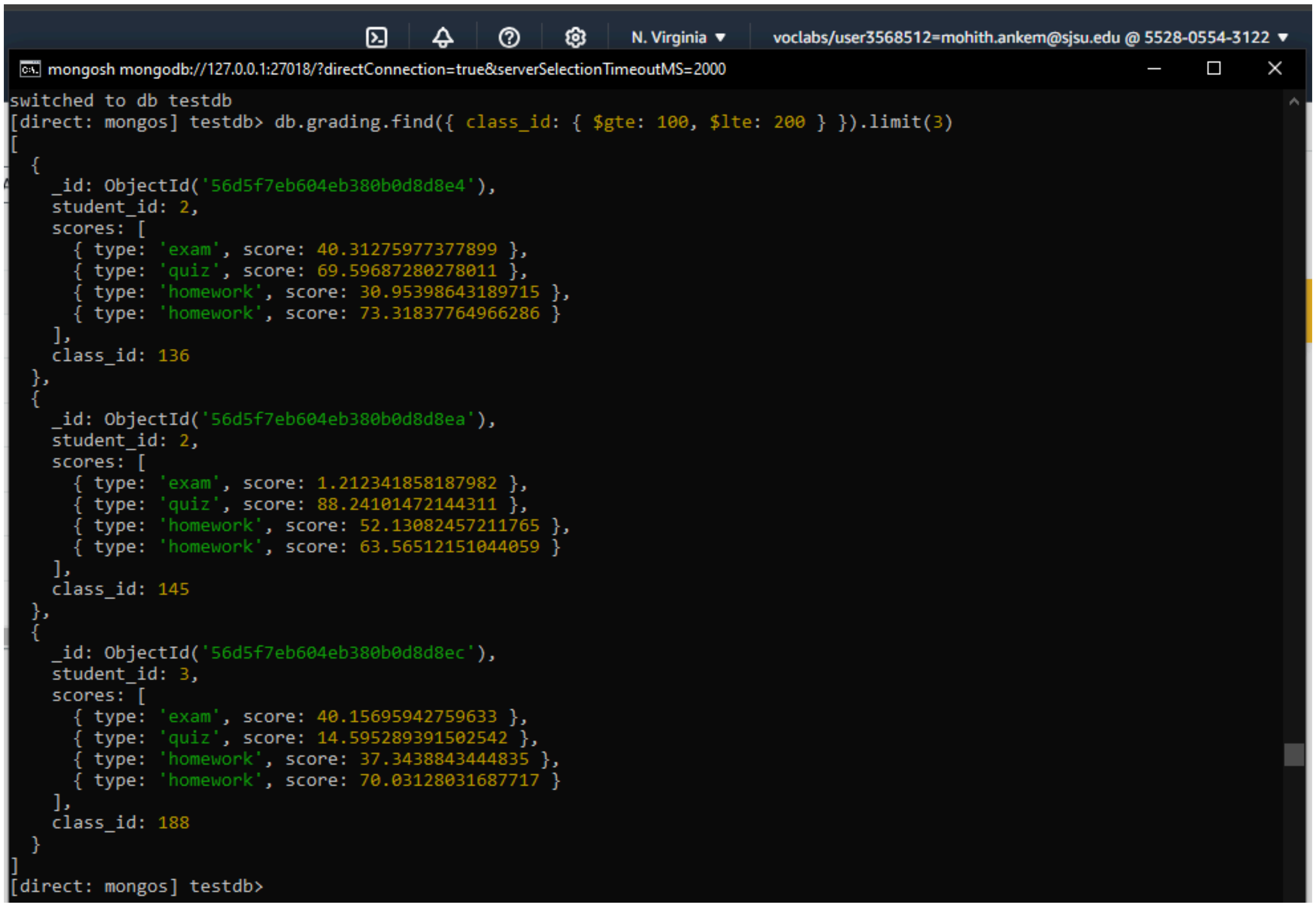
Since we used hashing strategy the documents are split into shard1ReplSet has 33660 documents and 2 chunks, shard2ReplSet has 32850 documents and 1 chunk, whereas shard3ReplSet has 33490 documents and 1 chunks.

12. Generate the following queries for the populated data. For each query, show its execution time and also show which shard served the query.

1. **A range query to find documents in a given range.**

db.grading.find({ student_id: { $gte: 100, $lte: 200 } }).explain("executionStats");

The range query i have used retrieves documents that has student_id in between 100 and 200. Output limited to 3.

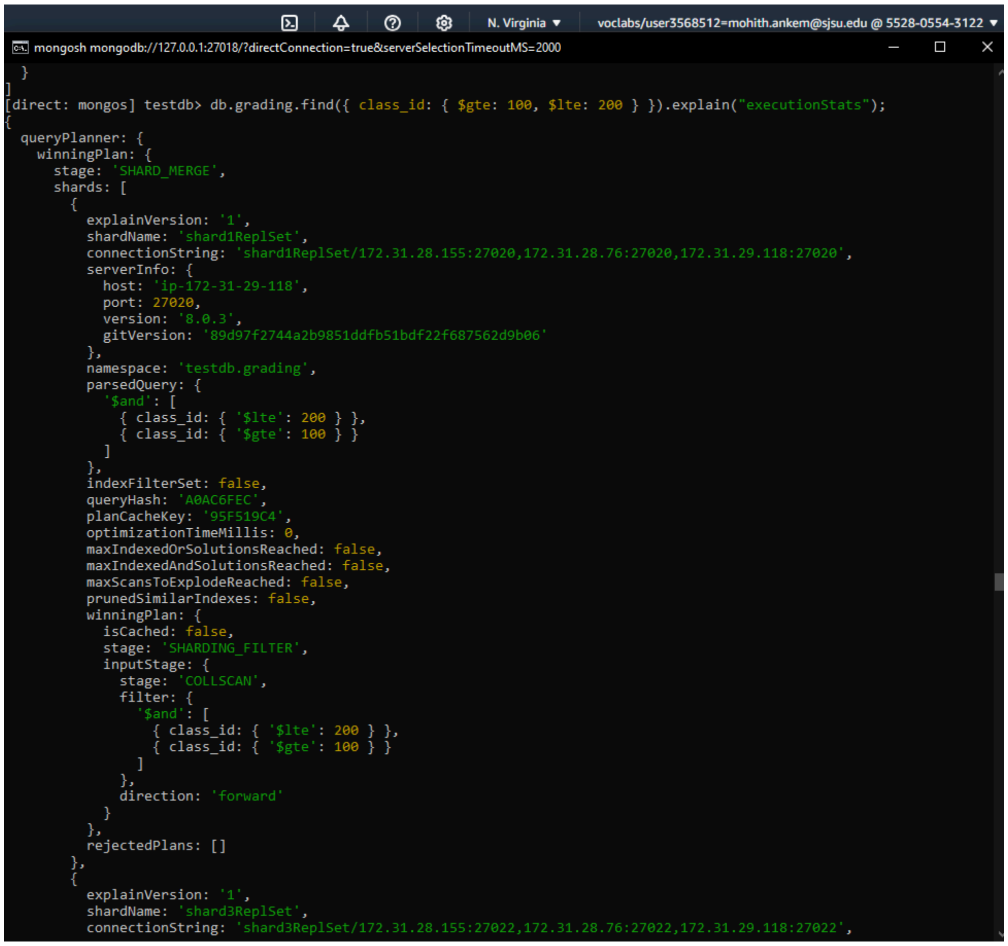

N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
switched to db testdb
[direct: mongos] testdb> db.grading.find({ class_id: { $gte: 100, $lte: 200 } }).limit(3)
[
  {
    _id: ObjectId('56d5f7eb604eb380b0d8d8e4'),
    student_id: 2,
    scores: [
      { type: 'exam', score: 40.31275977377899 },
      { type: 'quiz', score: 69.59687280278011 },
      { type: 'homework', score: 30.95398643189715 },
      { type: 'homework', score: 73.31837764966286 }
    ],
    class_id: 136
  },
  {
    _id: ObjectId('56d5f7eb604eb380b0d8d8ea'),
    student_id: 2,
    scores: [
      { type: 'exam', score: 1.212341858187982 },
      { type: 'quiz', score: 88.24101472144311 },
      { type: 'homework', score: 52.13082457211765 },
      { type: 'homework', score: 63.56512151044059 }
    ],
    class_id: 145
  },
  {
    _id: ObjectId('56d5f7eb604eb380b0d8d8ec'),
    student_id: 3,
    scores: [
      { type: 'exam', score: 40.15695942759633 },
      { type: 'quiz', score: 14.595289391502542 },
      { type: 'homework', score: 37.3438843444835 },
      { type: 'homework', score: 70.03128031687717 }
    ],
    class_id: 188
  }
]
[direct: mongos] testdb>
```

Execution stats after running the query:

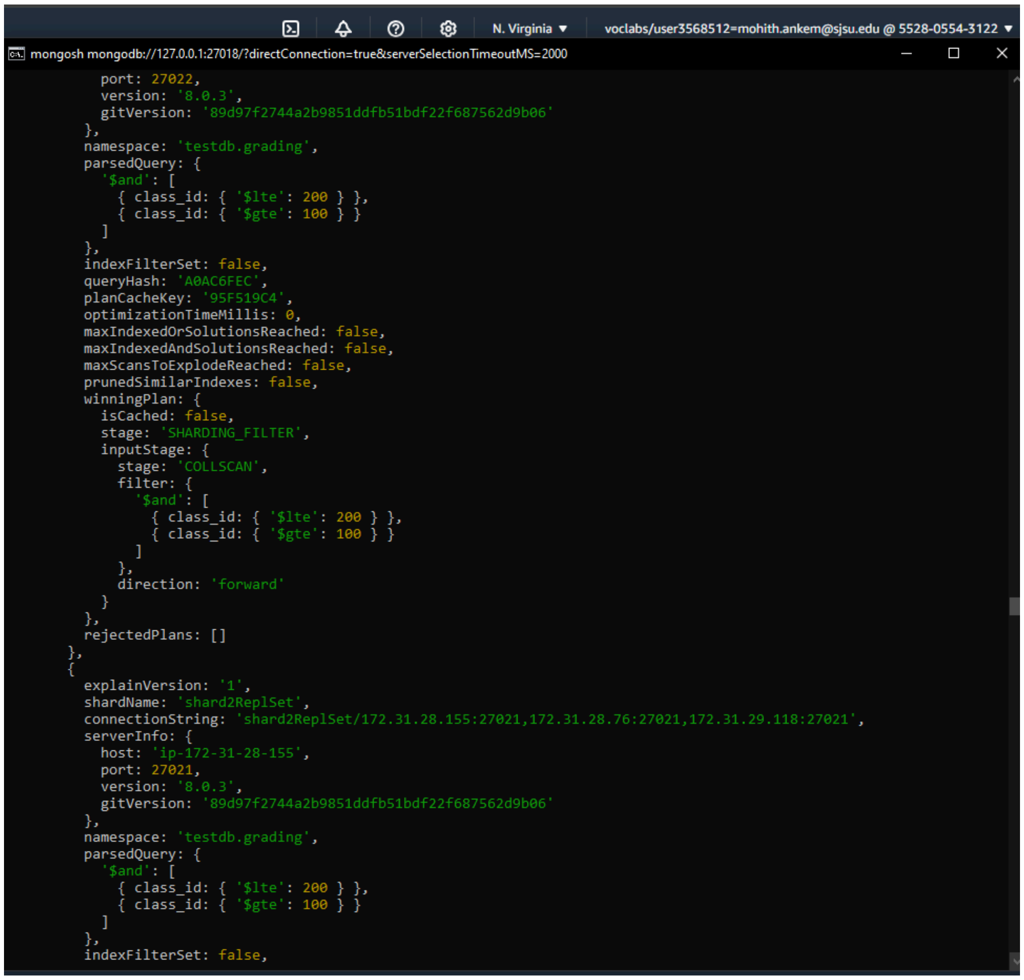
For this Range query since we used hashed strategy all 3 shards served the query.

Shard1 has returned around 6884 docs and shard2 has returned around 6686 docs and shard3 has returned around 6739 docs . The total Execution time it took to run the query is **49ms.**


N. Virginia ▾ voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122 ▾

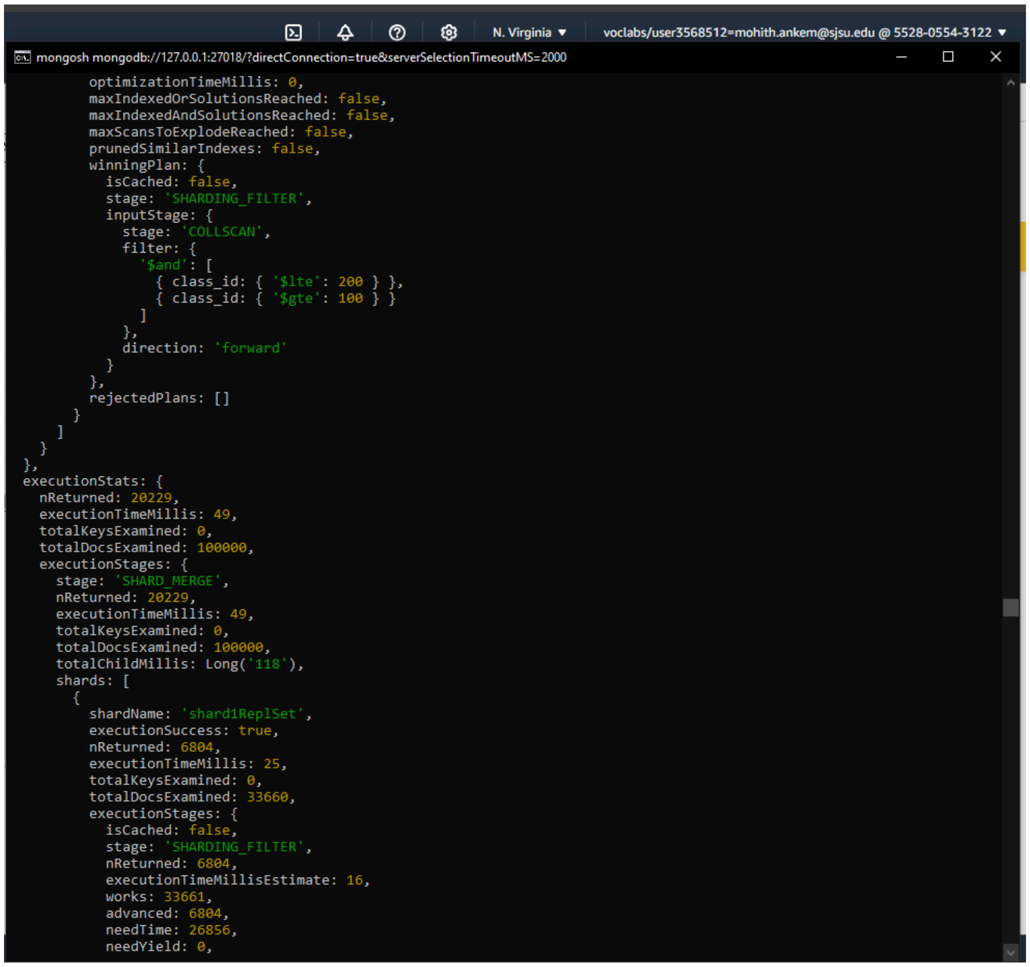
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
  }
]
[direct: mongos] testdb> db.grading.find({ class_id: { $gte: 100, $lte: 200 } }).explain("executionStats");
{
  queryPlanner: {
    winningPlan: {
      stage: 'SHARD_MERGE',
      shards: [
        {
          explainVersion: '1',
          shardName: 'shard1ReplSet',
          connectionString: 'shard1ReplSet/172.31.28.155:27020,172.31.28.76:27020,172.31.29.118:27020',
          serverInfo: {
            host: 'ip-172-31-29-118',
            port: 27020,
            version: '8.0.3',
            gitVersion: '89d97f2744a2b9851ddfb51bdf22f687562d9b06'
          },
          namespace: 'testdb.grading',
          parsedQuery: {
            '$and': [
              { class_id: { '$lte': 200 } },
              { class_id: { '$gte': 100 } }
            ]
          },
          indexFilterSet: false,
          queryHash: 'A0AC6FEC',
          planCacheKey: '95F519C4',
          optimizationTimeMillis: 0,
          maxIndexedOrSolutionsReached: false,
          maxIndexedAndSolutionsReached: false,
          maxScansToExplodeReached: false,
          prunedSimilarIndexes: false,
          winningPlan: {
            isCached: false,
            stage: 'SHARDING_FILTER',
            inputStage: {
              stage: 'COLLSCAN',
              filter: {
                '$and': [
                  { class_id: { '$lte': 200 } },
                  { class_id: { '$gte': 100 } }
                ]
              },
              direction: 'forward'
            }
          },
          rejectedPlans: []
        },
        {
          explainVersion: '1',
          shardName: 'shard3ReplSet',
          connectionString: 'shard3ReplSet/172.31.28.155:27022,172.31.28.76:27022,172.31.29.118:27022',
```


N. Virginia ▾ voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122 ▾

mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

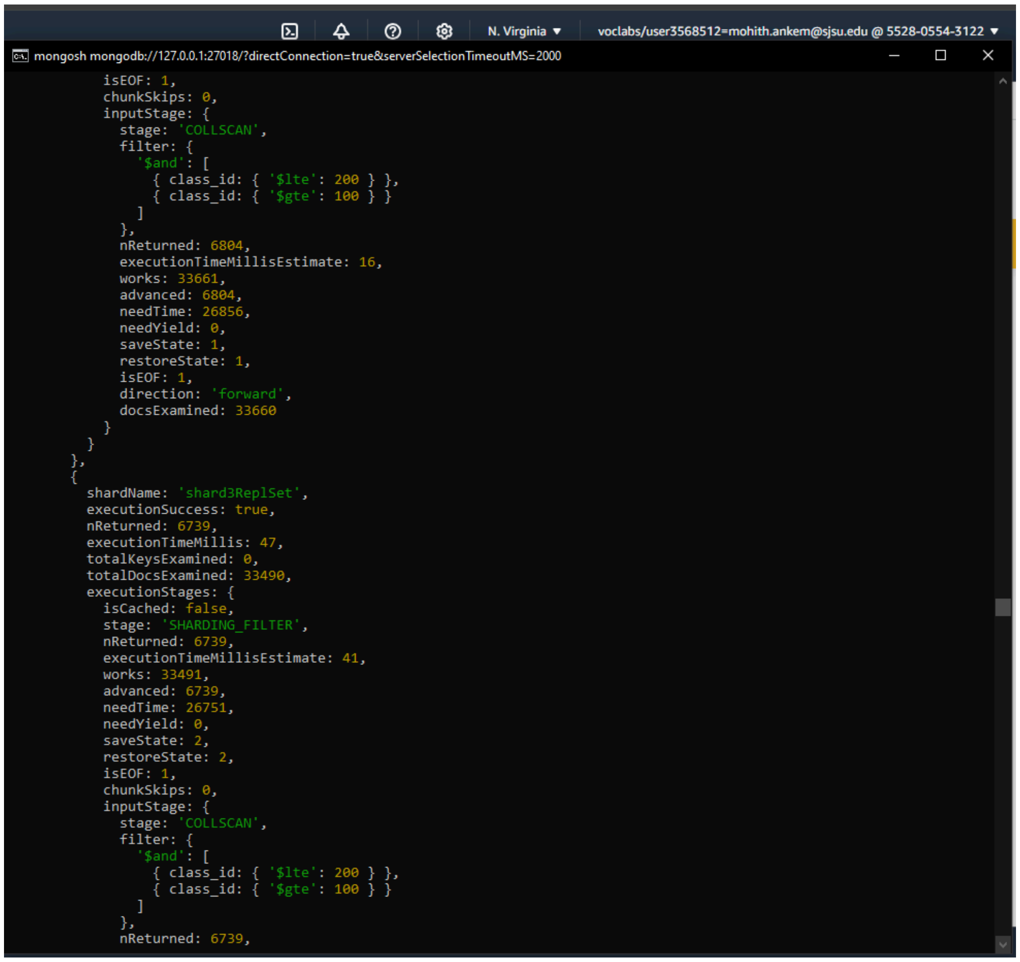
```
            port: 27022,
            version: '8.0.3',
            gitVersion: '89d97f2744a2b9851ddfb51bdf22f687562d9b06'
          },
          namespace: 'testdb.grading',
          parsedQuery: {
            '$and': [
              { class_id: { '$lte': 200 } },
              { class_id: { '$gte': 100 } }
            ]
          },
          indexFilterSet: false,
          queryHash: 'A0AC6FEC',
          planCacheKey: '95F519C4',
          optimizationTimeMillis: 0,
          maxIndexedOrSolutionsReached: false,
          maxIndexedAndSolutionsReached: false,
          maxScansToExplodeReached: false,
          prunedSimilarIndexes: false,
          winningPlan: {
            isCached: false,
            stage: 'SHARDING_FILTER',
            inputStage: {
              stage: 'COLLSCAN',
              filter: {
                '$and': [
                  { class_id: { '$lte': 200 } },
                  { class_id: { '$gte': 100 } }
                ]
              },
              direction: 'forward'
            }
          },
          rejectedPlans: []
        },
        {
          explainVersion: '1',
          shardName: 'shard2ReplSet',
          connectionString: 'shard2ReplSet/172.31.28.155:27021,172.31.28.76:27021,172.31.29.118:27021',
          serverInfo: {
            host: 'ip-172-31-28-155',
            port: 27021,
            version: '8.0.3',
            gitVersion: '89d97f2744a2b9851ddfb51bdf22f687562d9b06'
          },
          namespace: 'testdb.grading',
          parsedQuery: {
            '$and': [
              { class_id: { '$lte': 200 } },
              { class_id: { '$gte': 100 } }
            ]
          },
          indexFilterSet: false,
```



N. Virginia ▾   voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122 ▾

mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

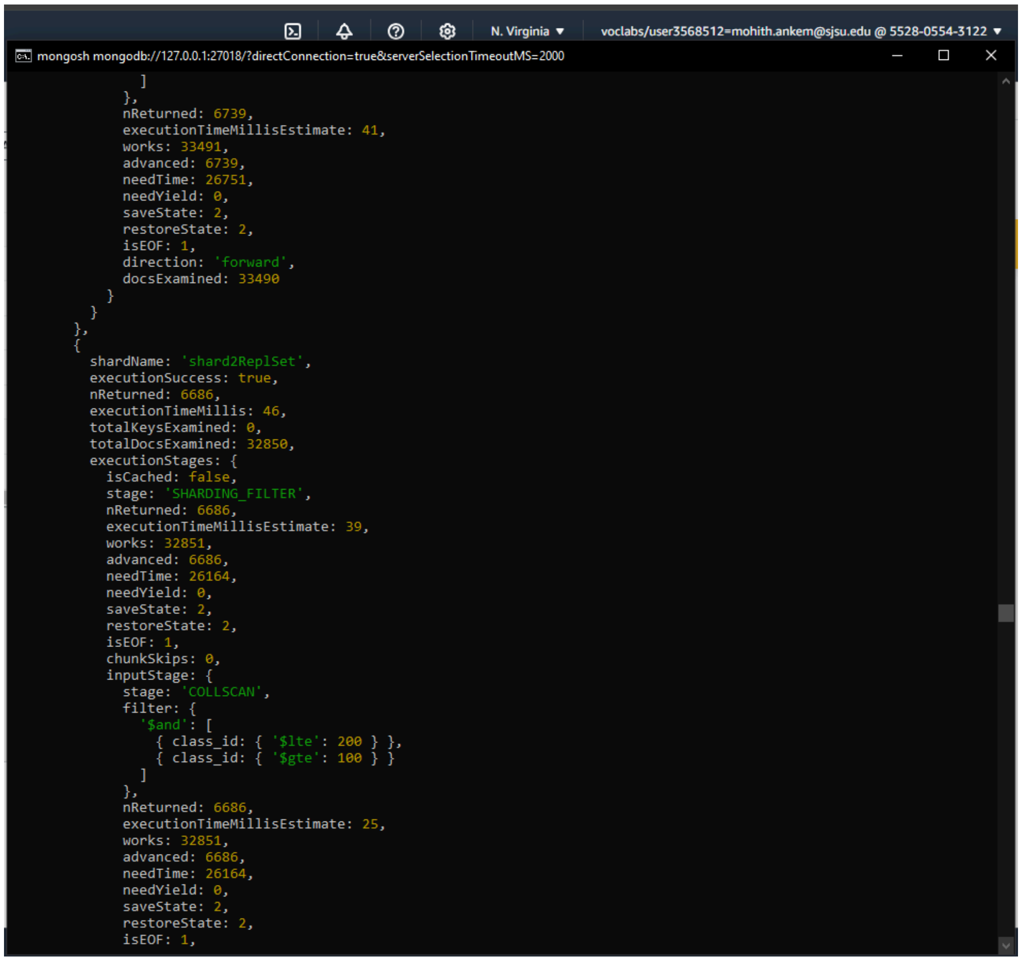
```
        optimizationTimeMillis: 0,
        maxIndexedOrSolutionsReached: false,
        maxIndexedAndSolutionsReached: false,
        maxScansToExplodeReached: false,
        prunedSimilarIndexes: false,
        winningPlan: {
          isCached: false,
          stage: 'SHARDING_FILTER',
          inputStage: {
            stage: 'COLLSCAN',
            filter: {
              '$and': [
                { class_id: { '$lte': 200 } },
                { class_id: { '$gte': 100 } }
              ]
            },
            direction: 'forward'
          }
        },
        rejectedPlans: []
      }
    ]
  }
},
executionStats: {
  nReturned: 20229,
  executionTimeMillis: 49,
  totalKeysExamined: 0,
  totalDocsExamined: 100000,
  executionStages: {
    stage: 'SHARD_MERGE',
    nReturned: 20229,
    executionTimeMillis: 49,
    totalKeysExamined: 0,
    totalDocsExamined: 100000,
    totalChildMillis: Long('118'),
    shards: [
      {
        shardName: 'shard1ReplSet',
        executionSuccess: true,
        nReturned: 6804,
        executionTimeMillis: 25,
        totalKeysExamined: 0,
        totalDocsExamined: 33660,
        executionStages: {
          isCached: false,
          stage: 'SHARDING_FILTER',
          nReturned: 6804,
          executionTimeMillisEstimate: 16,
          works: 33661,
          advanced: 6804,
          needTime: 26856,
          needYield: 0,
```



N. Virginia ▾   voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122 ▾

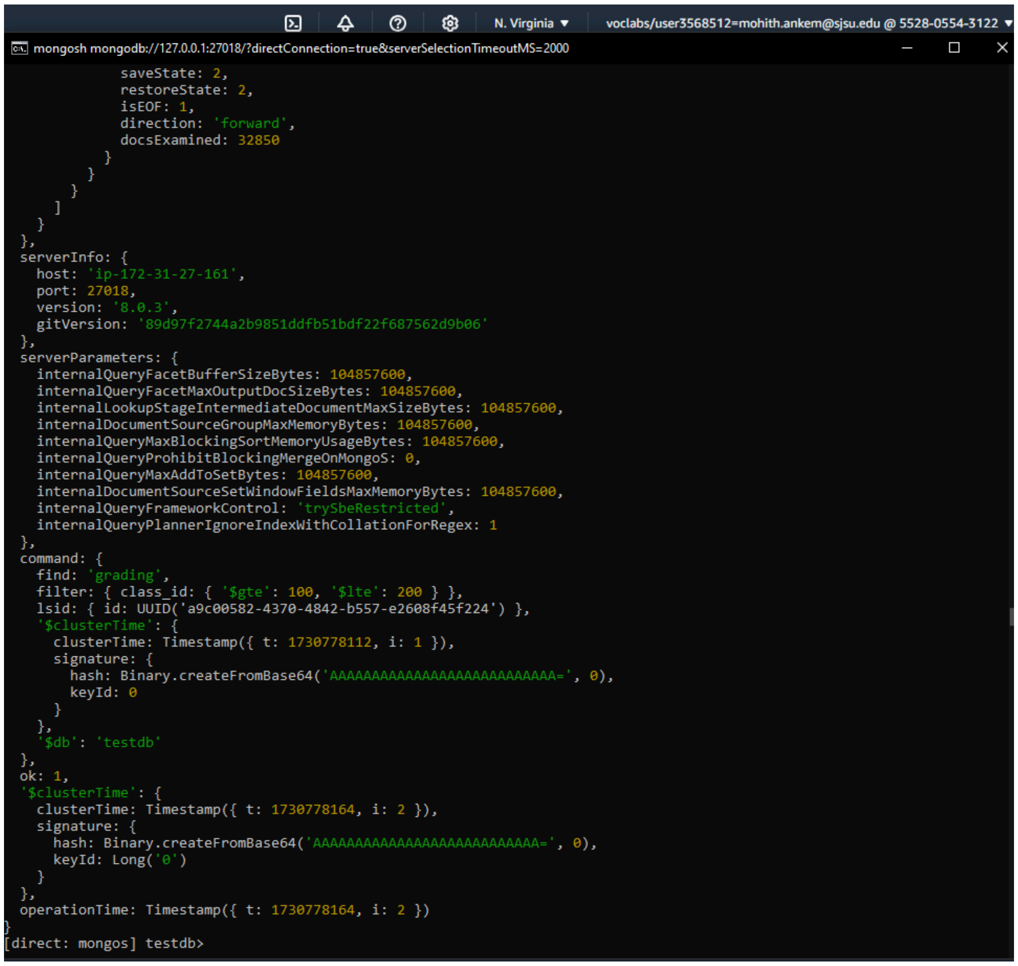
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
          isEOF: 1,
          chunkSkips: 0,
          inputStage: {
            stage: 'COLLSCAN',
            filter: {
              '$and': [
                { class_id: { '$lte': 200 } },
                { class_id: { '$gte': 100 } }
              ]
            },
            nReturned: 6804,
            executionTimeMillisEstimate: 16,
            works: 33661,
            advanced: 6804,
            needTime: 26856,
            needYield: 0,
            saveState: 1,
            restoreState: 1,
            isEOF: 1,
            direction: 'forward',
            docsExamined: 33660
          }
        }
      },
      {
        shardName: 'shard3ReplSet',
        executionSuccess: true,
        nReturned: 6739,
        executionTimeMillis: 47,
        totalKeysExamined: 0,
        totalDocsExamined: 33490,
        executionStages: {
          isCached: false,
          stage: 'SHARDING_FILTER',
          nReturned: 6739,
          executionTimeMillisEstimate: 41,
          works: 33491,
          advanced: 6739,
          needTime: 26751,
          needYield: 0,
          saveState: 2,
          restoreState: 2,
          isEOF: 1,
          chunkSkips: 0,
          inputStage: {
            stage: 'COLLSCAN',
            filter: {
              '$and': [
                { class_id: { '$lte': 200 } },
                { class_id: { '$gte': 100 } }
              ]
            },
            nReturned: 6739,
```



mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
              ]
            },
            nReturned: 6739,
            executionTimeMillisEstimate: 41,
            works: 33491,
            advanced: 6739,
            needTime: 26751,
            needYield: 0,
            saveState: 2,
            restoreState: 2,
            isEOF: 1,
            direction: 'forward',
            docsExamined: 33490
          }
        }
      },
      {
        shardName: 'shard2ReplSet',
        executionSuccess: true,
        nReturned: 6686,
        executionTimeMillis: 46,
        totalKeysExamined: 0,
        totalDocsExamined: 32850,
        executionStages: {
          isCached: false,
          stage: 'SHARDING_FILTER',
          nReturned: 6686,
          executionTimeMillisEstimate: 39,
          works: 32851,
          advanced: 6686,
          needTime: 26164,
          needYield: 0,
          saveState: 2,
          restoreState: 2,
          isEOF: 1,
          chunkSkips: 0,
          inputStage: {
            stage: 'COLLSCAN',
            filter: {
              '$and': [
                { class_id: { '$lte': 200 } },
                { class_id: { '$gte': 100 } }
              ]
            },
            nReturned: 6686,
            executionTimeMillisEstimate: 25,
            works: 32851,
            advanced: 6686,
            needTime: 26164,
            needYield: 0,
            saveState: 2,
            restoreState: 2,
            isEOF: 1,
```



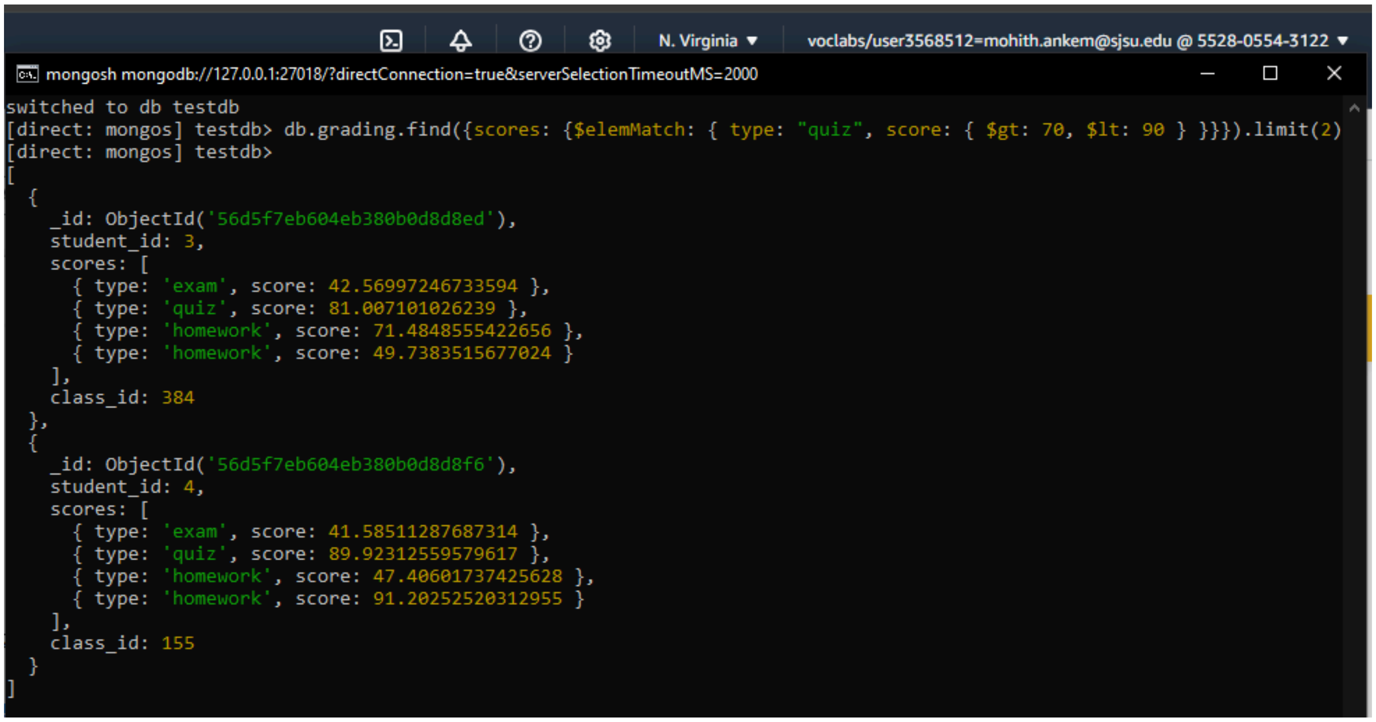
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
            saveState: 2,
            restoreState: 2,
            isEOF: 1,
            direction: 'forward',
            docsExamined: 32850
          }
        }
      }
    ]
  }
},
serverInfo: {
  host: 'ip-172-31-27-161',
  port: 27018,
  version: '8.0.3',
  gitVersion: '89d97f2744a2b9851ddfb51bdf22f687562d9b06'
},
serverParameters: {
  internalQueryFacetBufferSizeBytes: 104857600,
  internalQueryFacetMaxOutputDocSizeBytes: 104857600,
  internalLookupStageIntermediateDocumentMaxSizeBytes: 104857600,
  internalDocumentSourceGroupMaxMemoryBytes: 104857600,
  internalQueryMaxBlockingSortMemoryUsageBytes: 104857600,
  internalQueryProhibitBlockingMergeOnMongoS: 0,
  internalQueryMaxAddToSetBytes: 104857600,
  internalDocumentSourceSetWindowFieldsMaxMemoryBytes: 104857600,
  internalQueryFrameworkControl: 'trySbeRestricted',
  internalQueryPlannerIgnoreIndexWithCollationForRegex: 1
},
command: {
  find: 'grading',
  filter: { class_id: { '$gte': 100, '$lte': 200 } },
  lsid: { id: UUID('a9c00582-4370-4842-b557-e2608f45f224') },
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1730778112, i: 1 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: 0
    }
  },
  '$db': 'testdb'
},
ok: 1,
'$clusterTime': {
  clusterTime: Timestamp({ t: 1730778164, i: 2 }),
  signature: {
    hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
    keyId: Long('0')
  }
},
operationTime: Timestamp({ t: 1730778164, i: 2 })
}
[direct: mongos] testdb>
```

2. A query involving $elemMatch involving at least two conditions.

The $elemMatch condition applies to individual elements within the scores array. Specifically, it filters for documents where there is at least one score entry in the array with:

- type equal to "quiz".
- score greater than 70 and less than 90.

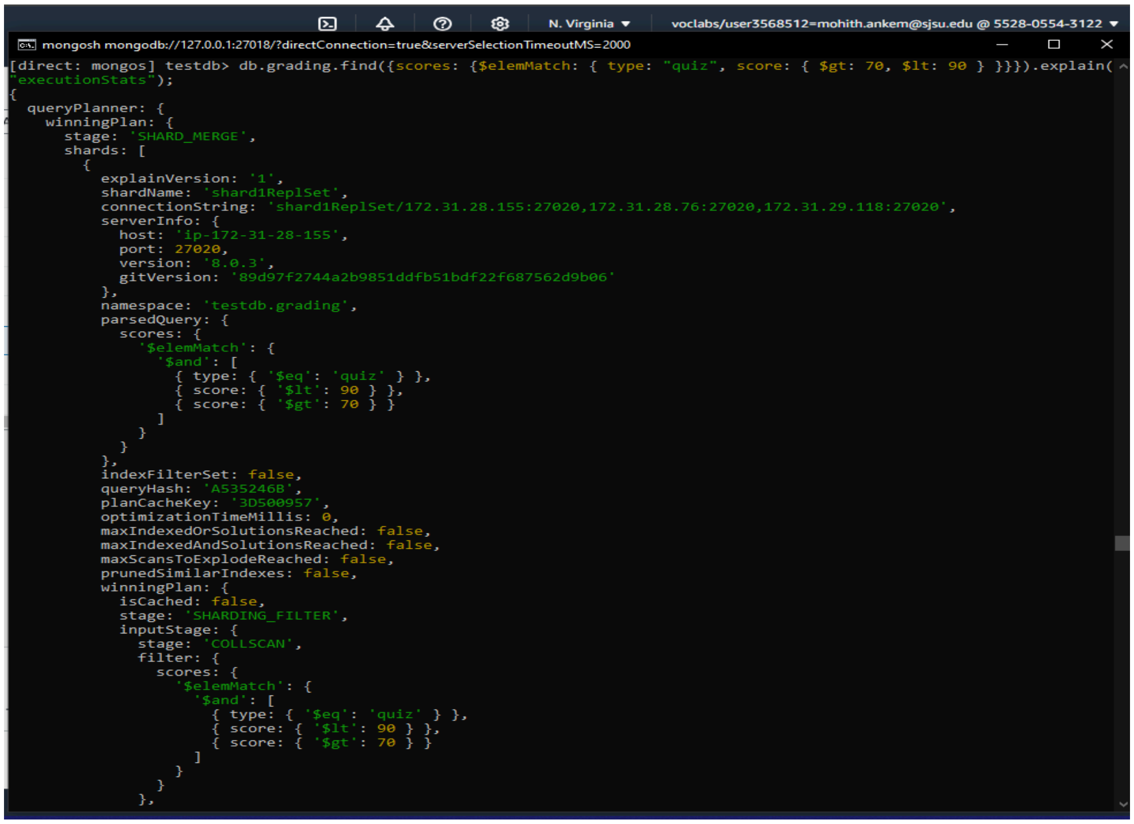


N. Virginia ▼ voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122 ▼

mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
switched to db testdb
[direct: mongos] testdb> db.grading.find({scores: {$elemMatch: { type: "quiz", score: { $gt: 70, $lt: 90 } }}}).limit(2)
[direct: mongos] testdb>
[
  {
    _id: ObjectId('56d5f7eb604eb380b0d8d8ed'),
    student_id: 3,
    scores: [
      { type: 'exam', score: 42.56997246733594 },
      { type: 'quiz', score: 81.007101026239 },
      { type: 'homework', score: 71.4848555422656 },
      { type: 'homework', score: 49.7383515677024 }
    ],
    class_id: 384
  },
  {
    _id: ObjectId('56d5f7eb604eb380b0d8d8f6'),
    student_id: 4,
    scores: [
      { type: 'exam', score: 41.58511287687314 },
      { type: 'quiz', score: 89.92312559579617 },
      { type: 'homework', score: 47.40601737425628 },
      { type: 'homework', score: 91.20252520312955 }
    ],
    class_id: 155
  }
]
```

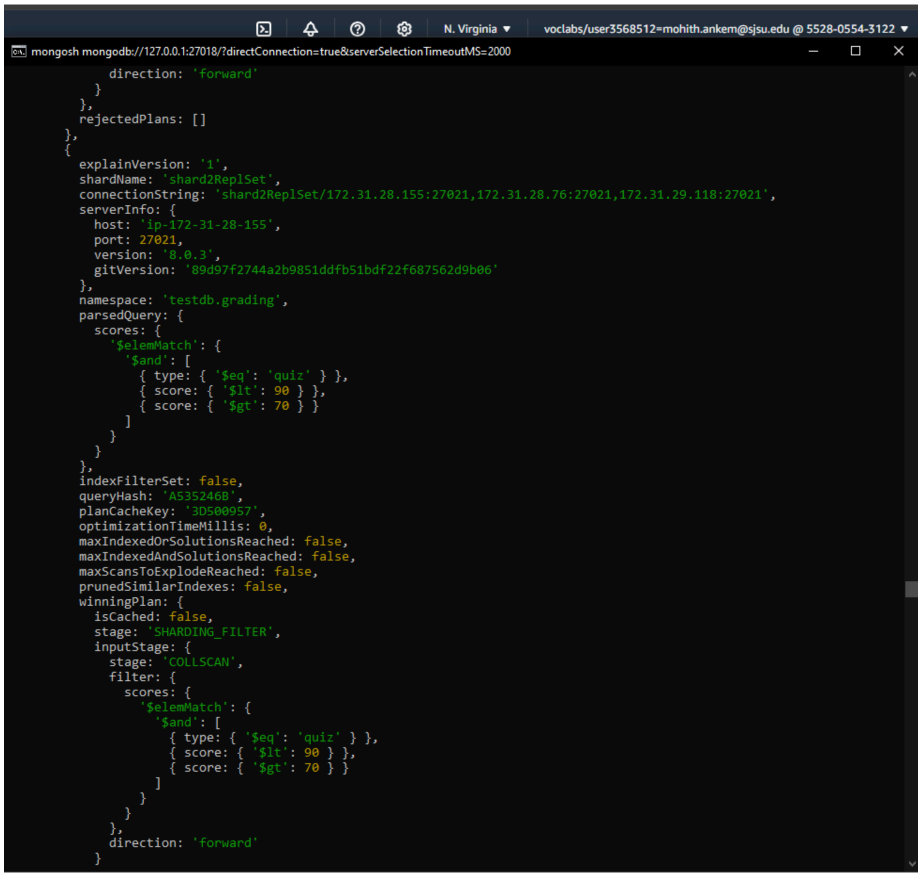
Execution stats:



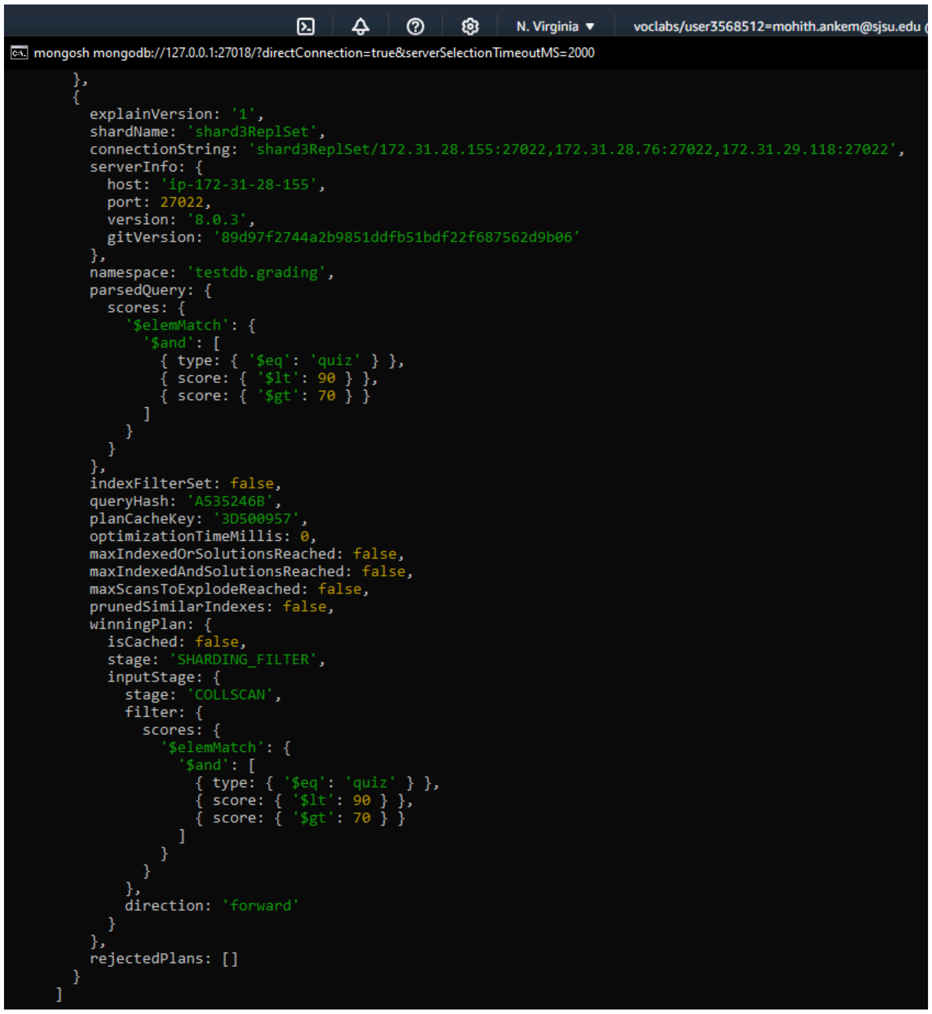
N. Virginia ▼ voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122 ▼

mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

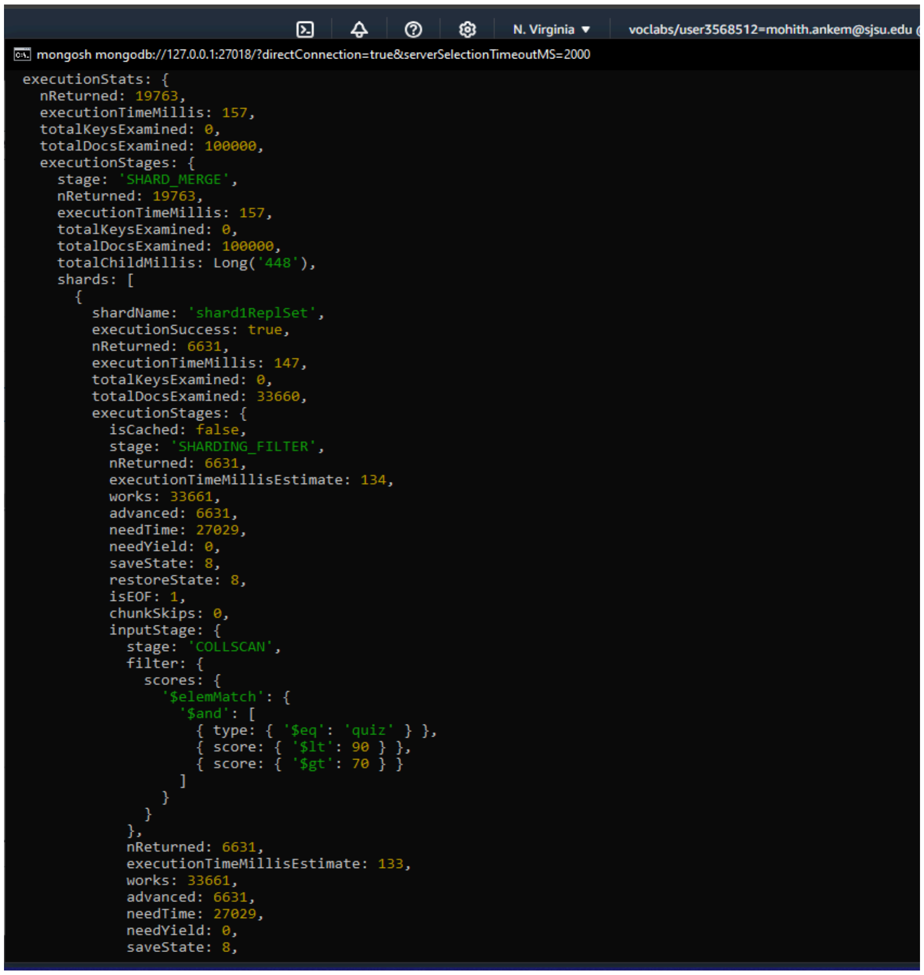
```
[direct: mongos] testdb> db.grading.find({scores: {$elemMatch: { type: "quiz", score: { $gt: 70, $lt: 90 } }}}).explain(
"executionStats");
{
  queryPlanner: {
    winningPlan: {
      stage: 'SHARD_MERGE',
      shards: [
        {
          explainVersion: '1',
          shardName: 'shard1ReplSet',
          connectionString: 'shard1ReplSet/172.31.28.155:27020,172.31.28.76:27020,172.31.29.118:27020',
          serverInfo: {
            host: 'ip-172-31-28-155',
            port: 27020,
            version: '8.0.3',
            gitVersion: '89d97f2744a2b9851ddfb51bdf22f687562d9b06'
          },
          namespace: 'testdb.grading',
          parsedQuery: {
            scores: {
              '$elemMatch': {
                '$and': [
                  { type: { '$eq': 'quiz' } },
                  { score: { '$lt': 90 } },
                  { score: { '$gt': 70 } }
                ]
              }
            }
          },
          indexFilterSet: false,
          queryHash: 'A535246B',
          planCacheKey: '3D500957',
          optimizationTimeMillis: 0,
          maxIndexedOrSolutionsReached: false,
          maxIndexedAndSolutionsReached: false,
          maxScansToExplodeReached: false,
          prunedSimilarIndexes: false,
          winningPlan: {
            isCached: false,
            stage: 'SHARDING_FILTER',
            inputStage: {
              stage: 'COLLSCAN',
              filter: {
                scores: {
                  '$elemMatch': {
                    '$and': [
                      { type: { '$eq': 'quiz' } },
                      { score: { '$lt': 90 } },
                      { score: { '$gt': 70 } }
                    ]
                  }
                }
              },
```



```
            direction: 'forward'
          }
        },
        rejectedPlans: []
      },
      {
        explainVersion: '1',
        shardName: 'shard2ReplSet',
        connectionString: 'shard2ReplSet/172.31.28.155:27021,172.31.28.76:27021,172.31.29.118:27021',
        serverInfo: {
          host: 'ip-172-31-28-155',
          port: 27021,
          version: '8.0.3',
          gitVersion: '89d97f2744a2b9851ddfb51bdf22f687562d9b06'
        },
        namespace: 'testdb.grading',
        parsedQuery: {
          scores: {
            '$elemMatch': {
              '$and': [
                { type: { '$eq': 'quiz' } },
                { score: { '$lt': 90 } },
                { score: { '$gt': 70 } }
              ]
            }
          }
        },
        indexFilterSet: false,
        queryHash: 'A535246B',
        planCacheKey: '3D500957',
        optimizationTimeMillis: 0,
        maxIndexedOrSolutionsReached: false,
        maxIndexedAndSolutionsReached: false,
        maxScansToExplodeReached: false,
        prunedSimilarIndexes: false,
        winningPlan: {
          isCached: false,
          stage: 'SHARDING_FILTER',
          inputStage: {
            stage: 'COLLSCAN',
            filter: {
              scores: {
                '$elemMatch': {
                  '$and': [
                    { type: { '$eq': 'quiz' } },
                    { score: { '$lt': 90 } },
                    { score: { '$gt': 70 } }
                  ]
                }
              }
            },
            direction: 'forward'
          }
```

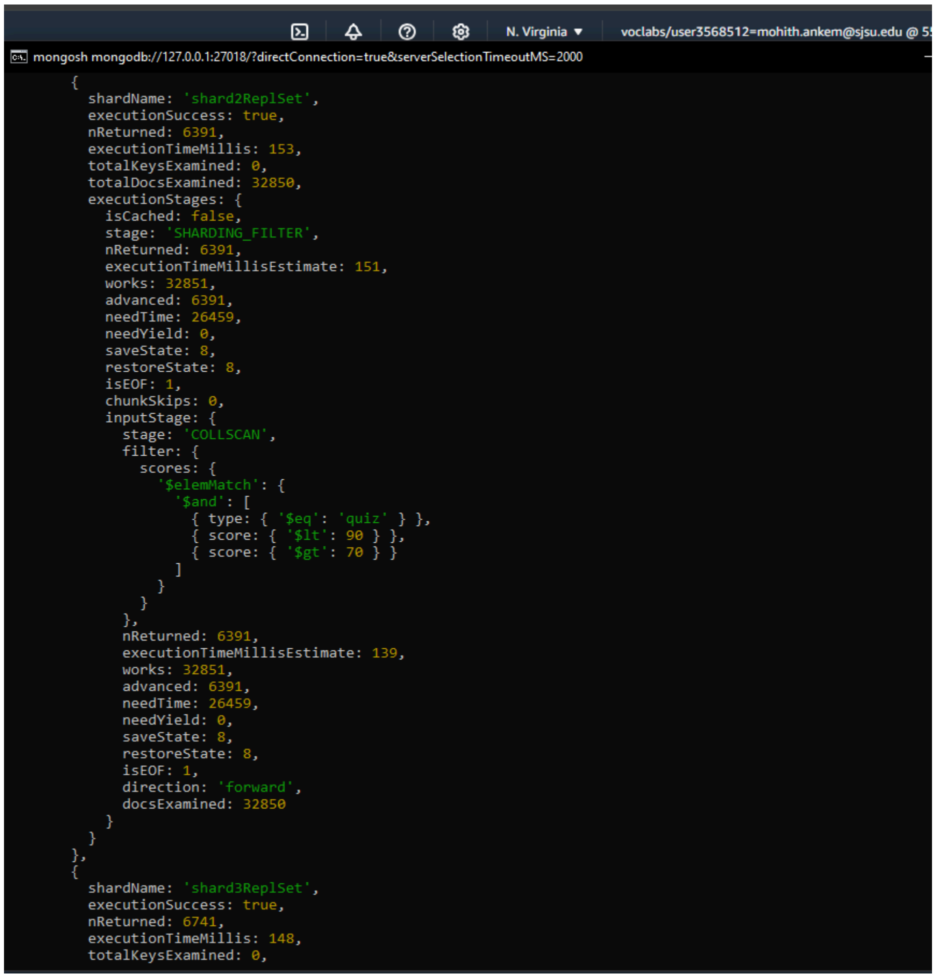


```
      },
      {
        explainVersion: '1',
        shardName: 'shard3ReplSet',
        connectionString: 'shard3ReplSet/172.31.28.155:27022,172.31.28.76:27022,172.31.29.118:27022',
        serverInfo: {
          host: 'ip-172-31-28-155',
          port: 27022,
          version: '8.0.3',
          gitVersion: '89d97f2744a2b9851ddfb51bdf22f687562d9b06'
        },
        namespace: 'testdb.grading',
        parsedQuery: {
          scores: {
            '$elemMatch': {
              '$and': [
                { type: { '$eq': 'quiz' } },
                { score: { '$lt': 90 } },
                { score: { '$gt': 70 } }
              ]
            }
          }
        },
        indexFilterSet: false,
        queryHash: 'A535246B',
        planCacheKey: '3D500957',
        optimizationTimeMillis: 0,
        maxIndexedOrSolutionsReached: false,
        maxIndexedAndSolutionsReached: false,
        maxScansToExplodeReached: false,
        prunedSimilarIndexes: false,
        winningPlan: {
          isCached: false,
          stage: 'SHARDING_FILTER',
          inputStage: {
            stage: 'COLLSCAN',
            filter: {
              scores: {
                '$elemMatch': {
                  '$and': [
                    { type: { '$eq': 'quiz' } },
                    { score: { '$lt': 90 } },
                    { score: { '$gt': 70 } }
                  ]
                }
              }
            },
            direction: 'forward'
          }
        },
        rejectedPlans: []
      }
    ]
```



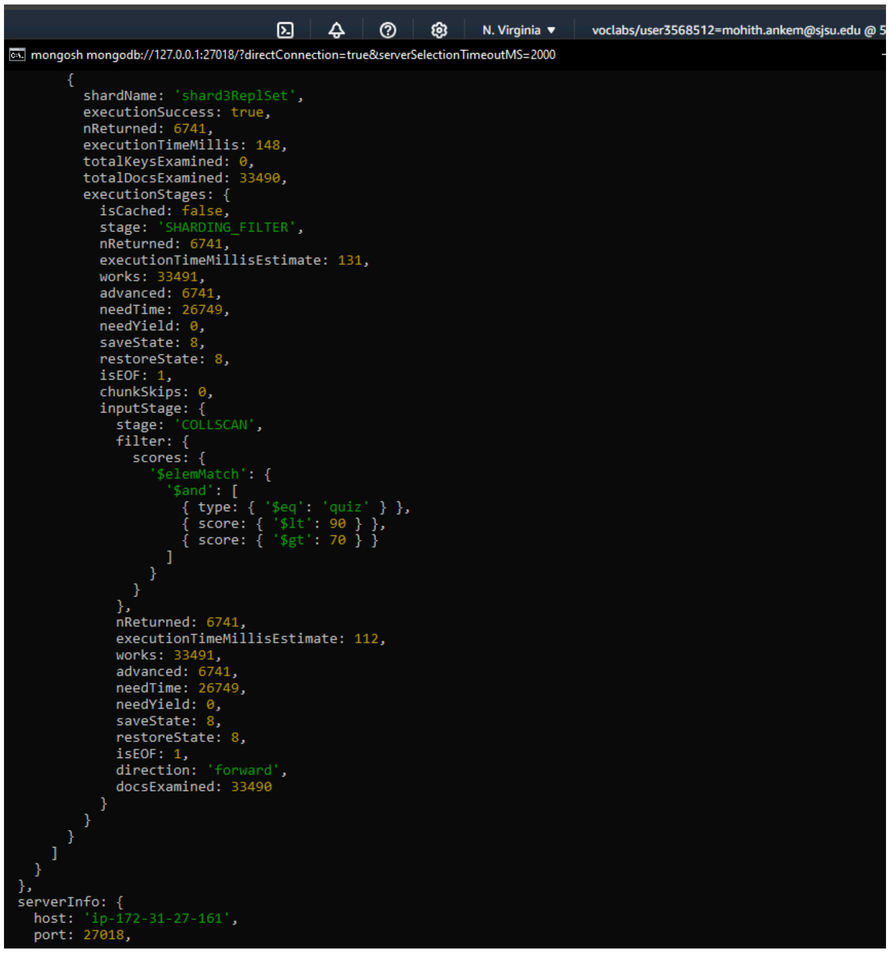
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
  executionStats: {
    nReturned: 19763,
    executionTimeMillis: 157,
    totalKeysExamined: 0,
    totalDocsExamined: 100000,
    executionStages: {
      stage: 'SHARD_MERGE',
      nReturned: 19763,
      executionTimeMillis: 157,
      totalKeysExamined: 0,
      totalDocsExamined: 100000,
      totalChildMillis: Long('448'),
      shards: [
        {
          shardName: 'shard1ReplSet',
          executionSuccess: true,
          nReturned: 6631,
          executionTimeMillis: 147,
          totalKeysExamined: 0,
          totalDocsExamined: 33660,
          executionStages: {
            isCached: false,
            stage: 'SHARDING_FILTER',
            nReturned: 6631,
            executionTimeMillisEstimate: 134,
            works: 33661,
            advanced: 6631,
            needTime: 27029,
            needYield: 0,
            saveState: 8,
            restoreState: 8,
            isEOF: 1,
            chunkSkips: 0,
            inputStage: {
              stage: 'COLLSCAN',
              filter: {
                scores: {
                  '$elemMatch': {
                    '$and': [
                      { type: { '$eq': 'quiz' } },
                      { score: { '$lt': 90 } },
                      { score: { '$gt': 70 } }
                    ]
                  }
                }
              },
              nReturned: 6631,
              executionTimeMillisEstimate: 133,
              works: 33661,
              advanced: 6631,
              needTime: 27029,
              needYield: 0,
              saveState: 8,
```



mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

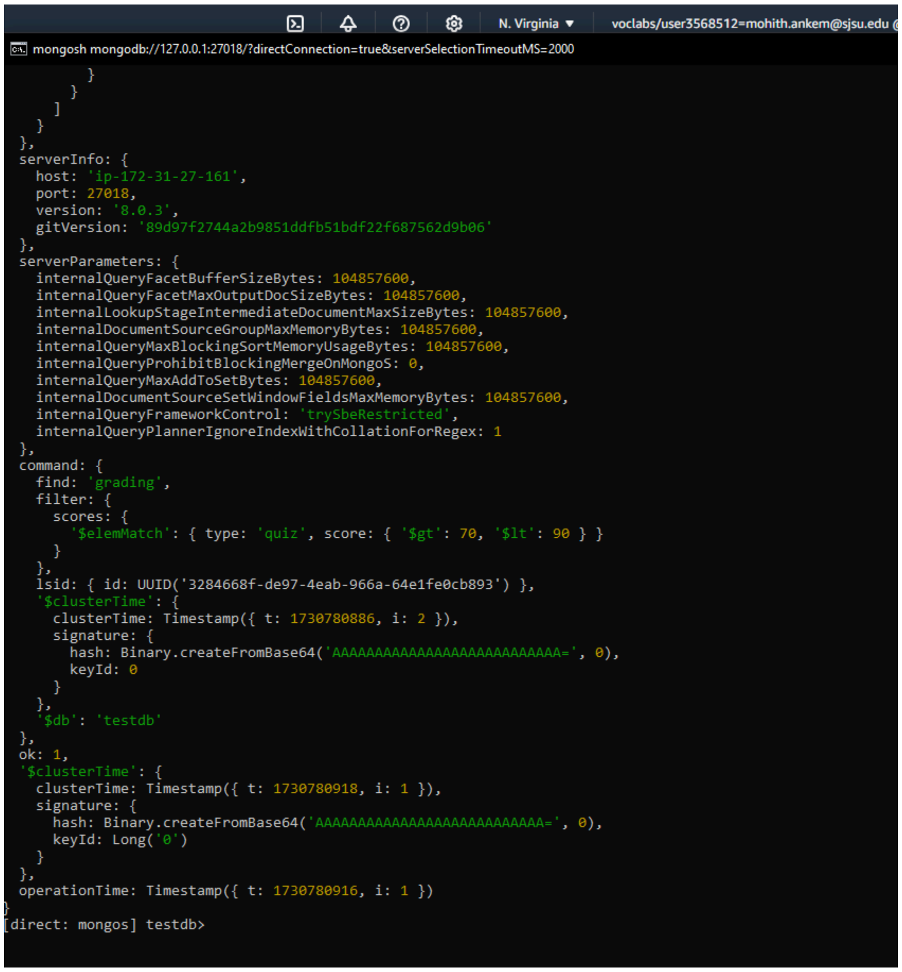
```
        {
          shardName: 'shard2ReplSet',
          executionSuccess: true,
          nReturned: 6391,
          executionTimeMillis: 153,
          totalKeysExamined: 0,
          totalDocsExamined: 32850,
          executionStages: {
            isCached: false,
            stage: 'SHARDING_FILTER',
            nReturned: 6391,
            executionTimeMillisEstimate: 151,
            works: 32851,
            advanced: 6391,
            needTime: 26459,
            needYield: 0,
            saveState: 8,
            restoreState: 8,
            isEOF: 1,
            chunkSkips: 0,
            inputStage: {
              stage: 'COLLSCAN',
              filter: {
                scores: {
                  '$elemMatch': {
                    '$and': [
                      { type: { '$eq': 'quiz' } },
                      { score: { '$lt': 90 } },
                      { score: { '$gt': 70 } }
                    ]
                  }
                }
              },
              nReturned: 6391,
              executionTimeMillisEstimate: 139,
              works: 32851,
              advanced: 6391,
              needTime: 26459,
              needYield: 0,
              saveState: 8,
              restoreState: 8,
              isEOF: 1,
              direction: 'forward',
              docsExamined: 32850
            }
          }
        },
        {
          shardName: 'shard3ReplSet',
          executionSuccess: true,
          nReturned: 6741,
          executionTimeMillis: 148,
          totalKeysExamined: 0,
```



N. Virginia ▼ voclabs/user3568512=mohith.ankem@sjsu.edu @ 5

mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
        {
          shardName: 'shard3ReplSet',
          executionSuccess: true,
          nReturned: 6741,
          executionTimeMillis: 148,
          totalKeysExamined: 0,
          totalDocsExamined: 33490,
          executionStages: {
            isCached: false,
            stage: 'SHARDING_FILTER',
            nReturned: 6741,
            executionTimeMillisEstimate: 131,
            works: 33491,
            advanced: 6741,
            needTime: 26749,
            needYield: 0,
            saveState: 8,
            restoreState: 8,
            isEOF: 1,
            chunkSkips: 0,
            inputStage: {
              stage: 'COLLSCAN',
              filter: {
                scores: {
                  '$elemMatch': {
                    '$and': [
                      { type: { '$eq': 'quiz' } },
                      { score: { '$lt': 90 } },
                      { score: { '$gt': 70 } }
                    ]
                  }
                }
              },
              nReturned: 6741,
              executionTimeMillisEstimate: 112,
              works: 33491,
              advanced: 6741,
              needTime: 26749,
              needYield: 0,
              saveState: 8,
              restoreState: 8,
              isEOF: 1,
              direction: 'forward',
              docsExamined: 33490
            }
          }
        }
      ]
    }
  },
  serverInfo: {
    host: 'ip-172-31-27-161',
    port: 27018,
```



N. Virginia ▼ voclabs/user3568512=mohith.ankem@sjsu.edu @

mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

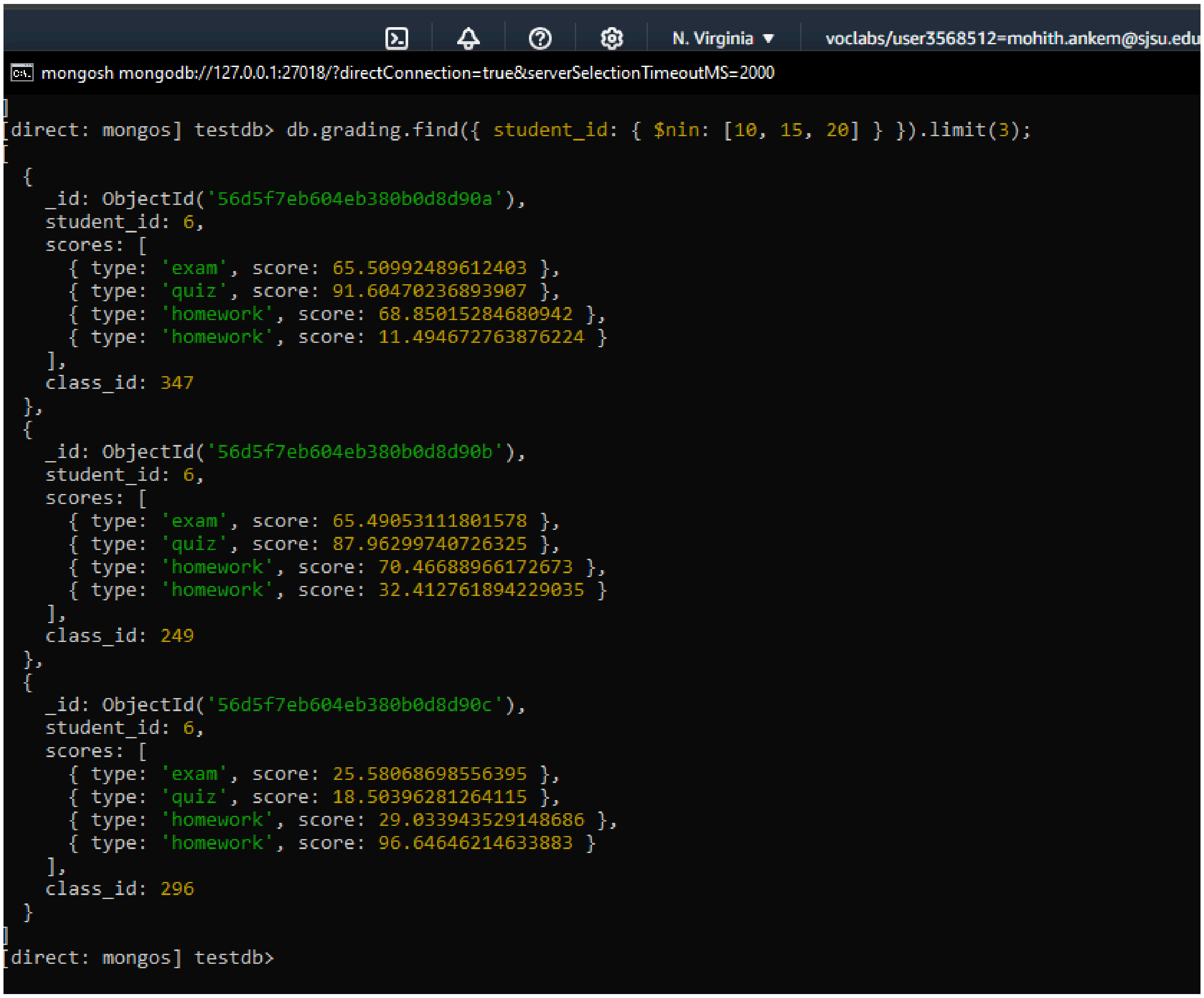
```
          }
        }
      ]
    }
  },
  serverInfo: {
    host: 'ip-172-31-27-161',
    port: 27018,
    version: '8.0.3',
    gitVersion: '89d97f2744a2b9851ddfb51bdf22f687562d9b06'
  },
  serverParameters: {
    internalQueryFacetBufferSizeBytes: 104857600,
    internalQueryFacetMaxOutputDocSizeBytes: 104857600,
    internalLookupStageIntermediateDocumentMaxSizeBytes: 104857600,
    internalDocumentSourceGroupMaxMemoryBytes: 104857600,
    internalQueryMaxBlockingSortMemoryUsageBytes: 104857600,
    internalQueryProhibitBlockingMergeOnMongoS: 0,
    internalQueryMaxAddToSetBytes: 104857600,
    internalDocumentSourceSetWindowFieldsMaxMemoryBytes: 104857600,
    internalQueryFrameworkControl: 'trySbeRestricted',
    internalQueryPlannerIgnoreIndexWithCollationForRegex: 1
  },
  command: {
    find: 'grading',
    filter: {
      scores: {
        '$elemMatch': { type: 'quiz', score: { '$gt': 70, '$lt': 90 } }
      }
    },
    lsid: { id: UUID('3284668f-de97-4eab-966a-64e1fe0cb893') },
    '$clusterTime': {
      clusterTime: Timestamp({ t: 1730780886, i: 2 }),
      signature: {
        hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
        keyId: 0
      }
    },
    '$db': 'testdb'
  },
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1730780918, i: 1 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: Long('0')
    }
  },
  operationTime: Timestamp({ t: 1730780916, i: 1 })
}
[direct: mongos] testdb>
```

It took a total of 157 ms to run the query and in this case also all 3 shards have returned the results with shard1 returned 6631, shard2 returned 6391 and shard3 returned 6741 docs.

3. <u>A query involving $in, $nin, or $all</u>

db.grading.find({ student_id: { $nin: [10, 15, 20] } }).limit(3);

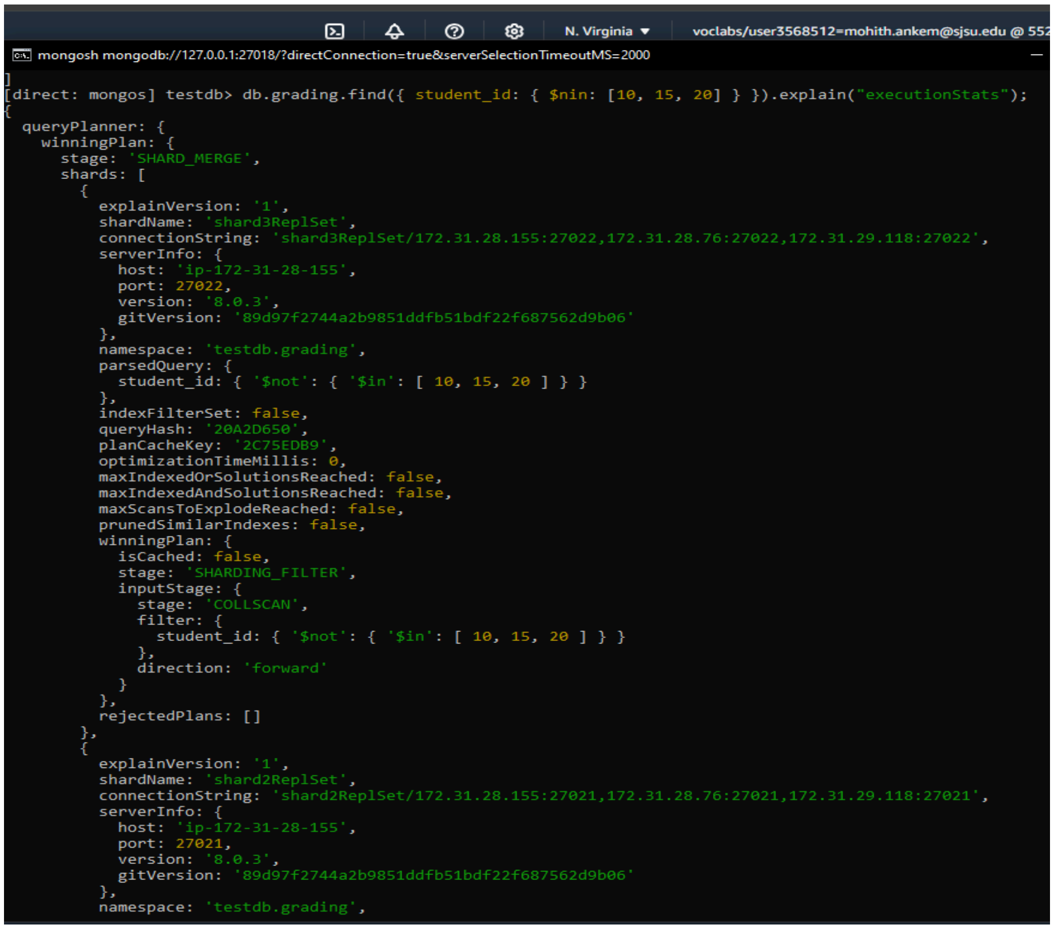
This query will retrieve documents where student_id is *not* 10, 15, or 20. By using $nin, MongoDB will filter out any documents where student_id is one of the values in the array [10, 15, 20].


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

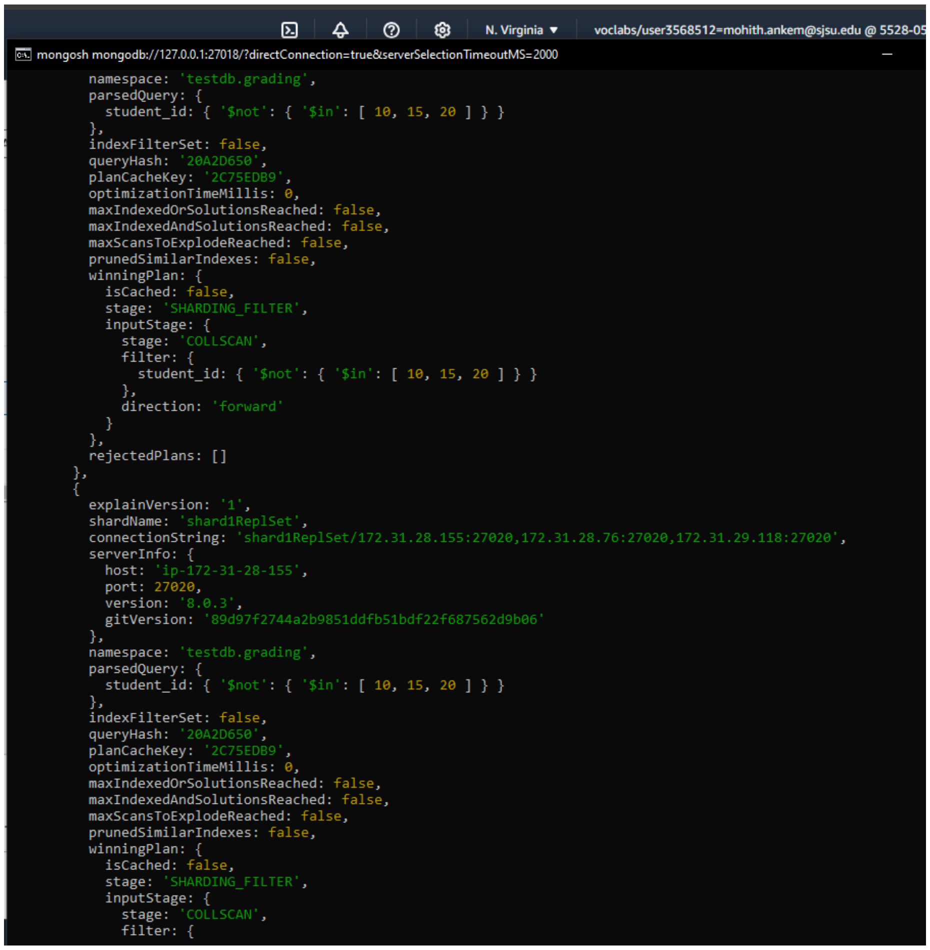
```
[direct: mongos] testdb> db.grading.find({ student_id: { $nin: [10, 15, 20] } }).limit(3);
[
  {
    _id: ObjectId('56d5f7eb604eb380b0d8d90a'),
    student_id: 6,
    scores: [
      { type: 'exam', score: 65.50992489612403 },
      { type: 'quiz', score: 91.60470236893907 },
      { type: 'homework', score: 68.85015284680942 },
      { type: 'homework', score: 11.494672763876224 }
    ],
    class_id: 347
  },
  {
    _id: ObjectId('56d5f7eb604eb380b0d8d90b'),
    student_id: 6,
    scores: [
      { type: 'exam', score: 65.49053111801578 },
      { type: 'quiz', score: 87.96299740726325 },
      { type: 'homework', score: 70.46688966172673 },
      { type: 'homework', score: 32.412761894229035 }
    ],
    class_id: 249
  },
  {
    _id: ObjectId('56d5f7eb604eb380b0d8d90c'),
    student_id: 6,
    scores: [
      { type: 'exam', score: 25.58068698556395 },
      { type: 'quiz', score: 18.50396281264115 },
      { type: 'homework', score: 29.033943529148686 },
      { type: 'homework', score: 96.64646214633883 }
    ],
    class_id: 296
  }
]
[direct: mongos] testdb>
```

Execution Stats:

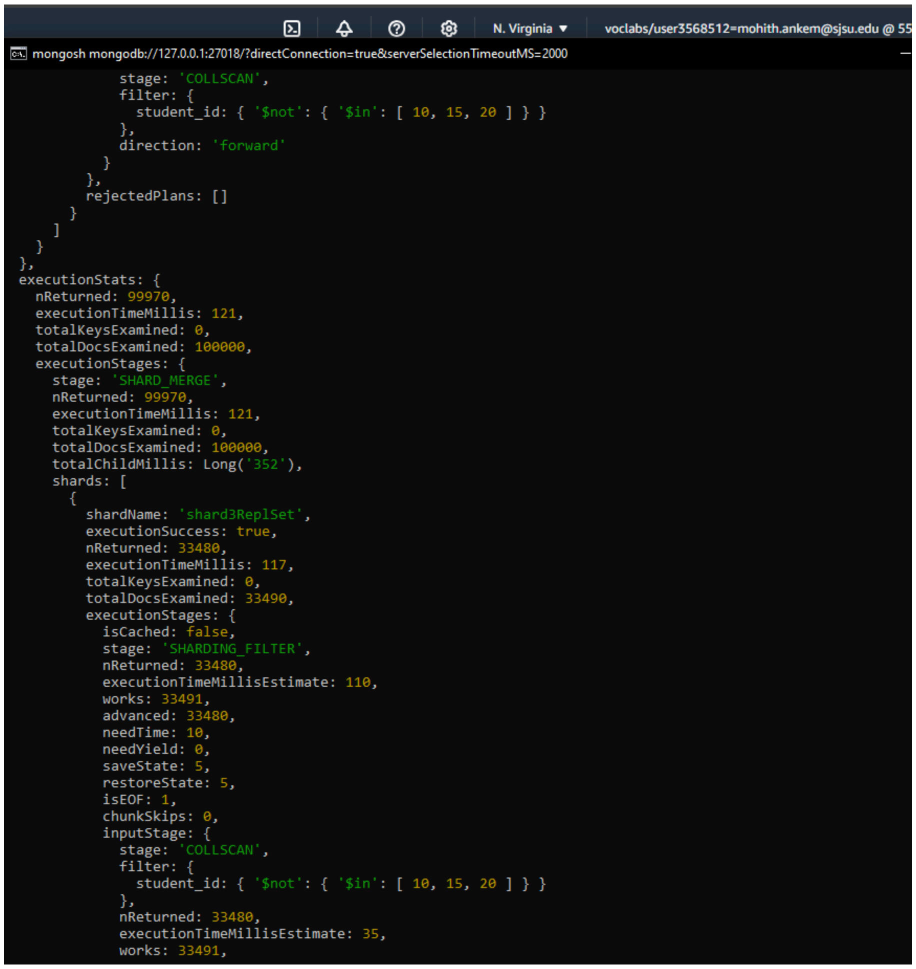
It took a total of 121 ms to run the query and in this case also all 3 shards have returned the results with shard1 returned 33660, shard2 returned 33830 and shard3 returned 33480 docs.



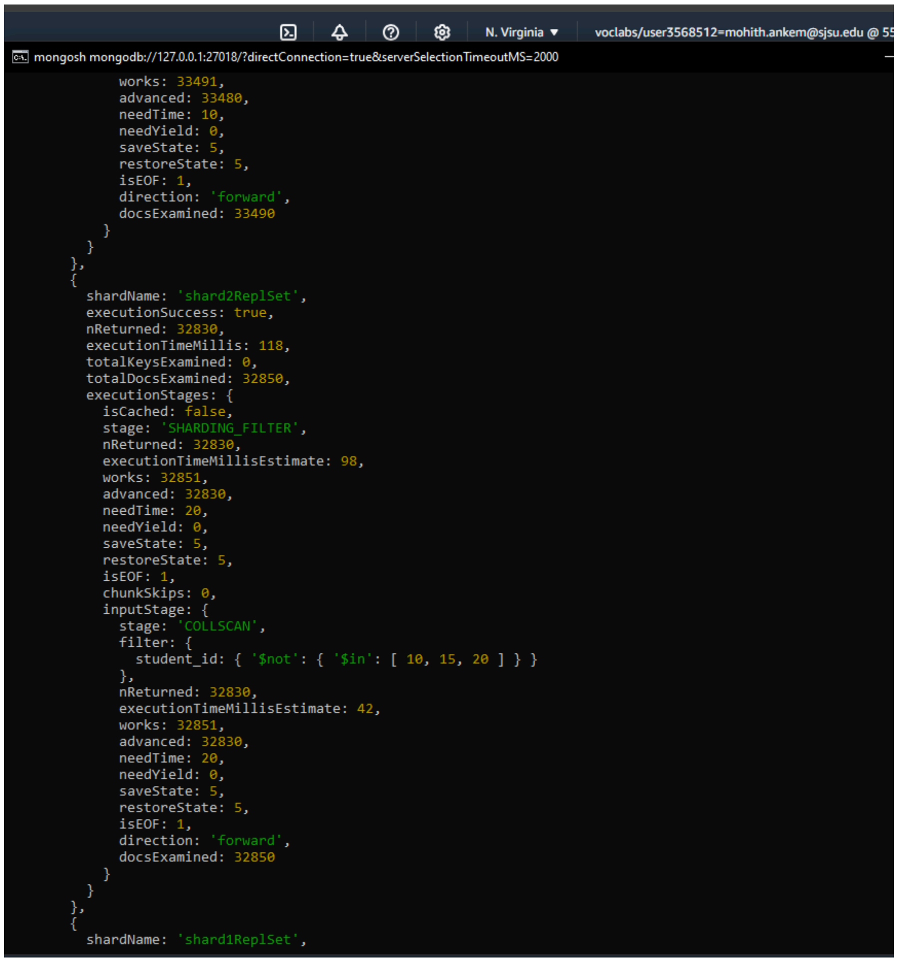
```
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000
]
[direct: mongos] testdb> db.grading.find({ student_id: { $nin: [10, 15, 20] } }).explain("executionStats");
{
  queryPlanner: {
    winningPlan: {
      stage: 'SHARD_MERGE',
      shards: [
        {
          explainVersion: '1',
          shardName: 'shard3ReplSet',
          connectionString: 'shard3ReplSet/172.31.28.155:27022,172.31.28.76:27022,172.31.29.118:27022',
          serverInfo: {
            host: 'ip-172-31-28-155',
            port: 27022,
            version: '8.0.3',
            gitVersion: '89d97f2744a2b9851ddfb51bdf22f687562d9b06'
          },
          namespace: 'testdb.grading',
          parsedQuery: {
            student_id: { '$not': { '$in': [ 10, 15, 20 ] } }
          },
          indexFilterSet: false,
          queryHash: '20A2D650',
          planCacheKey: '2C75EDB9',
          optimizationTimeMillis: 0,
          maxIndexedOrSolutionsReached: false,
          maxIndexedAndSolutionsReached: false,
          maxScansToExplodeReached: false,
          prunedSimilarIndexes: false,
          winningPlan: {
            isCached: false,
            stage: 'SHARDING_FILTER',
            inputStage: {
              stage: 'COLLSCAN',
              filter: {
                student_id: { '$not': { '$in': [ 10, 15, 20 ] } }
              },
              direction: 'forward'
            }
          },
          rejectedPlans: []
        },
        {
          explainVersion: '1',
          shardName: 'shard2ReplSet',
          connectionString: 'shard2ReplSet/172.31.28.155:27021,172.31.28.76:27021,172.31.29.118:27021',
          serverInfo: {
            host: 'ip-172-31-28-155',
            port: 27021,
            version: '8.0.3',
            gitVersion: '89d97f2744a2b9851ddfb51bdf22f687562d9b06'
          },
          namespace: 'testdb.grading',
```



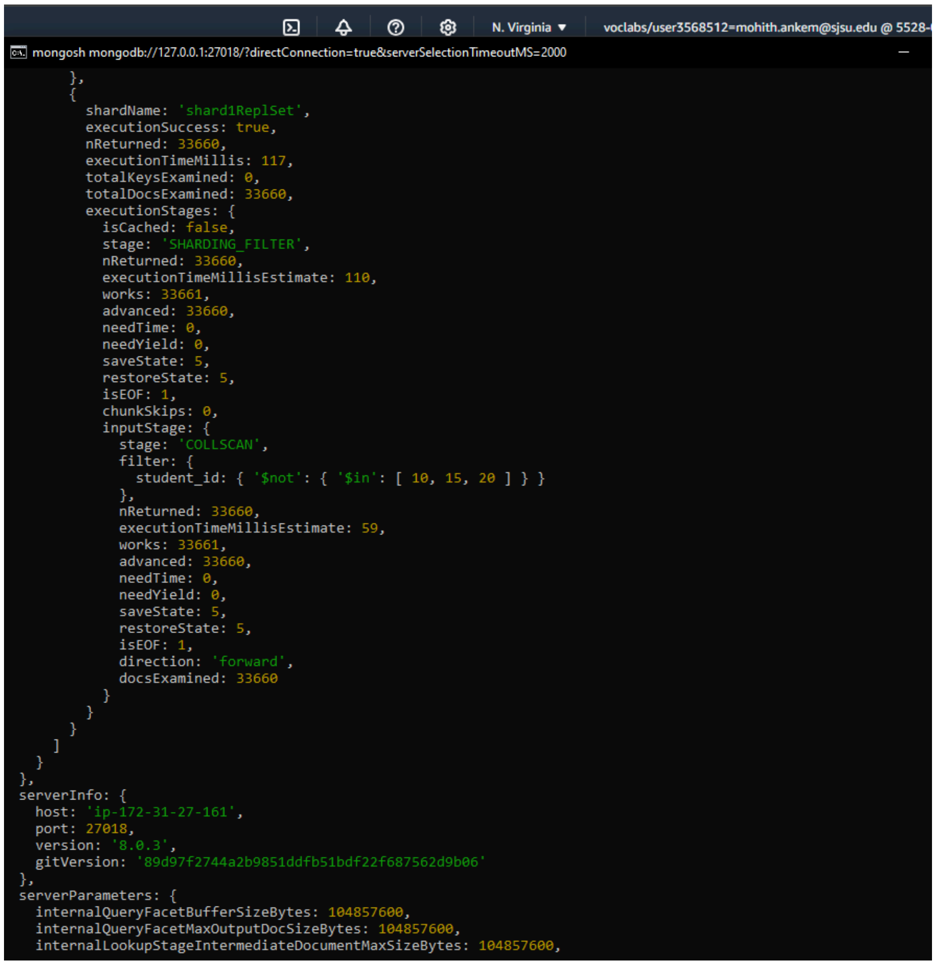
```
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000
          namespace: 'testdb.grading',
          parsedQuery: {
            student_id: { '$not': { '$in': [ 10, 15, 20 ] } }
          },
          indexFilterSet: false,
          queryHash: '20A2D650',
          planCacheKey: '2C75EDB9',
          optimizationTimeMillis: 0,
          maxIndexedOrSolutionsReached: false,
          maxIndexedAndSolutionsReached: false,
          maxScansToExplodeReached: false,
          prunedSimilarIndexes: false,
          winningPlan: {
            isCached: false,
            stage: 'SHARDING_FILTER',
            inputStage: {
              stage: 'COLLSCAN',
              filter: {
                student_id: { '$not': { '$in': [ 10, 15, 20 ] } }
              },
              direction: 'forward'
            }
          },
          rejectedPlans: []
        },
        {
          explainVersion: '1',
          shardName: 'shard1ReplSet',
          connectionString: 'shard1ReplSet/172.31.28.155:27020,172.31.28.76:27020,172.31.29.118:27020',
          serverInfo: {
            host: 'ip-172-31-28-155',
            port: 27020,
            version: '8.0.3',
            gitVersion: '89d97f2744a2b9851ddfb51bdf22f687562d9b06'
          },
          namespace: 'testdb.grading',
          parsedQuery: {
            student_id: { '$not': { '$in': [ 10, 15, 20 ] } }
          },
          indexFilterSet: false,
          queryHash: '20A2D650',
          planCacheKey: '2C75EDB9',
          optimizationTimeMillis: 0,
          maxIndexedOrSolutionsReached: false,
          maxIndexedAndSolutionsReached: false,
          maxScansToExplodeReached: false,
          prunedSimilarIndexes: false,
          winningPlan: {
            isCached: false,
            stage: 'SHARDING_FILTER',
            inputStage: {
              stage: 'COLLSCAN',
              filter: {
```


N. Virginia ▼
voclabs/user3568512=mohith.ankem@sjsu.edu @ 55
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

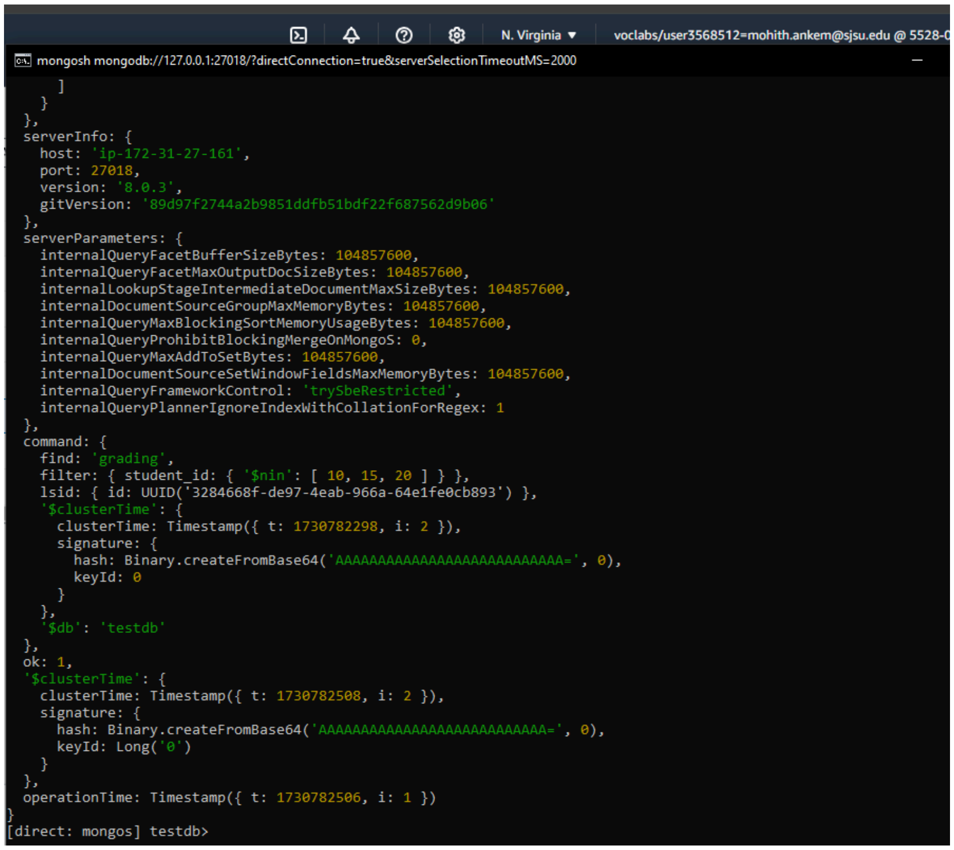
```
          stage: 'COLLSCAN',
          filter: {
            student_id: { '$not': { '$in': [ 10, 15, 20 ] } }
          },
          direction: 'forward'
        }
      },
      rejectedPlans: []
    }
  ]
 }
},
executionStats: {
  nReturned: 99970,
  executionTimeMillis: 121,
  totalKeysExamined: 0,
  totalDocsExamined: 100000,
  executionStages: {
    stage: 'SHARD_MERGE',
    nReturned: 99970,
    executionTimeMillis: 121,
    totalKeysExamined: 0,
    totalDocsExamined: 100000,
    totalChildMillis: Long('352'),
    shards: [
      {
        shardName: 'shard3ReplSet',
        executionSuccess: true,
        nReturned: 33480,
        executionTimeMillis: 117,
        totalKeysExamined: 0,
        totalDocsExamined: 33490,
        executionStages: {
          isCached: false,
          stage: 'SHARDING_FILTER',
          nReturned: 33480,
          executionTimeMillisEstimate: 110,
          works: 33491,
          advanced: 33480,
          needTime: 10,
          needYield: 0,
          saveState: 5,
          restoreState: 5,
          isEOF: 1,
          chunkSkips: 0,
          inputStage: {
            stage: 'COLLSCAN',
            filter: {
              student_id: { '$not': { '$in': [ 10, 15, 20 ] } }
            },
            nReturned: 33480,
            executionTimeMillisEstimate: 35,
            works: 33491,
```


N. Virginia ▼
voclabs/user3568512=mohith.ankem@sjsu.edu @ 55
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
            works: 33491,
            advanced: 33480,
            needTime: 10,
            needYield: 0,
            saveState: 5,
            restoreState: 5,
            isEOF: 1,
            direction: 'forward',
            docsExamined: 33490
          }
        }
      },
      {
        shardName: 'shard2ReplSet',
        executionSuccess: true,
        nReturned: 32830,
        executionTimeMillis: 118,
        totalKeysExamined: 0,
        totalDocsExamined: 32850,
        executionStages: {
          isCached: false,
          stage: 'SHARDING_FILTER',
          nReturned: 32830,
          executionTimeMillisEstimate: 98,
          works: 32851,
          advanced: 32830,
          needTime: 20,
          needYield: 0,
          saveState: 5,
          restoreState: 5,
          isEOF: 1,
          chunkSkips: 0,
          inputStage: {
            stage: 'COLLSCAN',
            filter: {
              student_id: { '$not': { '$in': [ 10, 15, 20 ] } }
            },
            nReturned: 32830,
            executionTimeMillisEstimate: 42,
            works: 32851,
            advanced: 32830,
            needTime: 20,
            needYield: 0,
            saveState: 5,
            restoreState: 5,
            isEOF: 1,
            direction: 'forward',
            docsExamined: 32850
          }
        }
      },
      {
        shardName: 'shard1ReplSet',
```



```
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000
        },
        {
          shardName: 'shard1ReplSet',
          executionSuccess: true,
          nReturned: 33660,
          executionTimeMillis: 117,
          totalKeysExamined: 0,
          totalDocsExamined: 33660,
          executionStages: {
            isCached: false,
            stage: 'SHARDING_FILTER',
            nReturned: 33660,
            executionTimeMillisEstimate: 110,
            works: 33661,
            advanced: 33660,
            needTime: 0,
            needYield: 0,
            saveState: 5,
            restoreState: 5,
            isEOF: 1,
            chunkSkips: 0,
            inputStage: {
              stage: 'COLLSCAN',
              filter: {
                student_id: { '$not': { '$in': [ 10, 15, 20 ] } }
              },
              nReturned: 33660,
              executionTimeMillisEstimate: 59,
              works: 33661,
              advanced: 33660,
              needTime: 0,
              needYield: 0,
              saveState: 5,
              restoreState: 5,
              isEOF: 1,
              direction: 'forward',
              docsExamined: 33660
            }
          }
        }
      ]
    }
  },
  serverInfo: {
    host: 'ip-172-31-27-161',
    port: 27018,
    version: '8.0.3',
    gitVersion: '89d97f2744a2b9851ddfb51bdf22f687562d9b06'
  },
  serverParameters: {
    internalQueryFacetBufferSizeBytes: 104857600,
    internalQueryFacetMaxOutputDocSizeBytes: 104857600,
    internalLookupStageIntermediateDocumentMaxSizeBytes: 104857600,
```



```
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000
      ]
    }
  },
  serverInfo: {
    host: 'ip-172-31-27-161',
    port: 27018,
    version: '8.0.3',
    gitVersion: '89d97f2744a2b9851ddfb51bdf22f687562d9b06'
  },
  serverParameters: {
    internalQueryFacetBufferSizeBytes: 104857600,
    internalQueryFacetMaxOutputDocSizeBytes: 104857600,
    internalLookupStageIntermediateDocumentMaxSizeBytes: 104857600,
    internalDocumentSourceGroupMaxMemoryBytes: 104857600,
    internalQueryMaxBlockingSortMemoryUsageBytes: 104857600,
    internalQueryProhibitBlockingMergeOnMongoS: 0,
    internalQueryMaxAddToSetBytes: 104857600,
    internalDocumentSourceSetWindowFieldsMaxMemoryBytes: 104857600,
    internalQueryFrameworkControl: 'trySbeRestricted',
    internalQueryPlannerIgnoreIndexWithCollationForRegex: 1
  },
  command: {
    find: 'grading',
    filter: { student_id: { '$nin': [ 10, 15, 20 ] } },
    lsid: { id: UUID('3284668f-de97-4eab-966a-64e1fe0cb893') },
    '$clusterTime': {
      clusterTime: Timestamp({ t: 1730782298, i: 2 }),
      signature: {
        hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
        keyId: 0
      }
    },
    '$db': 'testdb'
  },
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1730782508, i: 2 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: Long('0')
    }
  },
  operationTime: Timestamp({ t: 1730782506, i: 1 })
}
[direct: mongos] testdb>
```

4. A query involving aggregate()

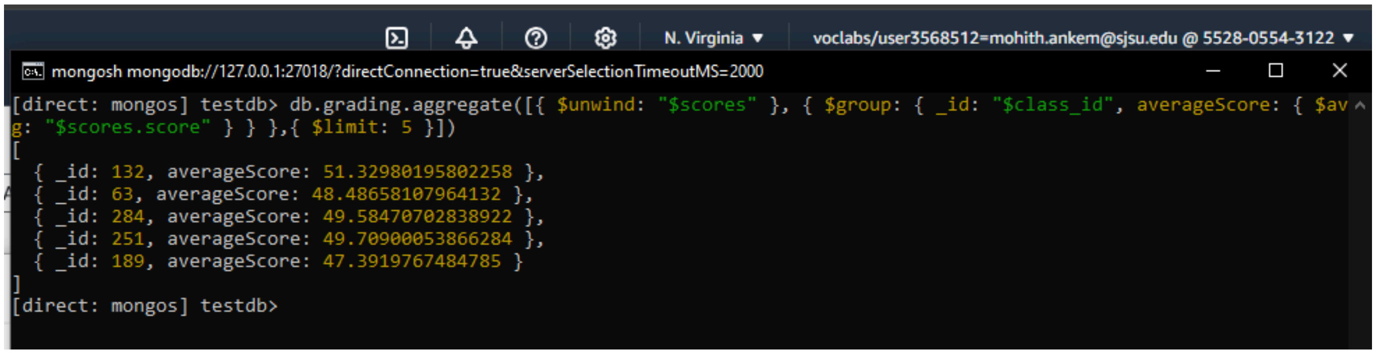
```
db.grading.aggregate([ { $unwind: "$scores" },

 { $group: { _id: "$class_id", averageScore: { $avg: "$scores.score" } } }

]).explain("executionStats");
```

$unwind:

- { $unwind: "$scores" } takes each document's scores array and "unwinds" it. This means that for each element in the scores array, it creates a separate document.
- If a document has multiple scores, each score type (like "exam", "quiz", or "homework") will become its own document, allowing us to handle each score individually.

$group:

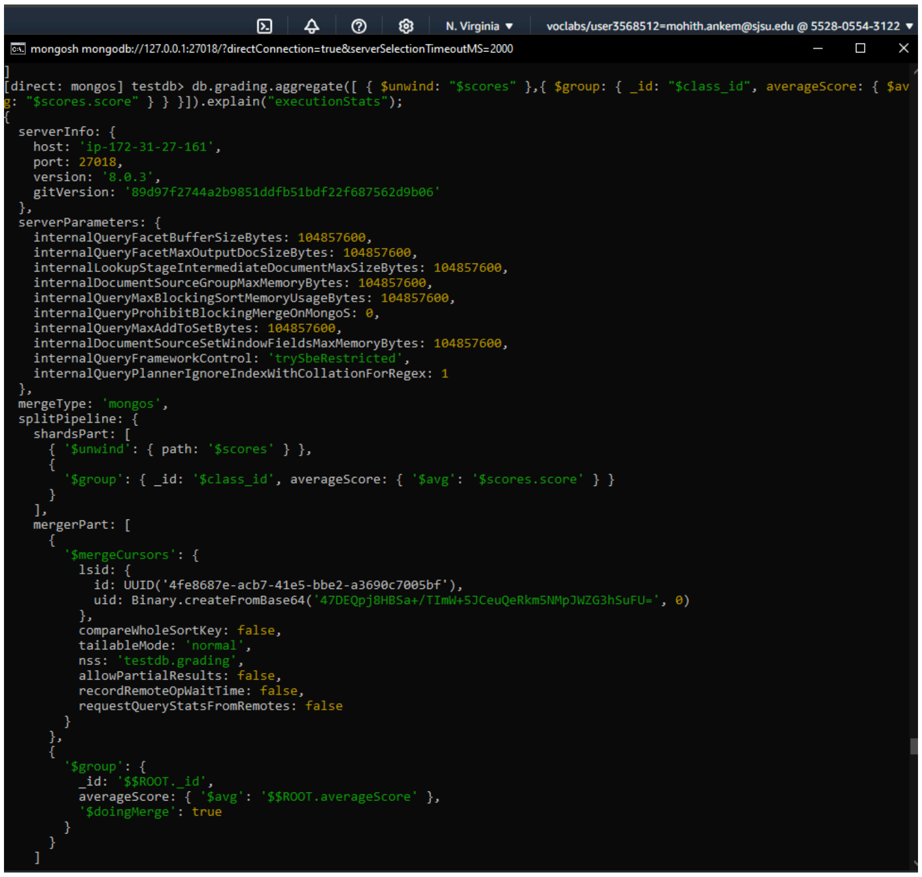
- { $group: { _id: "$class_id", averageScore: { $avg: "$scores.score" } } } groups the unwound documents by class_id and calculates the average of the score field for each class.
- The result is one document per class_id containing the calculated averageScore.


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
[direct: mongos] testdb> db.grading.aggregate([{ $unwind: "$scores" }, { $group: { _id: "$class_id", averageScore: { $av
g: "$scores.score" } } },{ $limit: 5 }])
[
  { _id: 132, averageScore: 51.32980195802258 },
  { _id: 63, averageScore: 48.48658107964132 },
  { _id: 284, averageScore: 49.58470702838922 },
  { _id: 251, averageScore: 49.70900053866284 },
  { _id: 189, averageScore: 47.3919767484785 }
]
[direct: mongos] testdb>
```

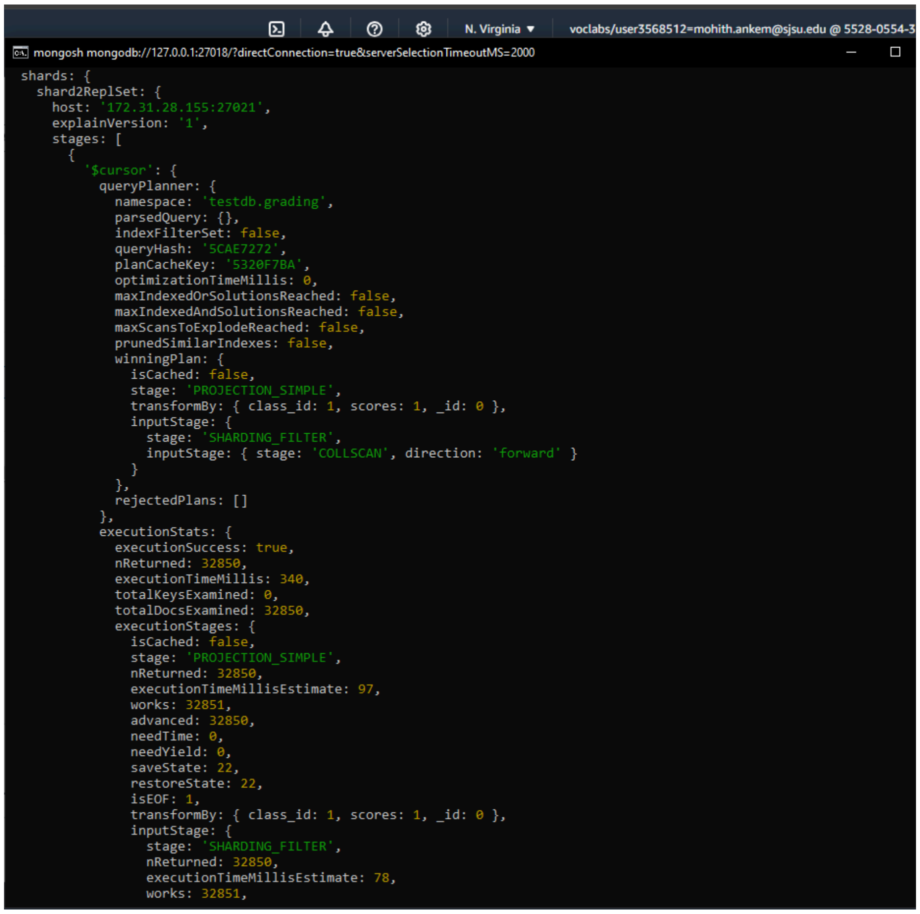
Execution Stats:

It took a total of 346 ms to run the query and in this case also all 3 shards have returned the results with shard1 returned 33660, shard2 returned 33850 and shard3 returned 33490 docs.


N. Virginia ▼ voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122 ▼

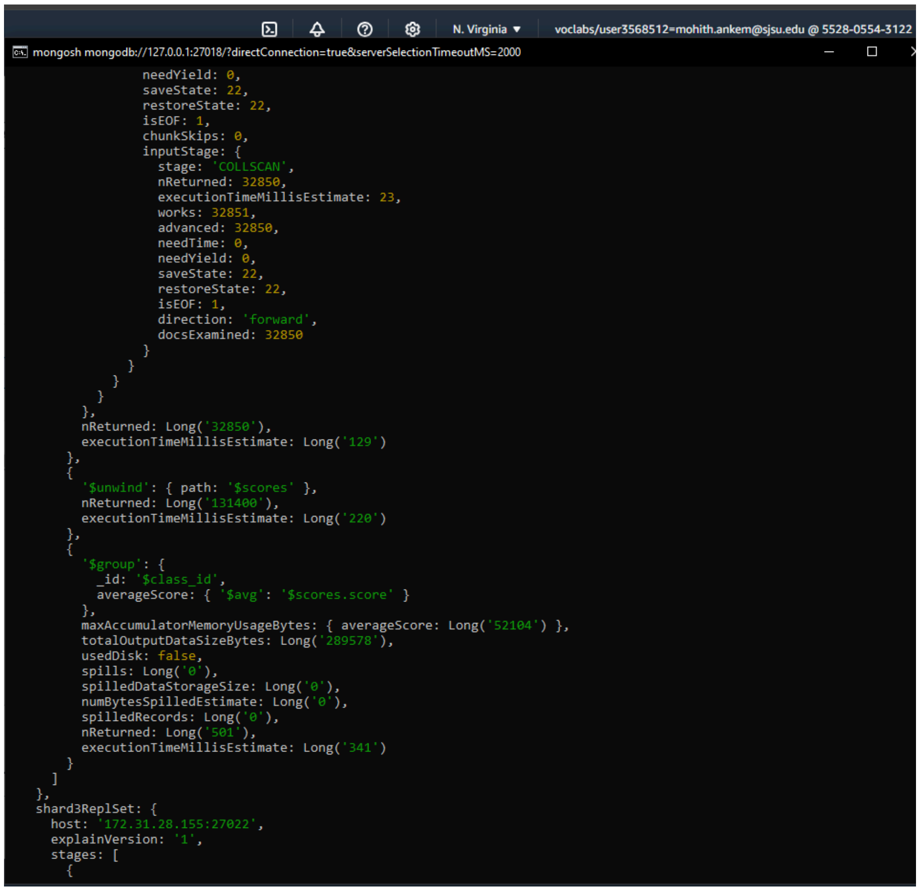
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
]
[direct: mongos] testdb> db.grading.aggregate([ { $unwind: "$scores" },{ $group: { _id: "$class_id", averageScore: { $av
g: "$scores.score" } } }]).explain("executionStats");
{
  serverInfo: {
    host: 'ip-172-31-27-161',
    port: 27018,
    version: '8.0.3',
    gitVersion: '89d97f2744a2b9851ddfb51bdf22f687562d9b06'
  },
  serverParameters: {
    internalQueryFacetBufferSizeBytes: 104857600,
    internalQueryFacetMaxOutputDocSizeBytes: 104857600,
    internalLookupStageIntermediateDocumentMaxSizeBytes: 104857600,
    internalDocumentSourceGroupMaxMemoryBytes: 104857600,
    internalQueryMaxBlockingSortMemoryUsageBytes: 104857600,
    internalQueryProhibitBlockingMergeOnMongoS: 0,
    internalQueryMaxAddToSetBytes: 104857600,
    internalDocumentSourceSetWindowFieldsMaxMemoryBytes: 104857600,
    internalQueryFrameworkControl: 'trySbeRestricted',
    internalQueryPlannerIgnoreIndexWithCollationForRegex: 1
  },
  mergeType: 'mongos',
  splitPipeline: {
    shardsPart: [
      { '$unwind': { path: '$scores' } },
      {
        '$group': { _id: '$class_id', averageScore: { '$avg': '$scores.score' } }
      }
    ],
    mergerPart: [
      {
        '$mergeCursors': {
          lsid: {
            id: UUID('4fe8687e-acb7-41e5-bbe2-a3690c7005bf'),
            uid: Binary.createFromBase64('47DEQpj8HBSa+/TImW+5JCeuQeRkm5NMpJWZG3hSuFU=', 0)
          },
          compareWholeSortKey: false,
          tailableMode: 'normal',
          nss: 'testdb.grading',
          allowPartialResults: false,
          recordRemoteOpWaitTime: false,
          requestQueryStatsFromRemotes: false
        }
      },
      {
        '$group': {
          _id: '$$ROOT._id',
          averageScore: { '$avg': '$$ROOT.averageScore' },
          '$doingMerge': true
        }
      }
    ]
```


N. Virginia ▼ voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3

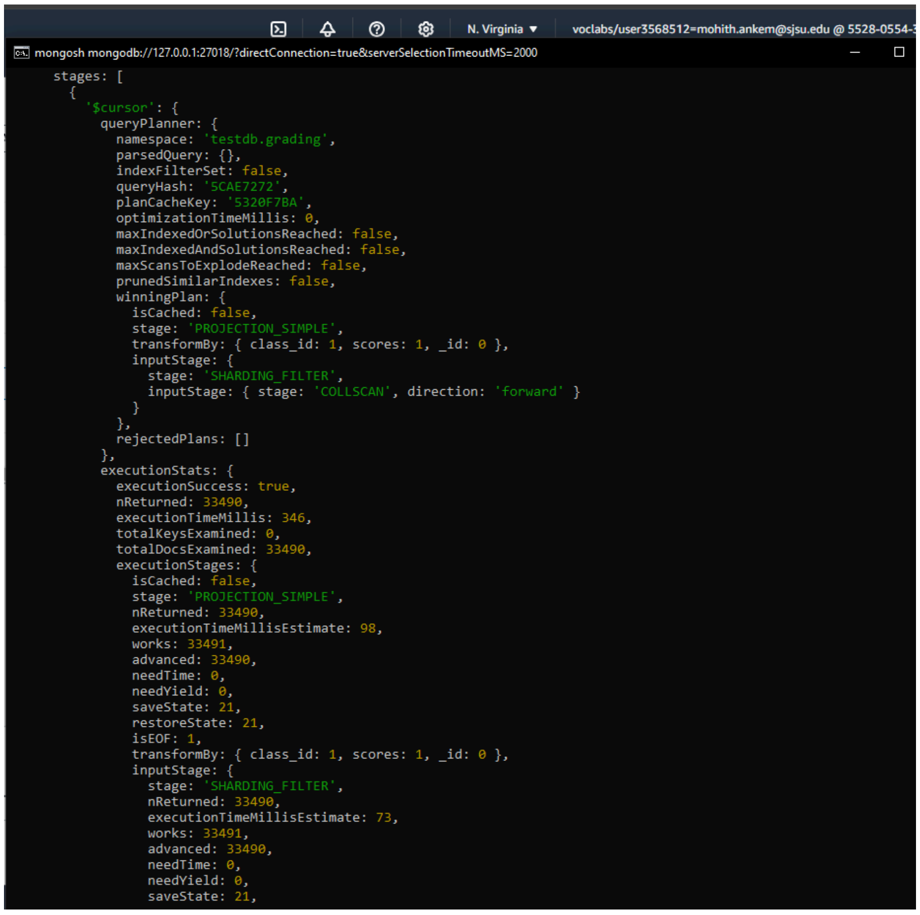
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
  shards: {
    shard2ReplSet: {
      host: '172.31.28.155:27021',
      explainVersion: '1',
      stages: [
        {
          '$cursor': {
            queryPlanner: {
              namespace: 'testdb.grading',
              parsedQuery: {},
              indexFilterSet: false,
              queryHash: '5CAE7272',
              planCacheKey: '5320F7BA',
              optimizationTimeMillis: 0,
              maxIndexedOrSolutionsReached: false,
              maxIndexedAndSolutionsReached: false,
              maxScansToExplodeReached: false,
              prunedSimilarIndexes: false,
              winningPlan: {
                isCached: false,
                stage: 'PROJECTION_SIMPLE',
                transformBy: { class_id: 1, scores: 1, _id: 0 },
                inputStage: {
                  stage: 'SHARDING_FILTER',
                  inputStage: { stage: 'COLLSCAN', direction: 'forward' }
                }
              },
              rejectedPlans: []
            },
            executionStats: {
              executionSuccess: true,
              nReturned: 32850,
              executionTimeMillis: 340,
              totalKeysExamined: 0,
              totalDocsExamined: 32850,
              executionStages: {
                isCached: false,
                stage: 'PROJECTION_SIMPLE',
                nReturned: 32850,
                executionTimeMillisEstimate: 97,
                works: 32851,
                advanced: 32850,
                needTime: 0,
                needYield: 0,
                saveState: 22,
                restoreState: 22,
                isEOF: 1,
                transformBy: { class_id: 1, scores: 1, _id: 0 },
                inputStage: {
                  stage: 'SHARDING_FILTER',
                  nReturned: 32850,
                  executionTimeMillisEstimate: 78,
                  works: 32851,
```



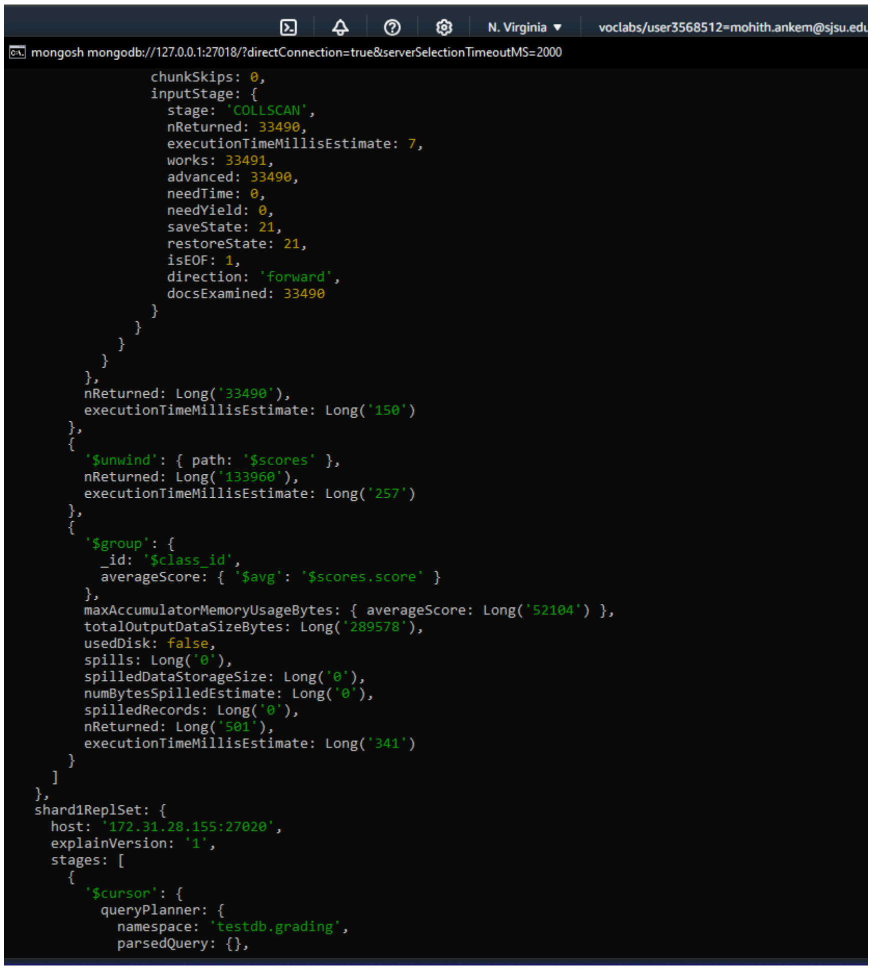
N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
                  needYield: 0,
                  saveState: 22,
                  restoreState: 22,
                  isEOF: 1,
                  chunkSkips: 0,
                  inputStage: {
                    stage: 'COLLSCAN',
                    nReturned: 32850,
                    executionTimeMillisEstimate: 23,
                    works: 32851,
                    advanced: 32850,
                    needTime: 0,
                    needYield: 0,
                    saveState: 22,
                    restoreState: 22,
                    isEOF: 1,
                    direction: 'forward',
                    docsExamined: 32850
                  }
                }
              }
            }
          },
          nReturned: Long('32850'),
          executionTimeMillisEstimate: Long('129')
        },
        {
          '$unwind': { path: '$scores' },
          nReturned: Long('131400'),
          executionTimeMillisEstimate: Long('220')
        },
        {
          '$group': {
            _id: '$class_id',
            averageScore: { '$avg': '$scores.score' }
          },
          maxAccumulatorMemoryUsageBytes: { averageScore: Long('52104') },
          totalOutputDataSizeBytes: Long('289578'),
          usedDisk: false,
          spills: Long('0'),
          spilledDataStorageSize: Long('0'),
          numBytesSpilledEstimate: Long('0'),
          spilledRecords: Long('0'),
          nReturned: Long('501'),
          executionTimeMillisEstimate: Long('341')
        }
      ]
    },
    shard3ReplSet: {
      host: '172.31.28.155:27022',
      explainVersion: '1',
      stages: [
        {
```

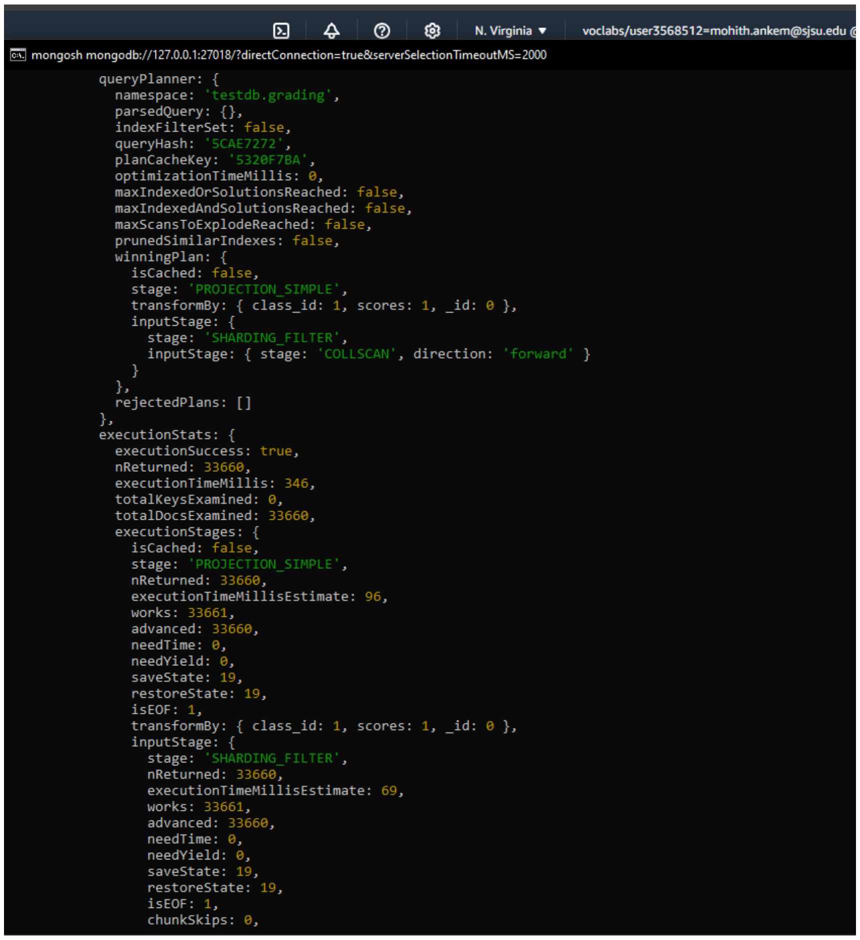


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

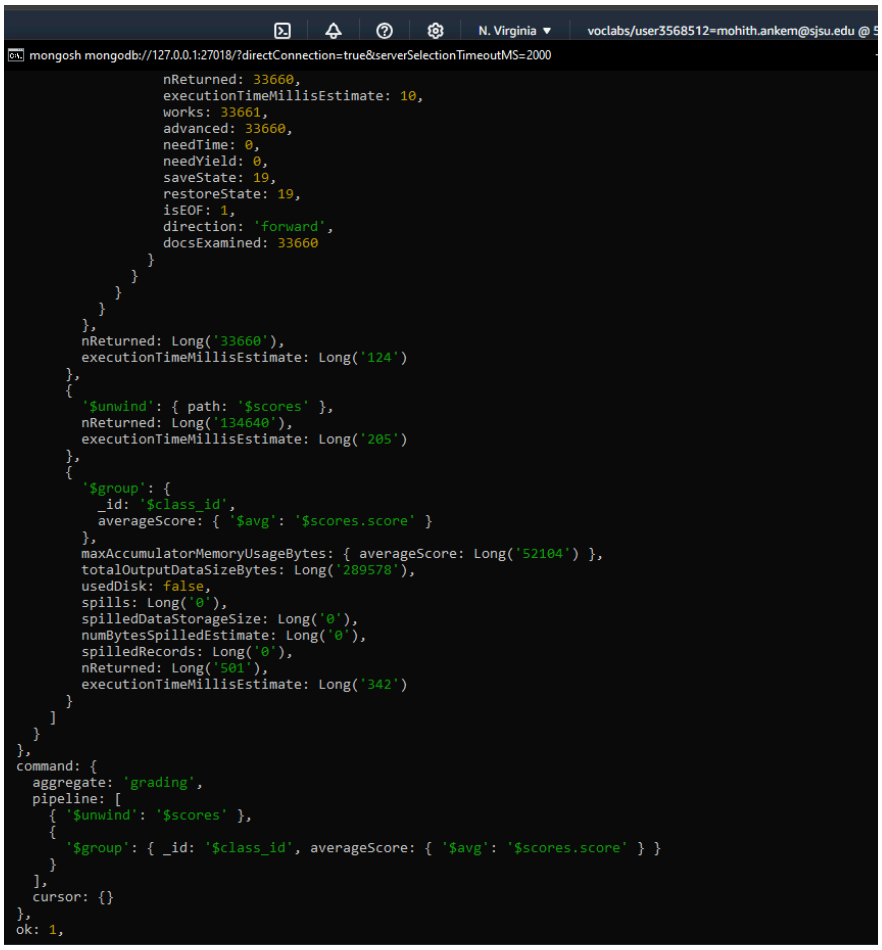
```
      stages: [
        {
          '$cursor': {
            queryPlanner: {
              namespace: 'testdb.grading',
              parsedQuery: {},
              indexFilterSet: false,
              queryHash: '5CAE7272',
              planCacheKey: '5320F7BA',
              optimizationTimeMillis: 0,
              maxIndexedOrSolutionsReached: false,
              maxIndexedAndSolutionsReached: false,
              maxScansToExplodeReached: false,
              prunedSimilarIndexes: false,
              winningPlan: {
                isCached: false,
                stage: 'PROJECTION_SIMPLE',
                transformBy: { class_id: 1, scores: 1, _id: 0 },
                inputStage: {
                  stage: 'SHARDING_FILTER',
                  inputStage: { stage: 'COLLSCAN', direction: 'forward' }
                }
              },
              rejectedPlans: []
            },
            executionStats: {
              executionSuccess: true,
              nReturned: 33490,
              executionTimeMillis: 346,
              totalKeysExamined: 0,
              totalDocsExamined: 33490,
              executionStages: {
                isCached: false,
                stage: 'PROJECTION_SIMPLE',
                nReturned: 33490,
                executionTimeMillisEstimate: 98,
                works: 33491,
                advanced: 33490,
                needTime: 0,
                needYield: 0,
                saveState: 21,
                restoreState: 21,
                isEOF: 1,
                transformBy: { class_id: 1, scores: 1, _id: 0 },
                inputStage: {
                  stage: 'SHARDING_FILTER',
                  nReturned: 33490,
                  executionTimeMillisEstimate: 73,
                  works: 33491,
                  advanced: 33490,
                  needTime: 0,
                  needYield: 0,
                  saveState: 21,
```



```
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000
                  chunkSkips: 0,
                  inputStage: {
                    stage: 'COLLSCAN',
                    nReturned: 33490,
                    executionTimeMillisEstimate: 7,
                    works: 33491,
                    advanced: 33490,
                    needTime: 0,
                    needYield: 0,
                    saveState: 21,
                    restoreState: 21,
                    isEOF: 1,
                    direction: 'forward',
                    docsExamined: 33490
                  }
                }
              }
            }
          },
          nReturned: Long('33490'),
          executionTimeMillisEstimate: Long('150')
        },
        {
          '$unwind': { path: '$scores' },
          nReturned: Long('133960'),
          executionTimeMillisEstimate: Long('257')
        },
        {
          '$group': {
            _id: '$class_id',
            averageScore: { '$avg': '$scores.score' }
          },
          maxAccumulatorMemoryUsageBytes: { averageScore: Long('52104') },
          totalOutputDataSizeBytes: Long('289578'),
          usedDisk: false,
          spills: Long('0'),
          spilledDataStorageSize: Long('0'),
          numBytesSpilledEstimate: Long('0'),
          spilledRecords: Long('0'),
          nReturned: Long('501'),
          executionTimeMillisEstimate: Long('341')
        }
      ]
    },
    shard1ReplSet: {
      host: '172.31.28.155:27020',
      explainVersion: '1',
      stages: [
        {
          '$cursor': {
            queryPlanner: {
              namespace: 'testdb.grading',
              parsedQuery: {},
```



```
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000
            queryPlanner: {
              namespace: 'testdb.grading',
              parsedQuery: {},
              indexFilterSet: false,
              queryHash: '5CAE7272',
              planCacheKey: '5320F7BA',
              optimizationTimeMillis: 0,
              maxIndexedOrSolutionsReached: false,
              maxIndexedAndSolutionsReached: false,
              maxScansToExplodeReached: false,
              prunedSimilarIndexes: false,
              winningPlan: {
                isCached: false,
                stage: 'PROJECTION_SIMPLE',
                transformBy: { class_id: 1, scores: 1, _id: 0 },
                inputStage: {
                  stage: 'SHARDING_FILTER',
                  inputStage: { stage: 'COLLSCAN', direction: 'forward' }
                }
              },
              rejectedPlans: []
            },
            executionStats: {
              executionSuccess: true,
              nReturned: 33660,
              executionTimeMillis: 346,
              totalKeysExamined: 0,
              totalDocsExamined: 33660,
              executionStages: {
                isCached: false,
                stage: 'PROJECTION_SIMPLE',
                nReturned: 33660,
                executionTimeMillisEstimate: 96,
                works: 33661,
                advanced: 33660,
                needTime: 0,
                needYield: 0,
                saveState: 19,
                restoreState: 19,
                isEOF: 1,
                transformBy: { class_id: 1, scores: 1, _id: 0 },
                inputStage: {
                  stage: 'SHARDING_FILTER',
                  nReturned: 33660,
                  executionTimeMillisEstimate: 69,
                  works: 33661,
                  advanced: 33660,
                  needTime: 0,
                  needYield: 0,
                  saveState: 19,
                  restoreState: 19,
                  isEOF: 1,
                  chunkSkips: 0,
```



```
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000
                  nReturned: 33660,
                  executionTimeMillisEstimate: 10,
                  works: 33661,
                  advanced: 33660,
                  needTime: 0,
                  needYield: 0,
                  saveState: 19,
                  restoreState: 19,
                  isEOF: 1,
                  direction: 'forward',
                  docsExamined: 33660
                }
              }
            }
          }
        },
        nReturned: Long('33660'),
        executionTimeMillisEstimate: Long('124')
      },
      {
        '$unwind': { path: '$scores' },
        nReturned: Long('134640'),
        executionTimeMillisEstimate: Long('205')
      },
      {
        '$group': {
          _id: '$class_id',
          averageScore: { '$avg': '$scores.score' }
        },
        maxAccumulatorMemoryUsageBytes: { averageScore: Long('52104') },
        totalOutputDataSizeBytes: Long('289578'),
        usedDisk: false,
        spills: Long('0'),
        spilledDataStorageSize: Long('0'),
        numBytesSpilledEstimate: Long('0'),
        spilledRecords: Long('0'),
        nReturned: Long('501'),
        executionTimeMillisEstimate: Long('342')
      }
    ]
  }
},
command: {
  aggregate: 'grading',
  pipeline: [
    { '$unwind': '$scores' },
    {
      '$group': { _id: '$class_id', averageScore: { '$avg': '$scores.score' } }
    }
  ],
  cursor: {}
},
ok: 1,
```

5. A update

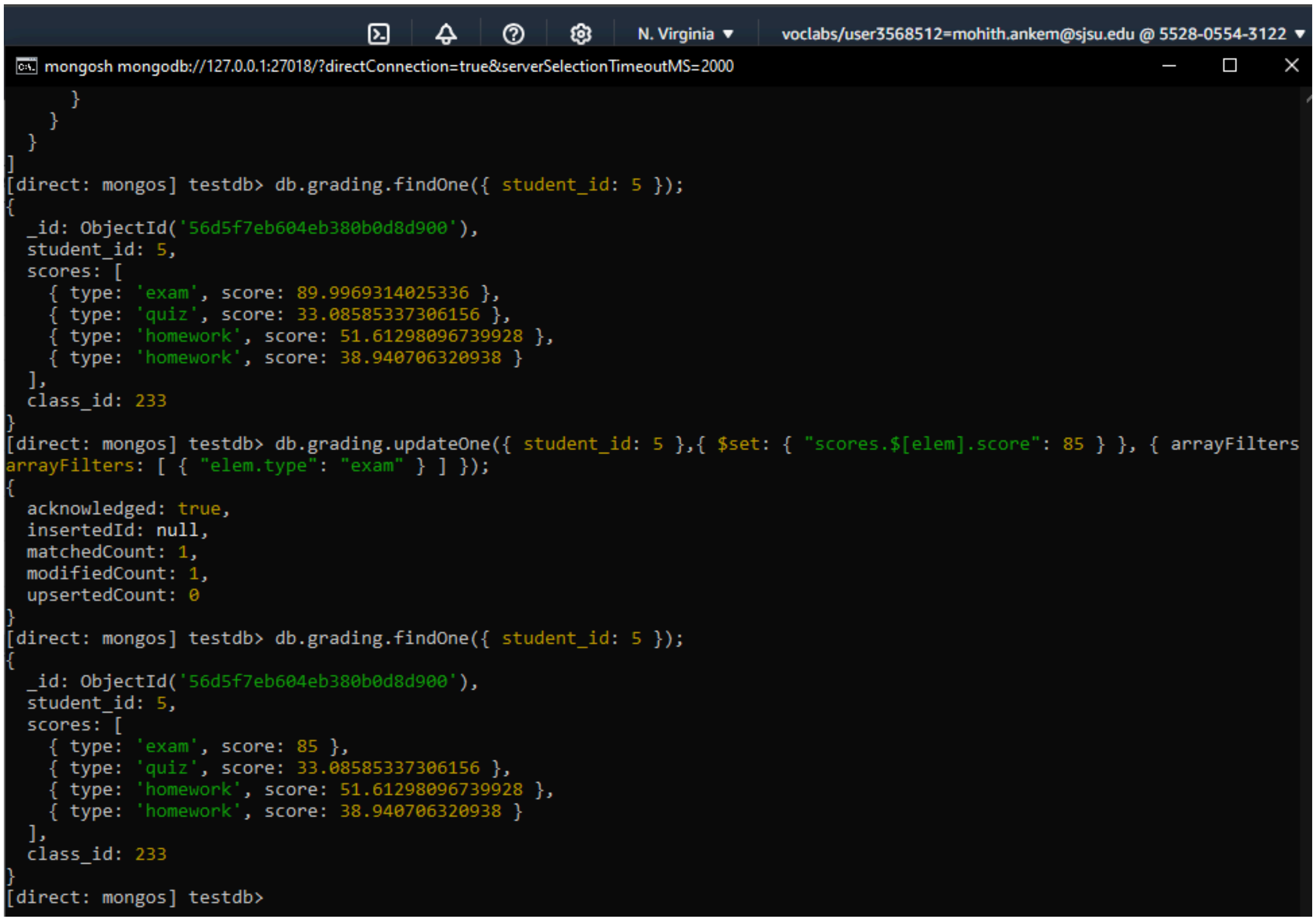
db.grading.updateOne({ student_id: 5 }, { $set: { "scores.$[elem].score": 85 } },

{ arrayFilters: [{ "elem.type": "exam" }] });

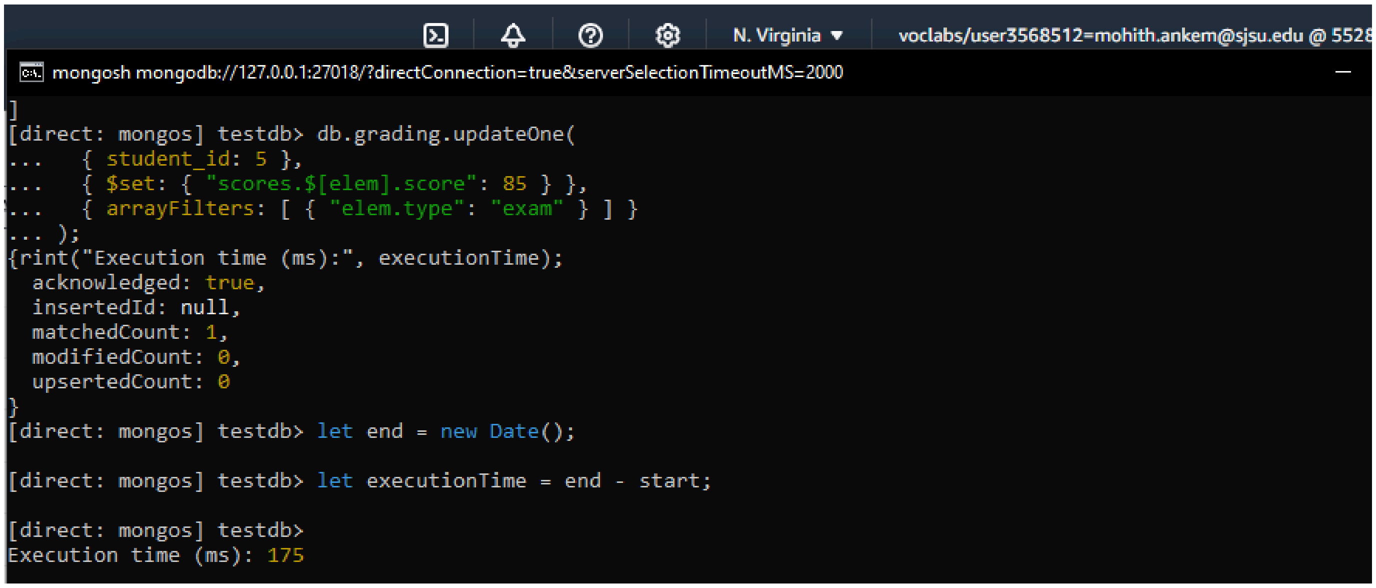
This command updates the score field to 85 for the score entry with "type": "exam" in the scores array for the document with student_id 5.

I have used commands to update the records and attached the screenshots before and after running the command for updating the record.

- It took 175 ms for the command to execute and as the explainStats command doesn't work for update query the 3 shards will serve the query.


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
      }
    }
  }
]
[direct: mongos] testdb> db.grading.findOne({ student_id: 5 });
{
  _id: ObjectId('56d5f7eb604eb380b0d8d900'),
  student_id: 5,
  scores: [
    { type: 'exam', score: 89.9969314025336 },
    { type: 'quiz', score: 33.08585337306156 },
    { type: 'homework', score: 51.61298096739928 },
    { type: 'homework', score: 38.940706320938 }
  ],
  class_id: 233
}
[direct: mongos] testdb> db.grading.updateOne({ student_id: 5 },{ $set: { "scores.$[elem].score": 85 } }, { arrayFilters
arrayFilters: [ { "elem.type": "exam" } ] });
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
[direct: mongos] testdb> db.grading.findOne({ student_id: 5 });
{
  _id: ObjectId('56d5f7eb604eb380b0d8d900'),
  student_id: 5,
  scores: [
    { type: 'exam', score: 85 },
    { type: 'quiz', score: 33.08585337306156 },
    { type: 'homework', score: 51.61298096739928 },
    { type: 'homework', score: 38.940706320938 }
  ],
  class_id: 233
}
[direct: mongos] testdb>
```

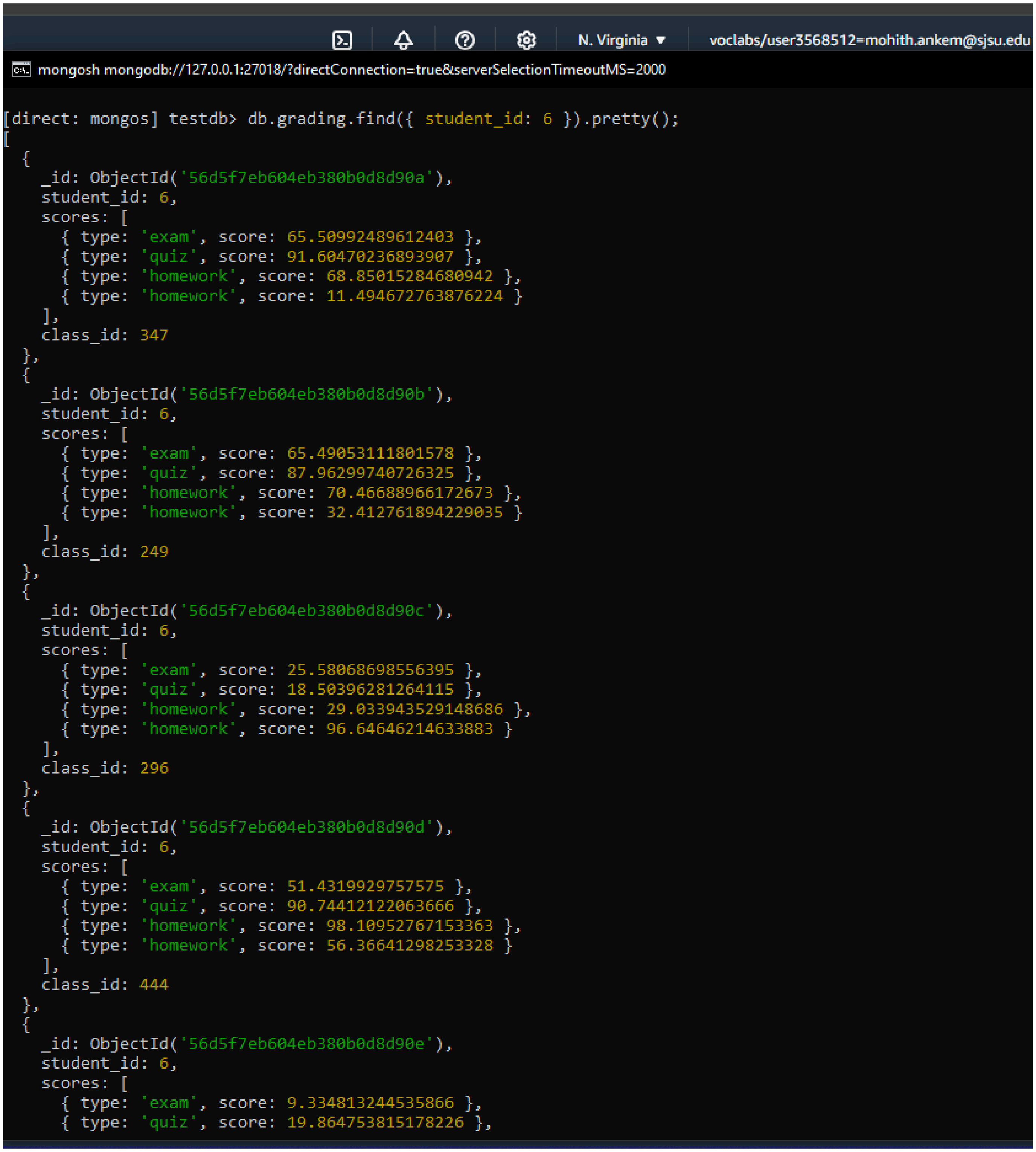

N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
]
[direct: mongos] testdb> db.grading.updateOne(
...    { student_id: 5 },
...    { $set: { "scores.$[elem].score": 85 } },
...    { arrayFilters: [ { "elem.type": "exam" } ] }
... );
{rint("Execution time (ms):", executionTime);
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 0,
  upsertedCount: 0
}
[direct: mongos] testdb> let end = new Date();

[direct: mongos] testdb> let executionTime = end - start;

[direct: mongos] testdb>
Execution time (ms): 175
```

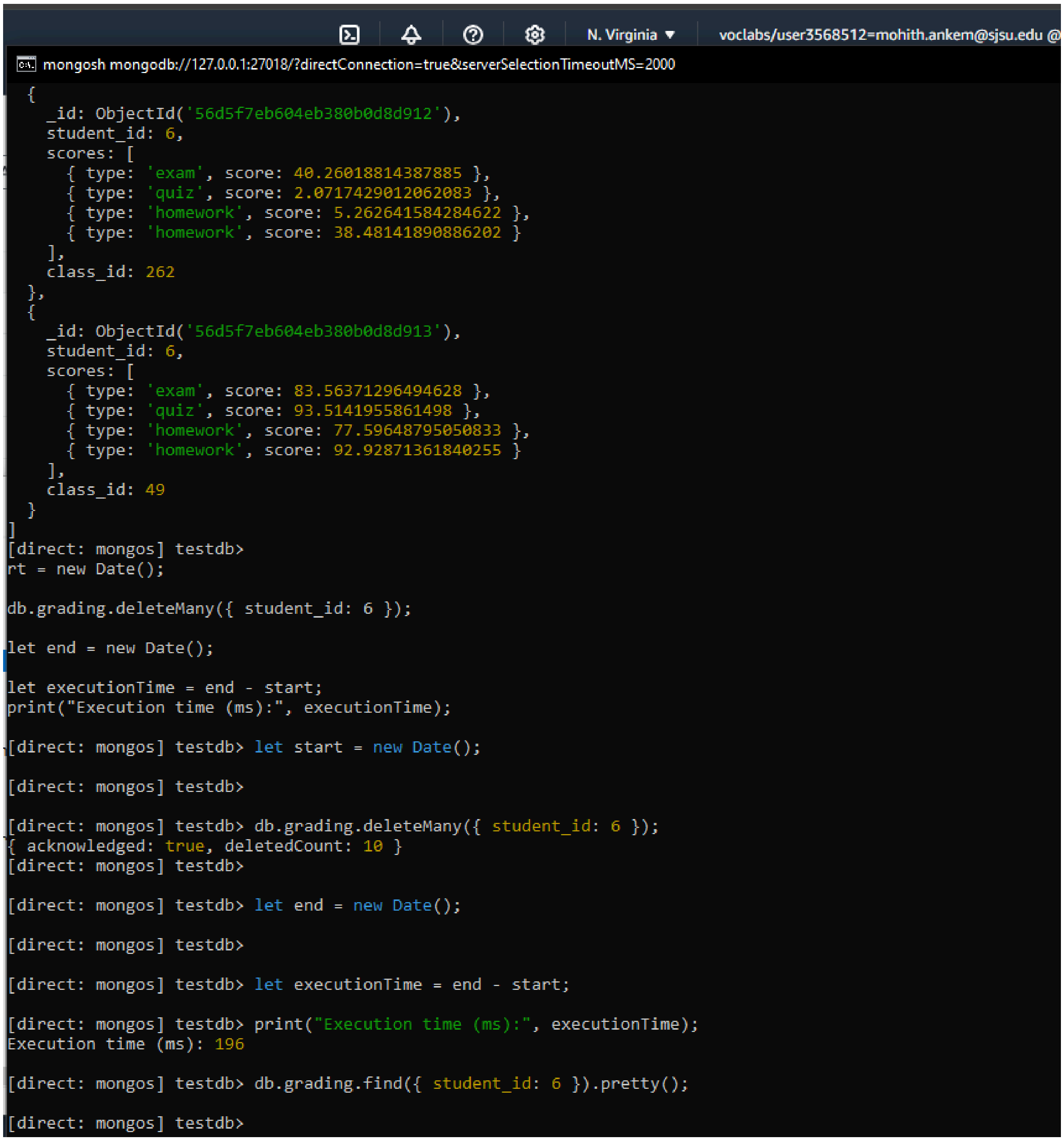
6. A delete

For Delete i have used **deleteMany()** Command to delete all the records of docs that have student_id as ‘6’ and have shown before and after deleting the records.

db.grading.deleteMany({ student_id: 6 });



```
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

[direct: mongos] testdb> db.grading.find({ student_id: 6 }).pretty();
[
  {
    _id: ObjectId('56d5f7eb604eb380b0d8d90a'),
    student_id: 6,
    scores: [
      { type: 'exam', score: 65.50992489612403 },
      { type: 'quiz', score: 91.60470236893907 },
      { type: 'homework', score: 68.85015284680942 },
      { type: 'homework', score: 11.494672763876224 }
    ],
    class_id: 347
  },
  {
    _id: ObjectId('56d5f7eb604eb380b0d8d90b'),
    student_id: 6,
    scores: [
      { type: 'exam', score: 65.49053111801578 },
      { type: 'quiz', score: 87.96299740726325 },
      { type: 'homework', score: 70.46688966172673 },
      { type: 'homework', score: 32.412761894229035 }
    ],
    class_id: 249
  },
  {
    _id: ObjectId('56d5f7eb604eb380b0d8d90c'),
    student_id: 6,
    scores: [
      { type: 'exam', score: 25.58068698556395 },
      { type: 'quiz', score: 18.50396281264115 },
      { type: 'homework', score: 29.033943529148686 },
      { type: 'homework', score: 96.64646214633883 }
    ],
    class_id: 296
  },
  {
    _id: ObjectId('56d5f7eb604eb380b0d8d90d'),
    student_id: 6,
    scores: [
      { type: 'exam', score: 51.4319929757575 },
      { type: 'quiz', score: 90.74412122063666 },
      { type: 'homework', score: 98.10952767153363 },
      { type: 'homework', score: 56.36641298253328 }
    ],
    class_id: 444
  },
  {
    _id: ObjectId('56d5f7eb604eb380b0d8d90e'),
    student_id: 6,
    scores: [
      { type: 'exam', score: 9.334813244535866 },
      { type: 'quiz', score: 19.864753815178226 },
```

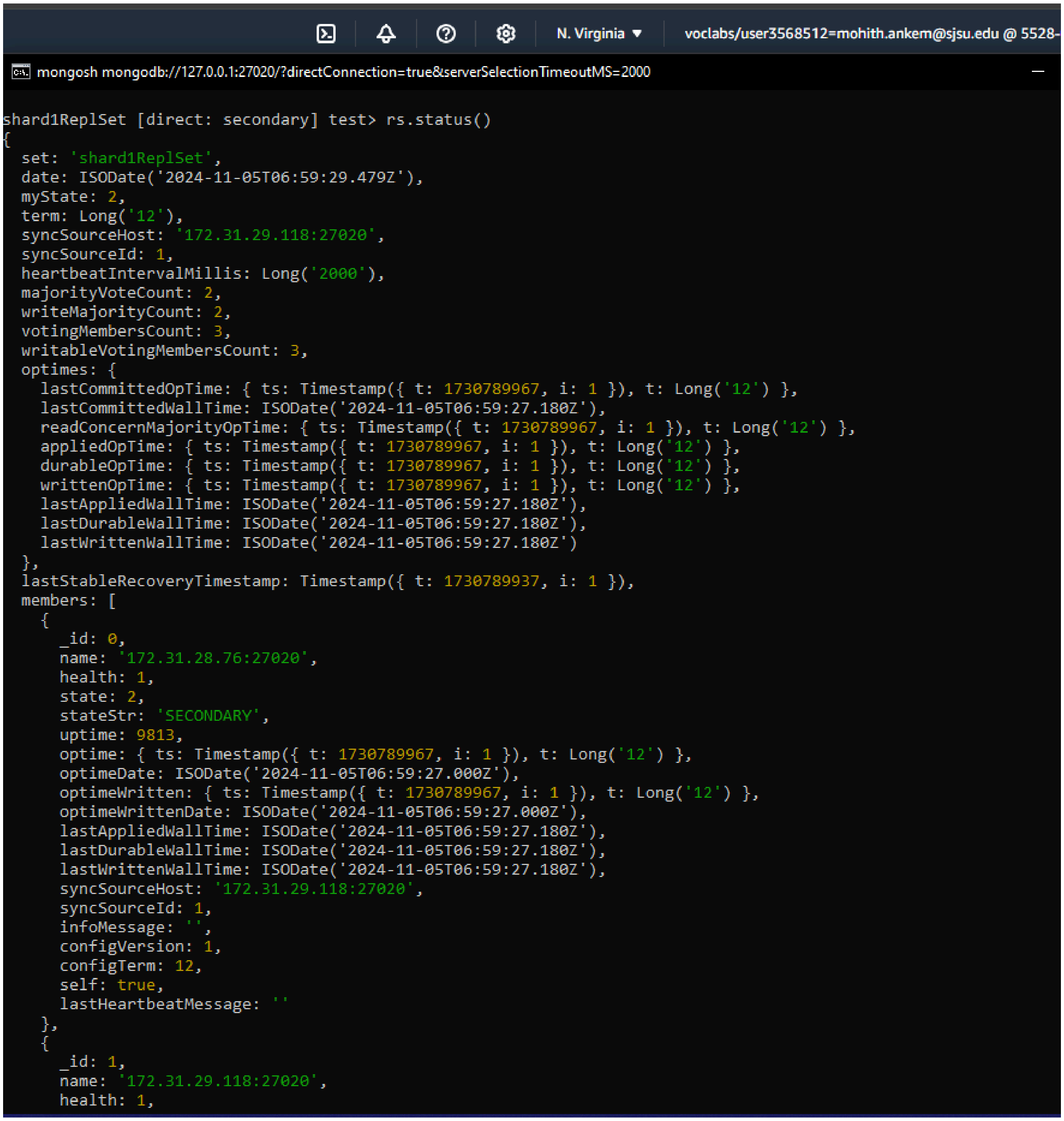

N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @
mongosh mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000

```
  {
    _id: ObjectId('56d5f7eb604eb380b0d8d912'),
    student_id: 6,
    scores: [
      { type: 'exam', score: 40.26018814387885 },
      { type: 'quiz', score: 2.0717429012062083 },
      { type: 'homework', score: 5.262641584284622 },
      { type: 'homework', score: 38.48141890886202 }
    ],
    class_id: 262
  },
  {
    _id: ObjectId('56d5f7eb604eb380b0d8d913'),
    student_id: 6,
    scores: [
      { type: 'exam', score: 83.56371296494628 },
      { type: 'quiz', score: 93.5141955861498 },
      { type: 'homework', score: 77.59648795050833 },
      { type: 'homework', score: 92.92871361840255 }
    ],
    class_id: 49
  }
]
[direct: mongos] testdb>
rt = new Date();

db.grading.deleteMany({ student_id: 6 });

let end = new Date();

let executionTime = end - start;
print("Execution time (ms):", executionTime);

[direct: mongos] testdb> let start = new Date();

[direct: mongos] testdb>

[direct: mongos] testdb> db.grading.deleteMany({ student_id: 6 });
{ acknowledged: true, deletedCount: 10 }
[direct: mongos] testdb>

[direct: mongos] testdb> let end = new Date();

[direct: mongos] testdb>

[direct: mongos] testdb> let executionTime = end - start;

[direct: mongos] testdb> print("Execution time (ms):", executionTime);
Execution time (ms): 196

[direct: mongos] testdb> db.grading.find({ student_id: 6 }).pretty();

[direct: mongos] testdb>
```

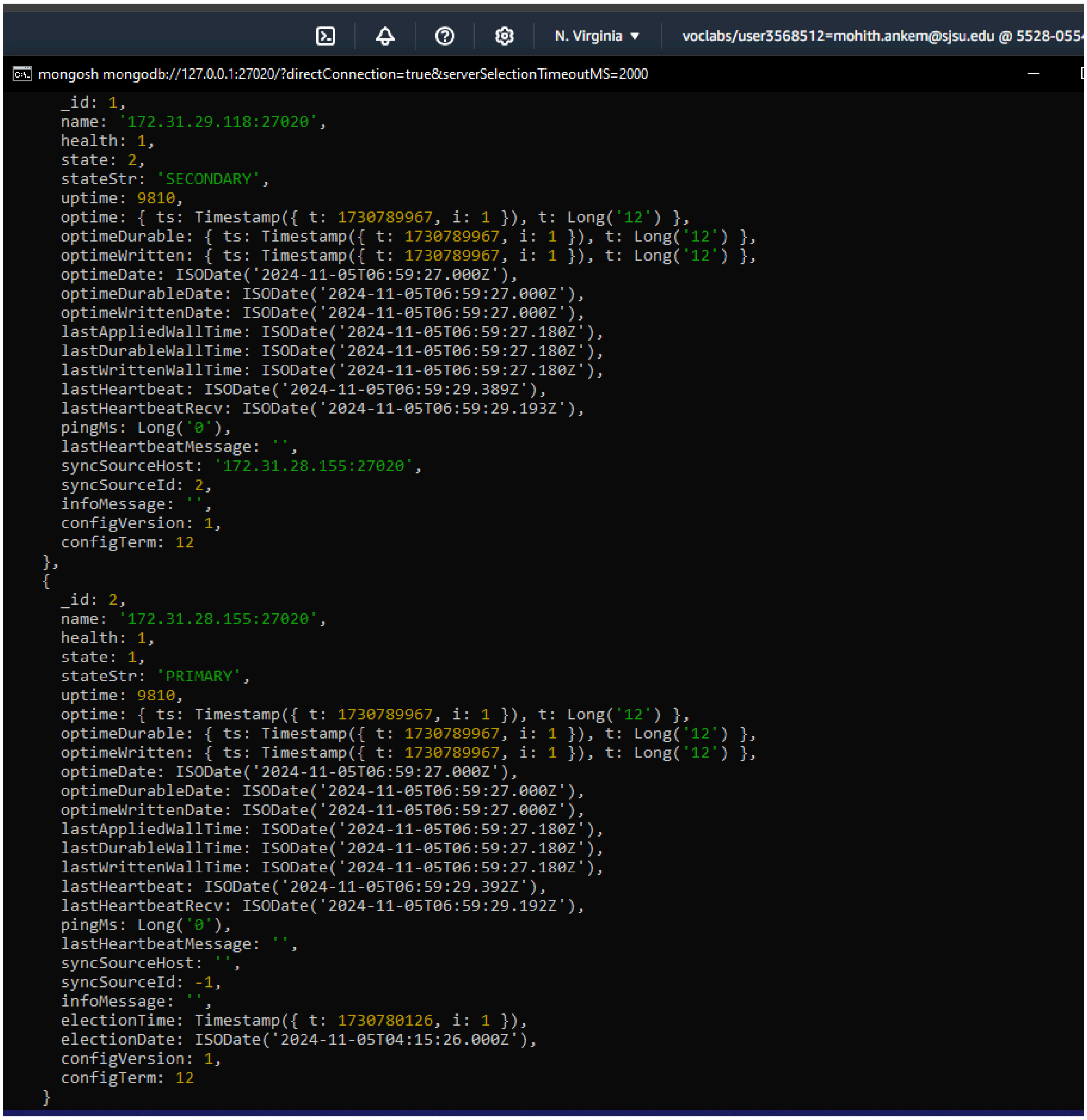
Deleted all the records that has student_id as ‘6’ and it took 196 ms to execute the query in which all 3 shards took part in this execution

13. Show shards are replicated using rs.status()

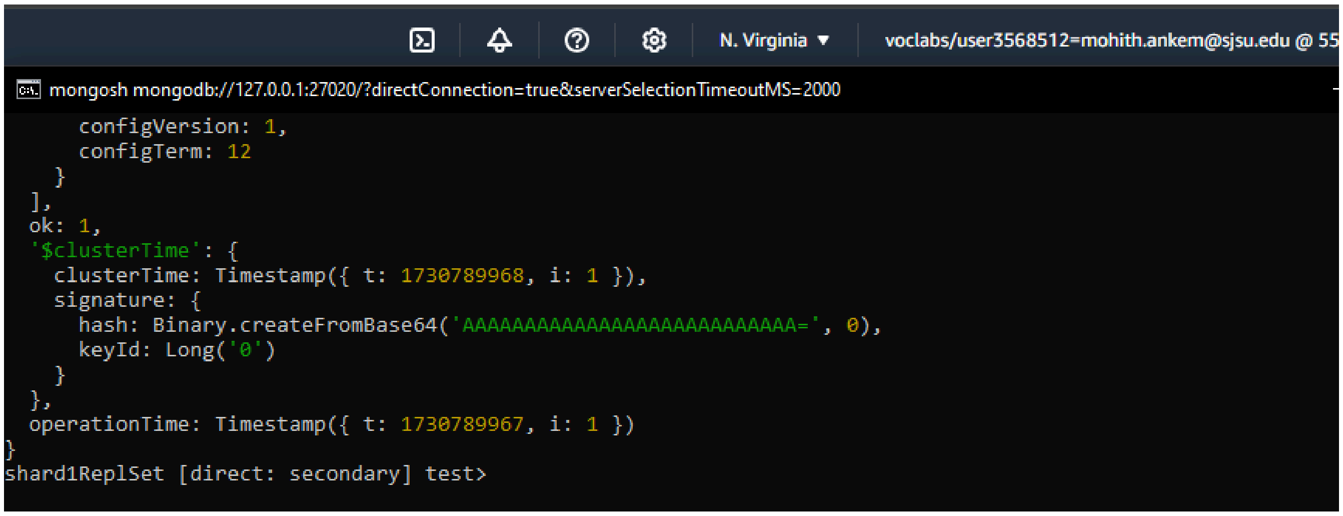
rs.status() for Shard1:


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-
mongosh mongodb://127.0.0.1:27020/?directConnection=true&serverSelectionTimeoutMS=2000

```
shard1ReplSet [direct: secondary] test> rs.status()
{
  set: 'shard1ReplSet',
  date: ISODate('2024-11-05T06:59:29.479Z'),
  myState: 2,
  term: Long('12'),
  syncSourceHost: '172.31.29.118:27020',
  syncSourceId: 1,
  heartbeatIntervalMillis: Long('2000'),
  majorityVoteCount: 2,
  writeMajorityCount: 2,
  votingMembersCount: 3,
  writableVotingMembersCount: 3,
  optimes: {
    lastCommittedOpTime: { ts: Timestamp({ t: 1730789967, i: 1 }), t: Long('12') },
    lastCommittedWallTime: ISODate('2024-11-05T06:59:27.180Z'),
    readConcernMajorityOpTime: { ts: Timestamp({ t: 1730789967, i: 1 }), t: Long('12') },
    appliedOpTime: { ts: Timestamp({ t: 1730789967, i: 1 }), t: Long('12') },
    durableOpTime: { ts: Timestamp({ t: 1730789967, i: 1 }), t: Long('12') },
    writtenOpTime: { ts: Timestamp({ t: 1730789967, i: 1 }), t: Long('12') },
    lastAppliedWallTime: ISODate('2024-11-05T06:59:27.180Z'),
    lastDurableWallTime: ISODate('2024-11-05T06:59:27.180Z'),
    lastWrittenWallTime: ISODate('2024-11-05T06:59:27.180Z')
  },
  lastStableRecoveryTimestamp: Timestamp({ t: 1730789937, i: 1 }),
  members: [
    {
      _id: 0,
      name: '172.31.28.76:27020',
      health: 1,
      state: 2,
      stateStr: 'SECONDARY',
      uptime: 9813,
      optime: { ts: Timestamp({ t: 1730789967, i: 1 }), t: Long('12') },
      optimeDate: ISODate('2024-11-05T06:59:27.000Z'),
      optimeWritten: { ts: Timestamp({ t: 1730789967, i: 1 }), t: Long('12') },
      optimeWrittenDate: ISODate('2024-11-05T06:59:27.000Z'),
      lastAppliedWallTime: ISODate('2024-11-05T06:59:27.180Z'),
      lastDurableWallTime: ISODate('2024-11-05T06:59:27.180Z'),
      lastWrittenWallTime: ISODate('2024-11-05T06:59:27.180Z'),
      syncSourceHost: '172.31.29.118:27020',
      syncSourceId: 1,
      infoMessage: '',
      configVersion: 1,
      configTerm: 12,
      self: true,
      lastHeartbeatMessage: ''
    },
    {
      _id: 1,
      name: '172.31.29.118:27020',
      health: 1,
```

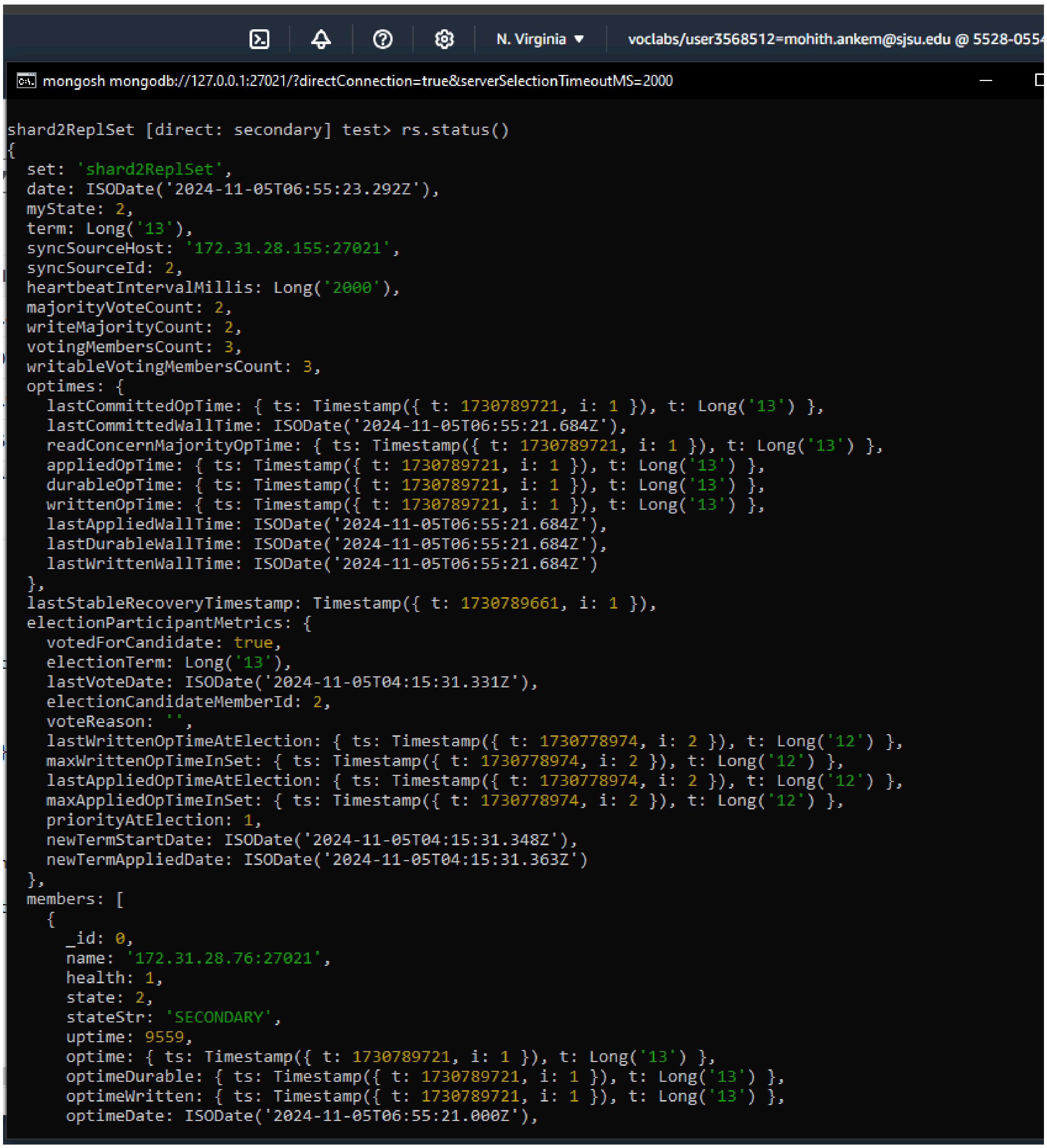

N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-055
mongosh mongodb://127.0.0.1:27020/?directConnection=true&serverSelectionTimeoutMS=2000

```
      _id: 1,
      name: '172.31.29.118:27020',
      health: 1,
      state: 2,
      stateStr: 'SECONDARY',
      uptime: 9810,
      optime: { ts: Timestamp({ t: 1730789967, i: 1 }), t: Long('12') },
      optimeDurable: { ts: Timestamp({ t: 1730789967, i: 1 }), t: Long('12') },
      optimeWritten: { ts: Timestamp({ t: 1730789967, i: 1 }), t: Long('12') },
      optimeDate: ISODate('2024-11-05T06:59:27.000Z'),
      optimeDurableDate: ISODate('2024-11-05T06:59:27.000Z'),
      optimeWrittenDate: ISODate('2024-11-05T06:59:27.000Z'),
      lastAppliedWallTime: ISODate('2024-11-05T06:59:27.180Z'),
      lastDurableWallTime: ISODate('2024-11-05T06:59:27.180Z'),
      lastWrittenWallTime: ISODate('2024-11-05T06:59:27.180Z'),
      lastHeartbeat: ISODate('2024-11-05T06:59:29.389Z'),
      lastHeartbeatRecv: ISODate('2024-11-05T06:59:29.193Z'),
      pingMs: Long('0'),
      lastHeartbeatMessage: '',
      syncSourceHost: '172.31.28.155:27020',
      syncSourceId: 2,
      infoMessage: '',
      configVersion: 1,
      configTerm: 12
    },
    {
      _id: 2,
      name: '172.31.28.155:27020',
      health: 1,
      state: 1,
      stateStr: 'PRIMARY',
      uptime: 9810,
      optime: { ts: Timestamp({ t: 1730789967, i: 1 }), t: Long('12') },
      optimeDurable: { ts: Timestamp({ t: 1730789967, i: 1 }), t: Long('12') },
      optimeWritten: { ts: Timestamp({ t: 1730789967, i: 1 }), t: Long('12') },
      optimeDate: ISODate('2024-11-05T06:59:27.000Z'),
      optimeDurableDate: ISODate('2024-11-05T06:59:27.000Z'),
      optimeWrittenDate: ISODate('2024-11-05T06:59:27.000Z'),
      lastAppliedWallTime: ISODate('2024-11-05T06:59:27.180Z'),
      lastDurableWallTime: ISODate('2024-11-05T06:59:27.180Z'),
      lastWrittenWallTime: ISODate('2024-11-05T06:59:27.180Z'),
      lastHeartbeat: ISODate('2024-11-05T06:59:29.392Z'),
      lastHeartbeatRecv: ISODate('2024-11-05T06:59:29.192Z'),
      pingMs: Long('0'),
      lastHeartbeatMessage: '',
      syncSourceHost: '',
      syncSourceId: -1,
      infoMessage: '',
      electionTime: Timestamp({ t: 1730780126, i: 1 }),
      electionDate: ISODate('2024-11-05T04:15:26.000Z'),
      configVersion: 1,
      configTerm: 12
    }
```

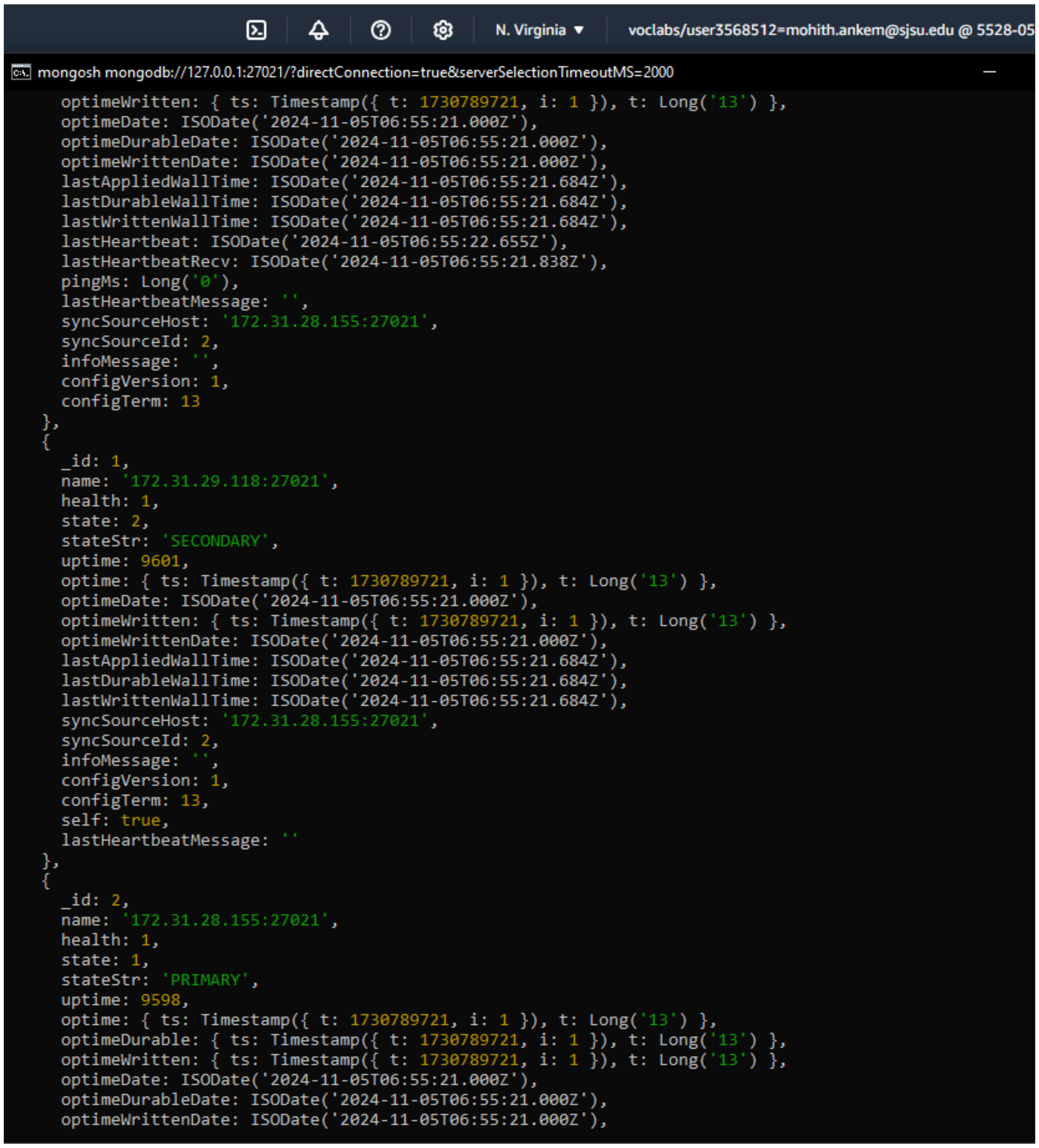

N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 55
mongosh mongodb://127.0.0.1:27020/?directConnection=true&serverSelectionTimeoutMS=2000

```
      configVersion: 1,
      configTerm: 12
    }
  ],
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1730789968, i: 1 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: Long('0')
    }
  },
  operationTime: Timestamp({ t: 1730789967, i: 1 })
}
shard1ReplSet [direct: secondary] test>
```

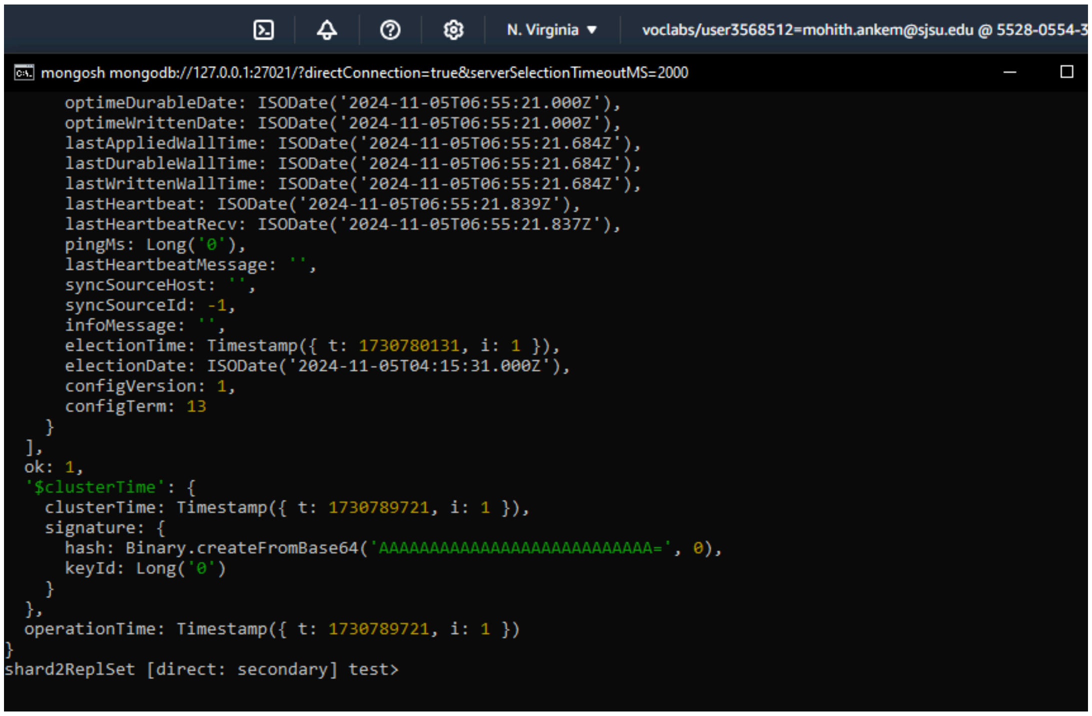
rs.status() for Shrad2:


N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-055
mongosh mongodb://127.0.0.1:27021/?directConnection=true&serverSelectionTimeoutMS=2000

```
shard2ReplSet [direct: secondary] test> rs.status()
{
  set: 'shard2ReplSet',
  date: ISODate('2024-11-05T06:55:23.292Z'),
  myState: 2,
  term: Long('13'),
  syncSourceHost: '172.31.28.155:27021',
  syncSourceId: 2,
  heartbeatIntervalMillis: Long('2000'),
  majorityVoteCount: 2,
  writeMajorityCount: 2,
  votingMembersCount: 3,
  writableVotingMembersCount: 3,
  optimes: {
    lastCommittedOpTime: { ts: Timestamp({ t: 1730789721, i: 1 }), t: Long('13') },
    lastCommittedWallTime: ISODate('2024-11-05T06:55:21.684Z'),
    readConcernMajorityOpTime: { ts: Timestamp({ t: 1730789721, i: 1 }), t: Long('13') },
    appliedOpTime: { ts: Timestamp({ t: 1730789721, i: 1 }), t: Long('13') },
    durableOpTime: { ts: Timestamp({ t: 1730789721, i: 1 }), t: Long('13') },
    writtenOpTime: { ts: Timestamp({ t: 1730789721, i: 1 }), t: Long('13') },
    lastAppliedWallTime: ISODate('2024-11-05T06:55:21.684Z'),
    lastDurableWallTime: ISODate('2024-11-05T06:55:21.684Z'),
    lastWrittenWallTime: ISODate('2024-11-05T06:55:21.684Z')
  },
  lastStableRecoveryTimestamp: Timestamp({ t: 1730789661, i: 1 }),
  electionParticipantMetrics: {
    votedForCandidate: true,
    electionTerm: Long('13'),
    lastVoteDate: ISODate('2024-11-05T04:15:31.331Z'),
    electionCandidateMemberId: 2,
    voteReason: '',
    lastWrittenOpTimeAtElection: { ts: Timestamp({ t: 1730778974, i: 2 }), t: Long('12') },
    maxWrittenOpTimeInSet: { ts: Timestamp({ t: 1730778974, i: 2 }), t: Long('12') },
    lastAppliedOpTimeAtElection: { ts: Timestamp({ t: 1730778974, i: 2 }), t: Long('12') },
    maxAppliedOpTimeInSet: { ts: Timestamp({ t: 1730778974, i: 2 }), t: Long('12') },
    priorityAtElection: 1,
    newTermStartDate: ISODate('2024-11-05T04:15:31.348Z'),
    newTermAppliedDate: ISODate('2024-11-05T04:15:31.363Z')
  },
  members: [
    {
      _id: 0,
      name: '172.31.28.76:27021',
      health: 1,
      state: 2,
      stateStr: 'SECONDARY',
      uptime: 9559,
      optime: { ts: Timestamp({ t: 1730789721, i: 1 }), t: Long('13') },
      optimeDurable: { ts: Timestamp({ t: 1730789721, i: 1 }), t: Long('13') },
      optimeWritten: { ts: Timestamp({ t: 1730789721, i: 1 }), t: Long('13') },
      optimeDate: ISODate('2024-11-05T06:55:21.000Z'),
```

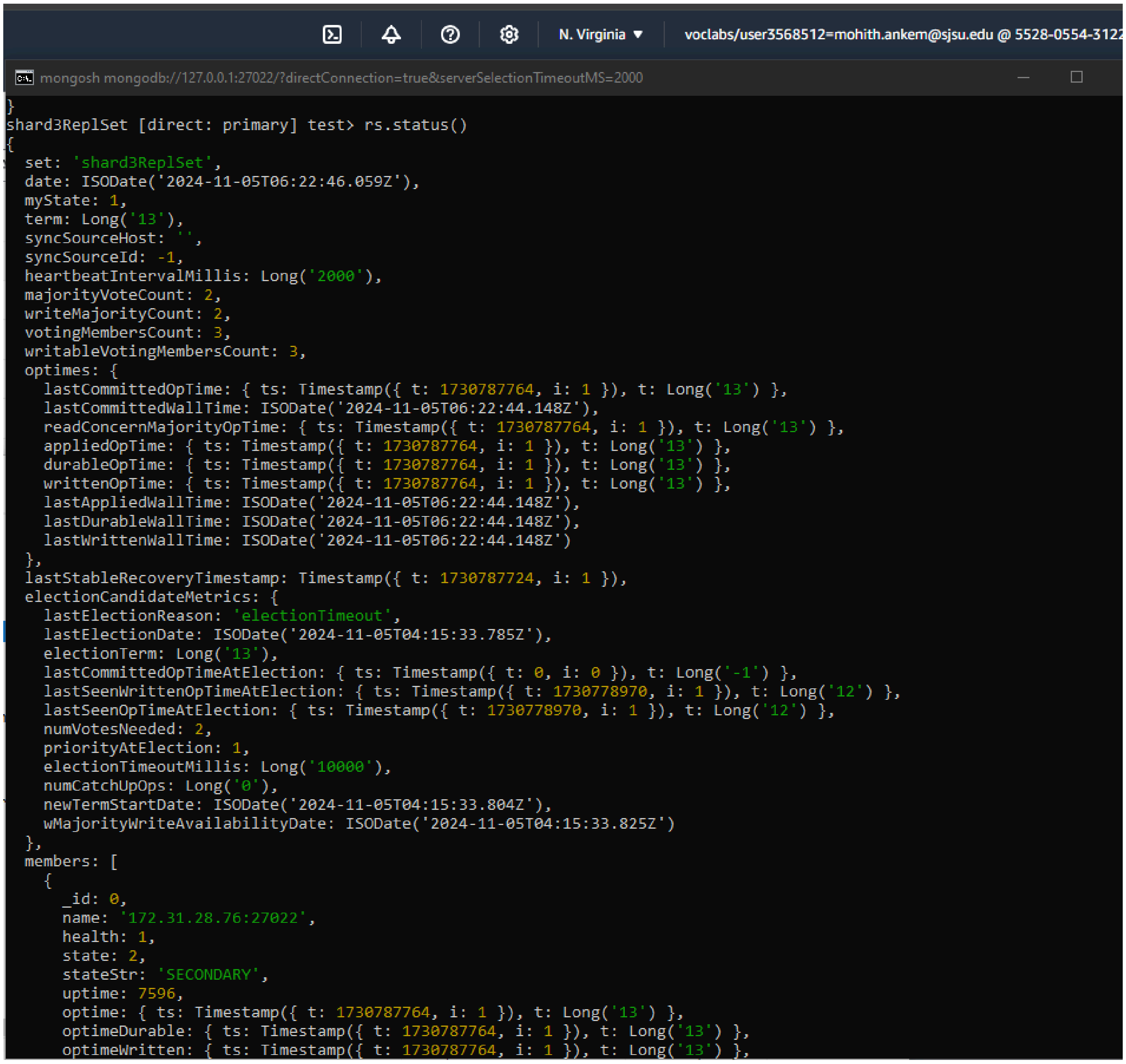

N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-05
mongosh mongodb://127.0.0.1:27021/?directConnection=true&serverSelectionTimeoutMS=2000

```
    optimeWritten: { ts: Timestamp({ t: 1730789721, i: 1 }), t: Long('13') },
    optimeDate: ISODate('2024-11-05T06:55:21.000Z'),
    optimeDurableDate: ISODate('2024-11-05T06:55:21.000Z'),
    optimeWrittenDate: ISODate('2024-11-05T06:55:21.000Z'),
    lastAppliedWallTime: ISODate('2024-11-05T06:55:21.684Z'),
    lastDurableWallTime: ISODate('2024-11-05T06:55:21.684Z'),
    lastWrittenWallTime: ISODate('2024-11-05T06:55:21.684Z'),
    lastHeartbeat: ISODate('2024-11-05T06:55:22.655Z'),
    lastHeartbeatRecv: ISODate('2024-11-05T06:55:21.838Z'),
    pingMs: Long('0'),
    lastHeartbeatMessage: '',
    syncSourceHost: '172.31.28.155:27021',
    syncSourceId: 2,
    infoMessage: '',
    configVersion: 1,
    configTerm: 13
  },
  {
    _id: 1,
    name: '172.31.29.118:27021',
    health: 1,
    state: 2,
    stateStr: 'SECONDARY',
    uptime: 9601,
    optime: { ts: Timestamp({ t: 1730789721, i: 1 }), t: Long('13') },
    optimeDate: ISODate('2024-11-05T06:55:21.000Z'),
    optimeWritten: { ts: Timestamp({ t: 1730789721, i: 1 }), t: Long('13') },
    optimeWrittenDate: ISODate('2024-11-05T06:55:21.000Z'),
    lastAppliedWallTime: ISODate('2024-11-05T06:55:21.684Z'),
    lastDurableWallTime: ISODate('2024-11-05T06:55:21.684Z'),
    lastWrittenWallTime: ISODate('2024-11-05T06:55:21.684Z'),
    syncSourceHost: '172.31.28.155:27021',
    syncSourceId: 2,
    infoMessage: '',
    configVersion: 1,
    configTerm: 13,
    self: true,
    lastHeartbeatMessage: ''
  },
  {
    _id: 2,
    name: '172.31.28.155:27021',
    health: 1,
    state: 1,
    stateStr: 'PRIMARY',
    uptime: 9598,
    optime: { ts: Timestamp({ t: 1730789721, i: 1 }), t: Long('13') },
    optimeDurable: { ts: Timestamp({ t: 1730789721, i: 1 }), t: Long('13') },
    optimeWritten: { ts: Timestamp({ t: 1730789721, i: 1 }), t: Long('13') },
    optimeDate: ISODate('2024-11-05T06:55:21.000Z'),
    optimeDurableDate: ISODate('2024-11-05T06:55:21.000Z'),
    optimeWrittenDate: ISODate('2024-11-05T06:55:21.000Z'),
```

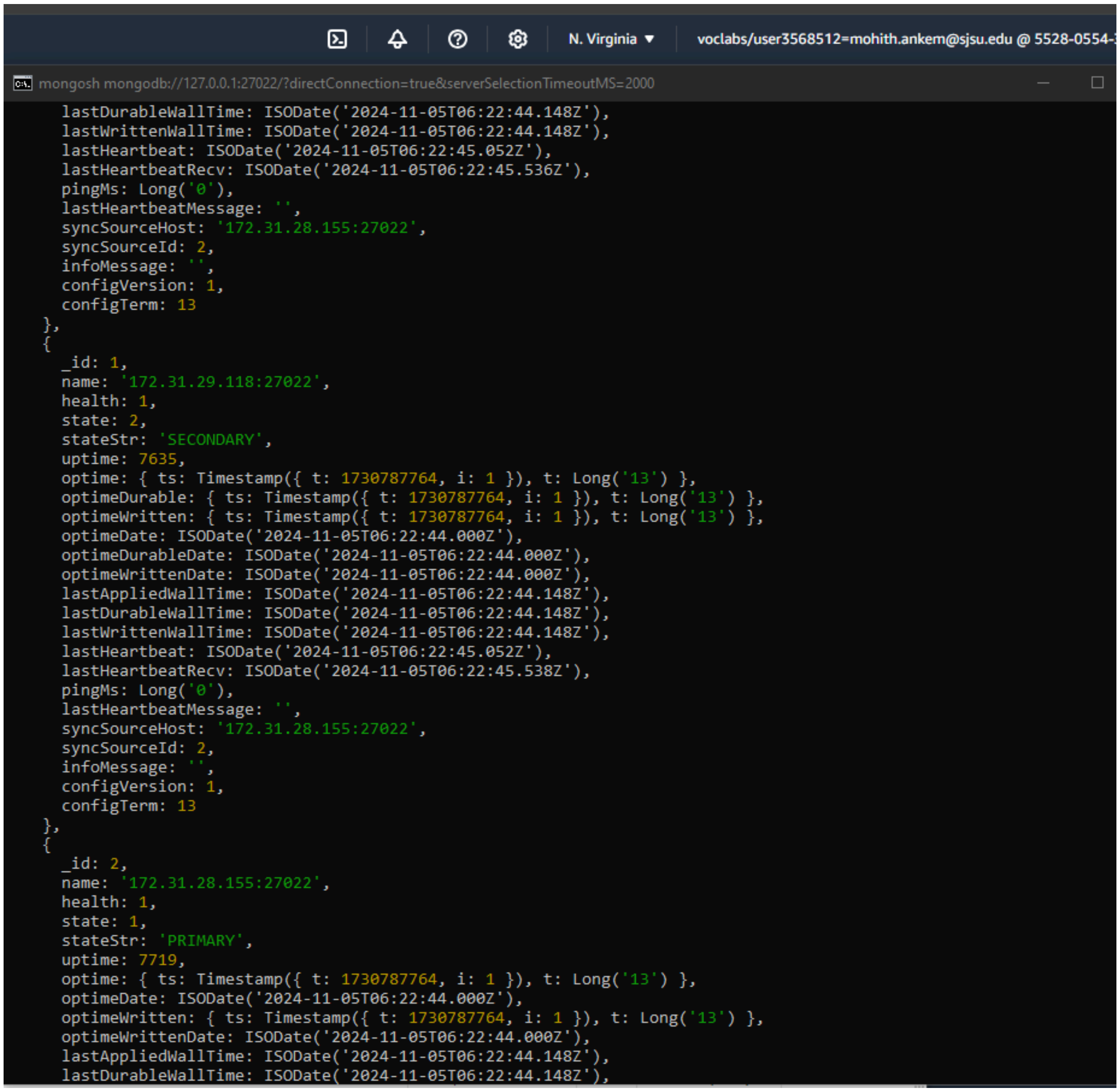

N. Virginia
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3
mongosh mongodb://127.0.0.1:27021/?directConnection=true&serverSelectionTimeoutMS=2000

```
    optimeDurableDate: ISODate('2024-11-05T06:55:21.000Z'),
    optimeWrittenDate: ISODate('2024-11-05T06:55:21.000Z'),
    lastAppliedWallTime: ISODate('2024-11-05T06:55:21.684Z'),
    lastDurableWallTime: ISODate('2024-11-05T06:55:21.684Z'),
    lastWrittenWallTime: ISODate('2024-11-05T06:55:21.684Z'),
    lastHeartbeat: ISODate('2024-11-05T06:55:21.839Z'),
    lastHeartbeatRecv: ISODate('2024-11-05T06:55:21.837Z'),
    pingMs: Long('0'),
    lastHeartbeatMessage: '',
    syncSourceHost: '',
    syncSourceId: -1,
    infoMessage: '',
    electionTime: Timestamp({ t: 1730780131, i: 1 }),
    electionDate: ISODate('2024-11-05T04:15:31.000Z'),
    configVersion: 1,
    configTerm: 13
  }
],
ok: 1,
'$clusterTime': {
  clusterTime: Timestamp({ t: 1730789721, i: 1 }),
  signature: {
    hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
    keyId: Long('0')
  }
},
operationTime: Timestamp({ t: 1730789721, i: 1 })
}
shard2ReplSet [direct: secondary] test>
```

rs.status() for shard3:


N. Virginia ▼
voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-3122
mongosh mongodb://127.0.0.1:27022/?directConnection=true&serverSelectionTimeoutMS=2000

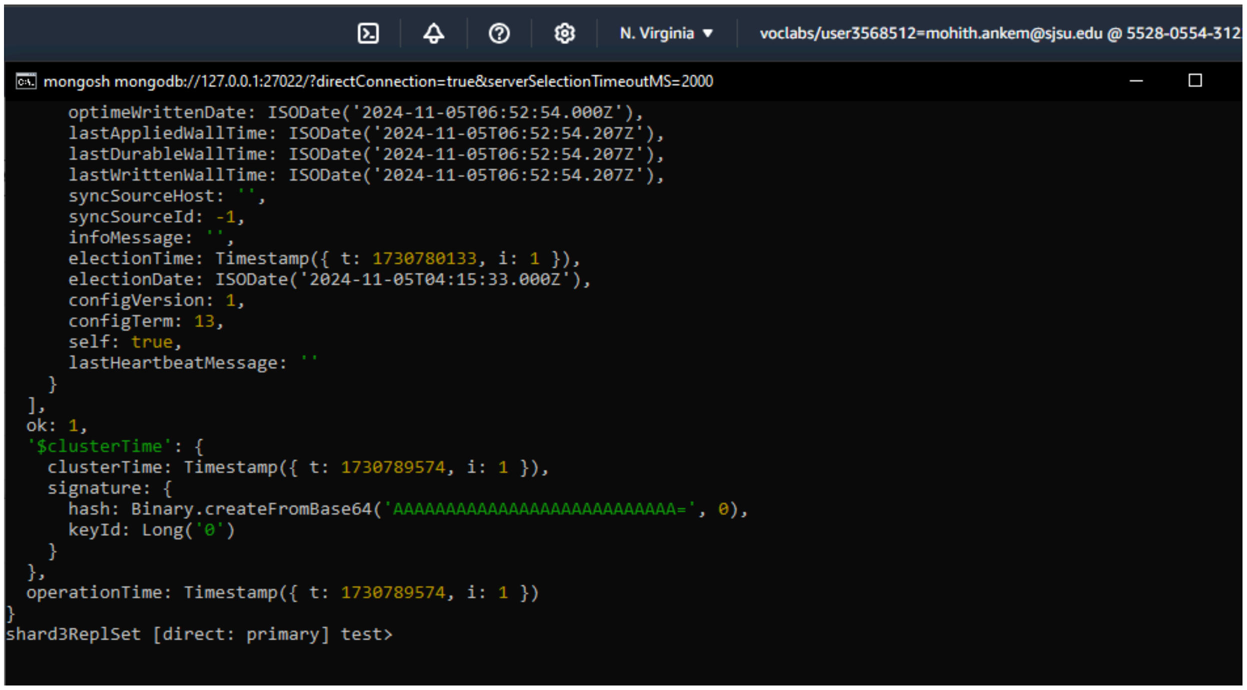
```
}
shard3ReplSet [direct: primary] test> rs.status()
{
  set: 'shard3ReplSet',
  date: ISODate('2024-11-05T06:22:46.059Z'),
  myState: 1,
  term: Long('13'),
  syncSourceHost: '',
  syncSourceId: -1,
  heartbeatIntervalMillis: Long('2000'),
  majorityVoteCount: 2,
  writeMajorityCount: 2,
  votingMembersCount: 3,
  writableVotingMembersCount: 3,
  optimes: {
    lastCommittedOpTime: { ts: Timestamp({ t: 1730787764, i: 1 }), t: Long('13') },
    lastCommittedWallTime: ISODate('2024-11-05T06:22:44.148Z'),
    readConcernMajorityOpTime: { ts: Timestamp({ t: 1730787764, i: 1 }), t: Long('13') },
    appliedOpTime: { ts: Timestamp({ t: 1730787764, i: 1 }), t: Long('13') },
    durableOpTime: { ts: Timestamp({ t: 1730787764, i: 1 }), t: Long('13') },
    writtenOpTime: { ts: Timestamp({ t: 1730787764, i: 1 }), t: Long('13') },
    lastAppliedWallTime: ISODate('2024-11-05T06:22:44.148Z'),
    lastDurableWallTime: ISODate('2024-11-05T06:22:44.148Z'),
    lastWrittenWallTime: ISODate('2024-11-05T06:22:44.148Z')
  },
  lastStableRecoveryTimestamp: Timestamp({ t: 1730787724, i: 1 }),
  electionCandidateMetrics: {
    lastElectionReason: 'electionTimeout',
    lastElectionDate: ISODate('2024-11-05T04:15:33.785Z'),
    electionTerm: Long('13'),
    lastCommittedOpTimeAtElection: { ts: Timestamp({ t: 0, i: 0 }), t: Long('-1') },
    lastSeenWrittenOpTimeAtElection: { ts: Timestamp({ t: 1730778970, i: 1 }), t: Long('12') },
    lastSeenOpTimeAtElection: { ts: Timestamp({ t: 1730778970, i: 1 }), t: Long('12') },
    numVotesNeeded: 2,
    priorityAtElection: 1,
    electionTimeoutMillis: Long('10000'),
    numCatchUpOps: Long('0'),
    newTermStartDate: ISODate('2024-11-05T04:15:33.804Z'),
    wMajorityWriteAvailabilityDate: ISODate('2024-11-05T04:15:33.825Z')
  },
  members: [
    {
      _id: 0,
      name: '172.31.28.76:27022',
      health: 1,
      state: 2,
      stateStr: 'SECONDARY',
      uptime: 7596,
      optime: { ts: Timestamp({ t: 1730787764, i: 1 }), t: Long('13') },
      optimeDurable: { ts: Timestamp({ t: 1730787764, i: 1 }), t: Long('13') },
      optimeWritten: { ts: Timestamp({ t: 1730787764, i: 1 }), t: Long('13') },
```



voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-

mongosh mongodb://127.0.0.1:27022/?directConnection=true&serverSelectionTimeoutMS=2000

```
      lastDurableWallTime: ISODate('2024-11-05T06:22:44.148Z'),
      lastWrittenWallTime: ISODate('2024-11-05T06:22:44.148Z'),
      lastHeartbeat: ISODate('2024-11-05T06:22:45.052Z'),
      lastHeartbeatRecv: ISODate('2024-11-05T06:22:45.536Z'),
      pingMs: Long('0'),
      lastHeartbeatMessage: '',
      syncSourceHost: '172.31.28.155:27022',
      syncSourceId: 2,
      infoMessage: '',
      configVersion: 1,
      configTerm: 13
    },
    {
      _id: 1,
      name: '172.31.29.118:27022',
      health: 1,
      state: 2,
      stateStr: 'SECONDARY',
      uptime: 7635,
      optime: { ts: Timestamp({ t: 1730787764, i: 1 }), t: Long('13') },
      optimeDurable: { ts: Timestamp({ t: 1730787764, i: 1 }), t: Long('13') },
      optimeWritten: { ts: Timestamp({ t: 1730787764, i: 1 }), t: Long('13') },
      optimeDate: ISODate('2024-11-05T06:22:44.000Z'),
      optimeDurableDate: ISODate('2024-11-05T06:22:44.000Z'),
      optimeWrittenDate: ISODate('2024-11-05T06:22:44.000Z'),
      lastAppliedWallTime: ISODate('2024-11-05T06:22:44.148Z'),
      lastDurableWallTime: ISODate('2024-11-05T06:22:44.148Z'),
      lastWrittenWallTime: ISODate('2024-11-05T06:22:44.148Z'),
      lastHeartbeat: ISODate('2024-11-05T06:22:45.052Z'),
      lastHeartbeatRecv: ISODate('2024-11-05T06:22:45.538Z'),
      pingMs: Long('0'),
      lastHeartbeatMessage: '',
      syncSourceHost: '172.31.28.155:27022',
      syncSourceId: 2,
      infoMessage: '',
      configVersion: 1,
      configTerm: 13
    },
    {
      _id: 2,
      name: '172.31.28.155:27022',
      health: 1,
      state: 1,
      stateStr: 'PRIMARY',
      uptime: 7719,
      optime: { ts: Timestamp({ t: 1730787764, i: 1 }), t: Long('13') },
      optimeDate: ISODate('2024-11-05T06:22:44.000Z'),
      optimeWritten: { ts: Timestamp({ t: 1730787764, i: 1 }), t: Long('13') },
      optimeWrittenDate: ISODate('2024-11-05T06:22:44.000Z'),
      lastAppliedWallTime: ISODate('2024-11-05T06:22:44.148Z'),
      lastDurableWallTime: ISODate('2024-11-05T06:22:44.148Z'),
```



voclabs/user3568512=mohith.ankem@sjsu.edu @ 5528-0554-312

mongosh mongodb://127.0.0.1:27022/?directConnection=true&serverSelectionTimeoutMS=2000

```
      optimeWrittenDate: ISODate('2024-11-05T06:52:54.000Z'),
      lastAppliedWallTime: ISODate('2024-11-05T06:52:54.207Z'),
      lastDurableWallTime: ISODate('2024-11-05T06:52:54.207Z'),
      lastWrittenWallTime: ISODate('2024-11-05T06:52:54.207Z'),
      syncSourceHost: '',
      syncSourceId: -1,
      infoMessage: '',
      electionTime: Timestamp({ t: 1730780133, i: 1 }),
      electionDate: ISODate('2024-11-05T04:15:33.000Z'),
      configVersion: 1,
      configTerm: 13,
      self: true,
      lastHeartbeatMessage: ''
    }
  ],
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1730789574, i: 1 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: Long('0')
    }
  },
  operationTime: Timestamp({ t: 1730789574, i: 1 })
}
shard3ReplSet [direct: primary] test>
```

14. <u>List of hosts and describe what are deployed in each host. For example,</u>

    node0:port# config server PRIMARY

    node1:port# config server SECONDARY

    …..

Ans:

node:port	**Description**
Node1_Instance1 : 27019	Config Server Primary
Node2_Server2 : 27019	Config Server Secondary
Node3_Server3 :27019	Config Server Secondary
Node4_Server4 : 27018 (mongos router)	Mongos Server
Node5_Server5 : 27020	Shard Server 1
Node6_Server6 : 27021	Shard Server 2
Node7_Server7 : 27022	Shard Server 3