CSP 554

MongoDB

Ashmita Gupta

CSP554—Big Data Technologies

Assignment #13

Worth: 5 points ALL EXTRA CREDIT

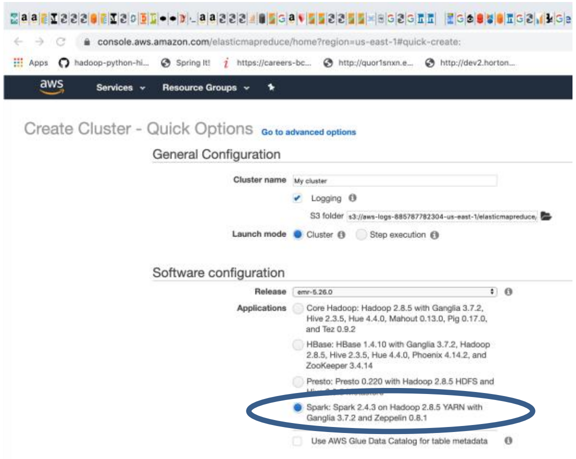
Due at the time you submit your final project or paper

Assignments should be uploaded via the Blackboard portal.

Set Up:

Step A – Start an EMR cluster

Start up an EMR/Hadoop cluster as previously, but instead of choosing the "Core Hadoop" configuration chose the "Spark" configuration (see below), otherwise proceed as before.



console.aws.amazon.com/elasticmapreduce/home?region=us-east-1#quick-create:

Create Cluster - Quick Options

General Configuration

Cluster name: My cluster

☑ Logging

S3 folder: s3://aws-logs-885787782304-us-east-1/elasticmapreduce

Launch mode: ● Cluster ○ Step execution

Software configuration

Release: emr-5.26.0

Applications:

○ Core Hadoop: Hadoop 2.8.5 with Ganglia 3.7.2, Hive 2.3.5, Hue 4.4.0, Mahout 0.13.0, Pig 0.17.0, and Tez 0.9.2

○ HBase: HBase 1.4.10 with Ganglia 3.7.2, Hadoop 2.8.5, Hive 2.3.5, Hue 4.4.0, Phoenix 4.14.2, and ZooKeeper 3.4.14

○ Presto: Presto 0.220 with Hadoop 2.8.5 HDFS and

● Spark: Spark 2.4.3 on Hadoop 2.8.5 YARN with Ganglia 3.7.2 and Zeppelin 0.8.1

□ Use AWS Glue Data Catalog for table metadata

Step B – Download the assignment software (mongoex.tar, mongodb-org-4.2.repo) to master node

Download "mongoex.tar" (included as a file with the assignment) to your PC or MAC. Now, using "scp" copy this file to the EMR master node using something like the following (just an example):

```
scp -i ./emr-key-pair-2.cer /Users/nachdaph/csp554-fall-2021/assignments/mongoex.tar
hadoop@ec2-44-199-215-205.compute-1.amazonaws.com:/home/hadoop
```

Now download "mongodb-org-4.2.repo" (included as a file with the assignment) to your PC or MAC. Now, using "scp" copy this file to the EMR master node using something like the following (just an example):

```
scp -i ./emr-key-pair-2.cer /Users/nachdaph/csp554-fall-2021/assignments/mongodb-org-
4.2.repo hadoop@ec2-44-199-215-205.compute-1.amazonaws.com:/home/hadoop
```

Step C – Install assignment software (mongoex.zip, mongodb-org-4.2.repo)

Enter the following into a terminal window which you have connected to the EMR master node. Going forward we will call this terminal connection Init-Term:

```
sudo cp mongodb-org-4.2.repo /etc/yum.repos.d
```

Then enter this into Init-Term to unzip mongoex.tar:

```
tar -xvf mongoex.tar
```

Step D – Install and start MongoDB

Enter the following into Init-Term to install MongoDB:

```
sudo yum install -y mongodb-org-4.2.15 mongodb-org-server-4.2.15 mongodb-org-shell-4.2.15
mongodb-org-mongos-4.2.15 mongodb-org-tools-4.2.15
```

Now enter this into Init-Term to start mongodb:

```
sudo systemctl start mongod
```

Step E – Start the MongoDB Shell (Command Line Interpreter)

Open a second terminal connection to the EMR master node. Going forward we will call this terminal connection: CLI-Term.

You will use this terminal window to start and run the mongodb shell as follows:

```
mongo
```

Step F – Edit mongo query language files

Open a third terminal connection to the EMR master node. Going forward we will call this terminal connection: CLI-Term. You will use this terminal window to run the 'vi' editor to create your Mongo code files.

As an alternative you could edit your MongoDB code files on your PC/MAC and then 'scp' them to the EMR mater node.

Step G – Setting up the assignment database

Now, in the MongoDB shell, using the CLI-Term, create a database called "assignment" by entering the following into the MongoDB shell:

```
    use assignment;
```

This will set the shell variable 'db' to this new database.

Load a collection called ‘unicorns’ with sample data by executing the script load.js in the MongoDB shell as follows (don’t cut and paste this, type it in manually):

```
load(‘./load.js’);
```

Note, look at the content of the script file (via the other terminal window you have opened to the EC2 instance) to see how each unicorn is described.

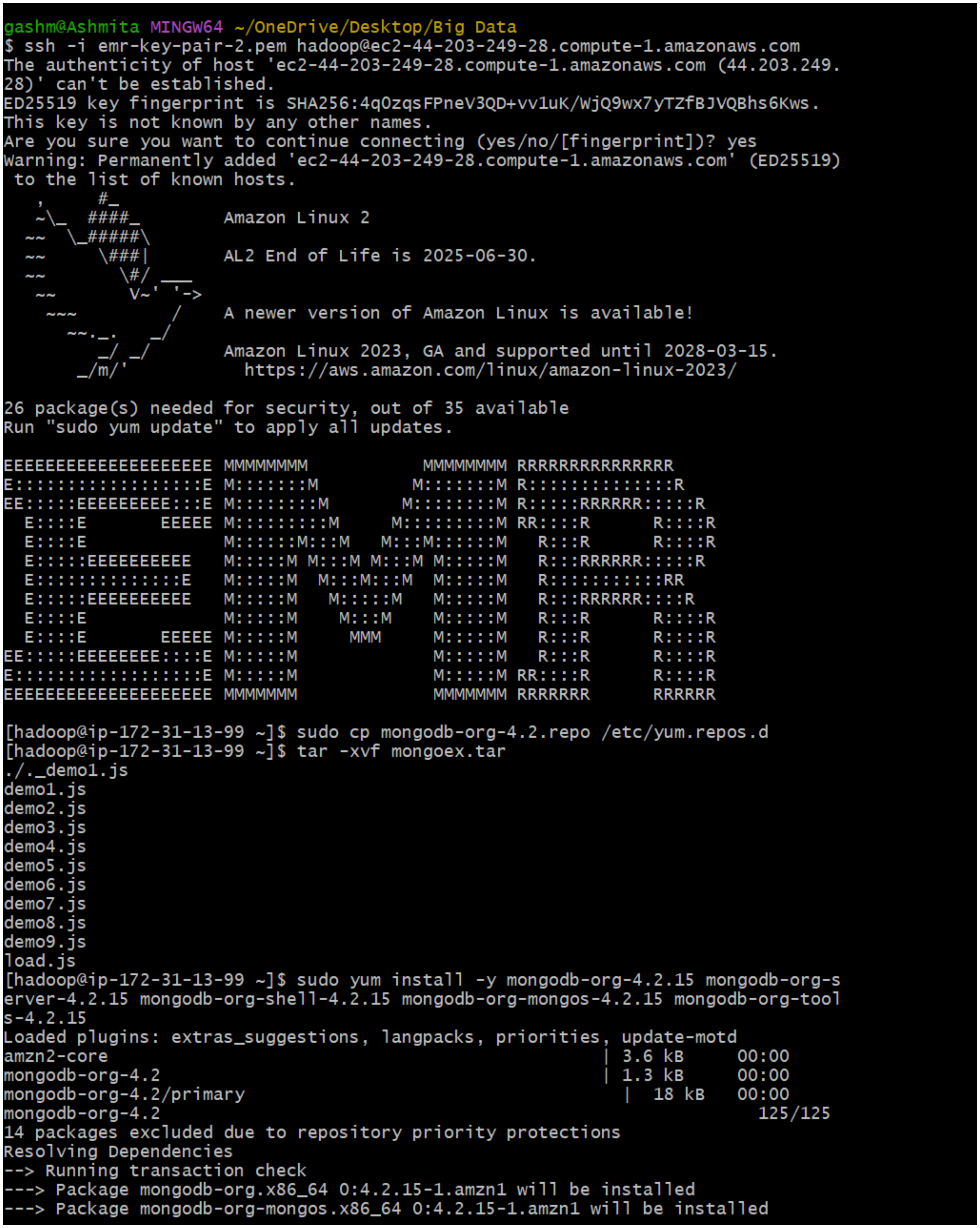
Confirm this has all worked by executing the following command in the MongoDB shell:

```
db.unicorns.find();
```

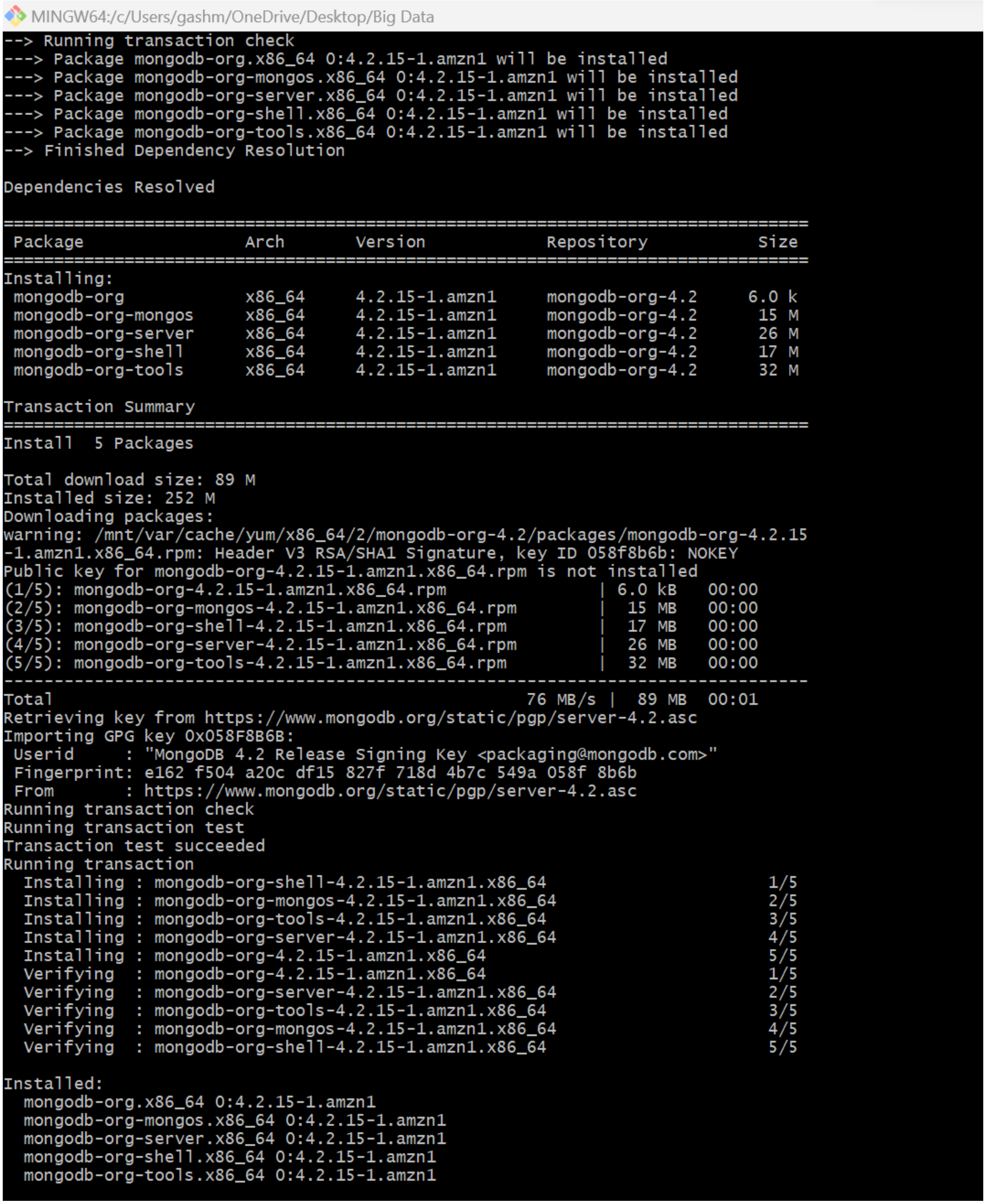
Note, the files named “demo*.js” (also included in the mongoex.tar file) provide examples of how to operate in the unicorn collection. These are a VERY good idea to review and understand and will present you with information helpful in completing the assignment. Also, try them out by typing something like

```
load(./demo1.js’);
```

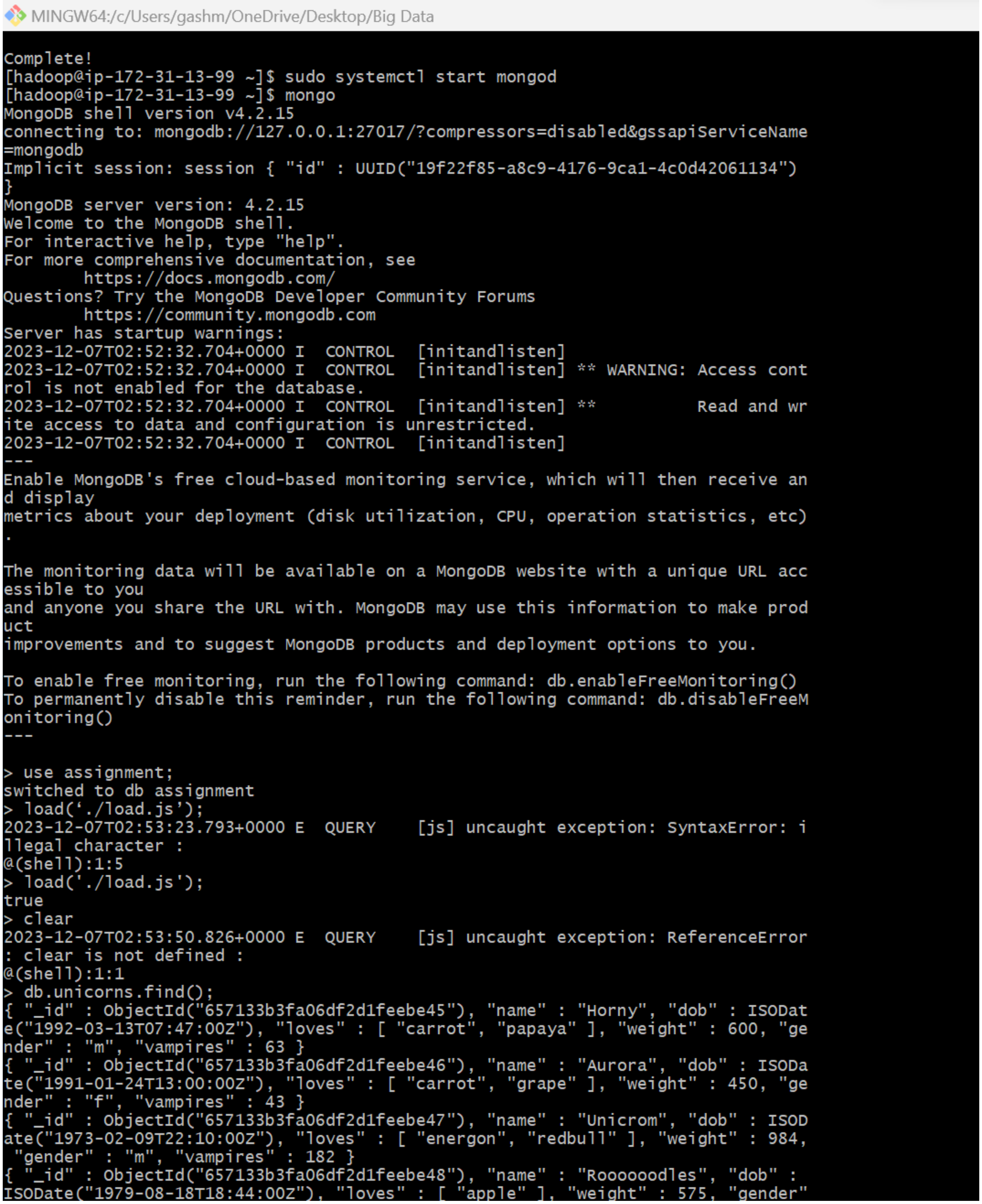
Below are the steps performed:



```
gashm@Ashmita MINGW64 ~/OneDrive/Desktop/Big Data
$ ssh -i emr-key-pair-2.pem hadoop@ec2-44-203-249-28.compute-1.amazonaws.com
The authenticity of host 'ec2-44-203-249-28.compute-1.amazonaws.com (44.203.249.
28)' can't be established.
ED25519 key fingerprint is SHA256:4q0zqsFPneV3QD+vv1uK/WjQ9wx7yTZfBJVQBhs6Kws.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-44-203-249-28.compute-1.amazonaws.com' (ED25519)
 to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2025-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

26 package(s) needed for security, out of 35 available
Run "sudo yum update" to apply all updates.

EEEEEEEEEEEEEEEEEEEE MMMMMMMM           MMMMMMMM RRRRRRRRRRRRRRR
E::::::::::::::::::E M:::::::M         M:::::::M R::::::::::::::R
EE:::::EEEEEEEEE:::E M::::::::M       M::::::::M R:::::RRRRRR:::::R
  E::::E       EEEEE M:::::::::M     M:::::::::M RR::::R      R::::R
  E::::E             M::::::M:::M   M:::M::::::M   R:::R      R::::R
  E:::::EEEEEEEEEE   M:::::M M:::M M:::M M:::::M   R:::RRRRRR:::::R
  E::::::::::::::E   M:::::M  M:::M:::M  M:::::M   R:::::::::::RR
  E:::::EEEEEEEEEE   M:::::M   M:::::M   M:::::M   R:::RRRRRR::::R
  E::::E             M:::::M    M:::M    M:::::M   R:::R      R::::R
  E::::E       EEEEE M:::::M     MMM     M:::::M   R:::R      R::::R
EE:::::EEEEEEEE::::E M:::::M             M:::::M   R:::R      R::::R
E::::::::::::::::::E M:::::M             M:::::M RR::::R      R::::R
EEEEEEEEEEEEEEEEEEEE MMMMMMM             MMMMMMM RRRRRRR      RRRRRR

[hadoop@ip-172-31-13-99 ~]$ sudo cp mongodb-org-4.2.repo /etc/yum.repos.d
[hadoop@ip-172-31-13-99 ~]$ tar -xvf mongoex.tar
./._demo1.js
demo1.js
demo2.js
demo3.js
demo4.js
demo5.js
demo6.js
demo7.js
demo8.js
demo9.js
load.js
[hadoop@ip-172-31-13-99 ~]$ sudo yum install -y mongodb-org-4.2.15 mongodb-org-s
erver-4.2.15 mongodb-org-shell-4.2.15 mongodb-org-mongos-4.2.15 mongodb-org-tool
s-4.2.15
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                               | 3.6 kB     00:00
mongodb-org-4.2                                          | 1.3 kB     00:00
mongodb-org-4.2/primary                                    |  18 kB   00:00
mongodb-org-4.2                                                           125/125
14 packages excluded due to repository priority protections
Resolving Dependencies
--> Running transaction check
---> Package mongodb-org.x86_64 0:4.2.15-1.amzn1 will be installed
---> Package mongodb-org-mongos.x86_64 0:4.2.15-1.amzn1 will be installed
```



```
--> Running transaction check
---> Package mongodb-org.x86_64 0:4.2.15-1.amzn1 will be installed
---> Package mongodb-org-mongos.x86_64 0:4.2.15-1.amzn1 will be installed
---> Package mongodb-org-server.x86_64 0:4.2.15-1.amzn1 will be installed
---> Package mongodb-org-shell.x86_64 0:4.2.15-1.amzn1 will be installed
---> Package mongodb-org-tools.x86_64 0:4.2.15-1.amzn1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package                Arch       Version            Repository             Size
================================================================================
Installing:
 mongodb-org            x86_64     4.2.15-1.amzn1     mongodb-org-4.2       6.0 k
 mongodb-org-mongos     x86_64     4.2.15-1.amzn1     mongodb-org-4.2        15 M
 mongodb-org-server     x86_64     4.2.15-1.amzn1     mongodb-org-4.2        26 M
 mongodb-org-shell      x86_64     4.2.15-1.amzn1     mongodb-org-4.2        17 M
 mongodb-org-tools      x86_64     4.2.15-1.amzn1     mongodb-org-4.2        32 M

Transaction Summary
================================================================================
Install  5 Packages

Total download size: 89 M
Installed size: 252 M
Downloading packages:
warning: /mnt/var/cache/yum/x86_64/2/mongodb-org-4.2/packages/mongodb-org-4.2.15
-1.amzn1.x86_64.rpm: Header V3 RSA/SHA1 Signature, key ID 058f8b6b: NOKEY
Public key for mongodb-org-4.2.15-1.amzn1.x86_64.rpm is not installed
(1/5): mongodb-org-4.2.15-1.amzn1.x86_64.rpm               | 6.0 kB  00:00
(2/5): mongodb-org-mongos-4.2.15-1.amzn1.x86_64.rpm        |  15 MB  00:00
(3/5): mongodb-org-shell-4.2.15-1.amzn1.x86_64.rpm         |  17 MB  00:00
(4/5): mongodb-org-server-4.2.15-1.amzn1.x86_64.rpm        |  26 MB  00:00
(5/5): mongodb-org-tools-4.2.15-1.amzn1.x86_64.rpm         |  32 MB  00:00
--------------------------------------------------------------------------------
Total                                              76 MB/s |  89 MB  00:01
Retrieving key from https://www.mongodb.org/static/pgp/server-4.2.asc
Importing GPG key 0x058F8B6B:
 Userid     : "MongoDB 4.2 Release Signing Key <packaging@mongodb.com>"
 Fingerprint: e162 f504 a20c df15 827f 718d 4b7c 549a 058f 8b6b
 From       : https://www.mongodb.org/static/pgp/server-4.2.asc
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : mongodb-org-shell-4.2.15-1.amzn1.x86_64                      1/5
  Installing : mongodb-org-mongos-4.2.15-1.amzn1.x86_64                     2/5
  Installing : mongodb-org-tools-4.2.15-1.amzn1.x86_64                      3/5
  Installing : mongodb-org-server-4.2.15-1.amzn1.x86_64                     4/5
  Installing : mongodb-org-4.2.15-1.amzn1.x86_64                            5/5
  Verifying  : mongodb-org-4.2.15-1.amzn1.x86_64                            1/5
  Verifying  : mongodb-org-server-4.2.15-1.amzn1.x86_64                     2/5
  Verifying  : mongodb-org-tools-4.2.15-1.amzn1.x86_64                      3/5
  Verifying  : mongodb-org-mongos-4.2.15-1.amzn1.x86_64                     4/5
  Verifying  : mongodb-org-shell-4.2.15-1.amzn1.x86_64                      5/5

Installed:
  mongodb-org.x86_64 0:4.2.15-1.amzn1
  mongodb-org-mongos.x86_64 0:4.2.15-1.amzn1
  mongodb-org-server.x86_64 0:4.2.15-1.amzn1
  mongodb-org-shell.x86_64 0:4.2.15-1.amzn1
  mongodb-org-tools.x86_64 0:4.2.15-1.amzn1
```



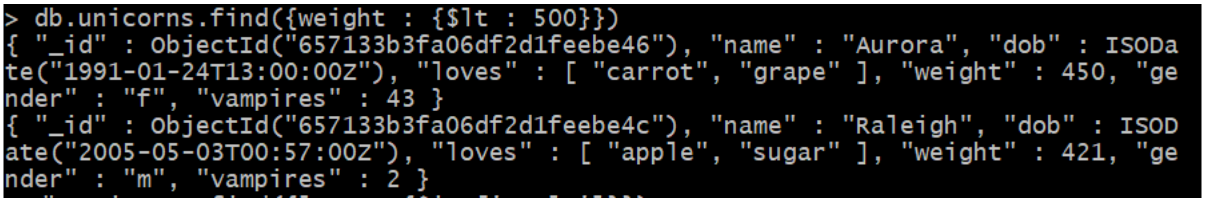
```
Complete!
[hadoop@ip-172-31-13-99 ~]$ sudo systemctl start mongod
[hadoop@ip-172-31-13-99 ~]$ mongo
MongoDB shell version v4.2.15
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName
=mongodb
Implicit session: session { "id" : UUID("19f22f85-a8c9-4176-9ca1-4c0d42061134")
}
MongoDB server version: 4.2.15
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
        https://docs.mongodb.com/
Questions? Try the MongoDB Developer Community Forums
        https://community.mongodb.com
Server has startup warnings:
2023-12-07T02:52:32.704+0000 I  CONTROL  [initandlisten]
2023-12-07T02:52:32.704+0000 I  CONTROL  [initandlisten] ** WARNING: Access cont
rol is not enabled for the database.
2023-12-07T02:52:32.704+0000 I  CONTROL  [initandlisten] **          Read and wr
ite access to data and configuration is unrestricted.
2023-12-07T02:52:32.704+0000 I  CONTROL  [initandlisten]
---
Enable MongoDB's free cloud-based monitoring service, which will then receive an
d display
metrics about your deployment (disk utilization, CPU, operation statistics, etc)
.

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

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeM
onitoring()
---

> use assignment;
switched to db assignment
> load(‘./load.js’);
2023-12-07T02:53:23.793+0000 E  QUERY    [js] uncaught exception: SyntaxError: i
llegal character :
@(shell):1:5
> load('./load.js');
true
> clear
2023-12-07T02:53:50.826+0000 E  QUERY    [js] uncaught exception: ReferenceError
: clear is not defined :
@(shell):1:1
> db.unicorns.find();
{ "_id" : ObjectId("657133b3fa06df2d1feebe45"), "name" : "Horny", "dob" : ISODat
e("1992-03-13T07:47:00Z"), "loves" : [ "carrot", "papaya" ], "weight" : 600, "ge
nder" : "m", "vampires" : 63 }
{ "_id" : ObjectId("657133b3fa06df2d1feebe46"), "name" : "Aurora", "dob" : ISODa
te("1991-01-24T13:00:00Z"), "loves" : [ "carrot", "grape" ], "weight" : 450, "ge
nder" : "f", "vampires" : 43 }
{ "_id" : ObjectId("657133b3fa06df2d1feebe47"), "name" : "Unicrom", "dob" : ISOD
ate("1973-02-09T22:10:00Z"), "loves" : [ "energon", "redbull" ], "weight" : 984,
 "gender" : "m", "vampires" : 182 }
{ "_id" : ObjectId("657133b3fa06df2d1feebe48"), "name" : "Roooooodles", "dob" :
ISODate("1979-08-18T18:44:00Z"), "loves" : [ "apple" ], "weight" : 575, "gender"
```

Exercises:

Exercise 1) (1 point)

Write a command that finds all unicorns having weight less than 500 pounds. Include the code you executed and some sample output as the result of this exercise. Recall you can place the command, if you choose, into a file, say ‘ex1.js’ and execute it with the load command as above and similarly for the following exercises.

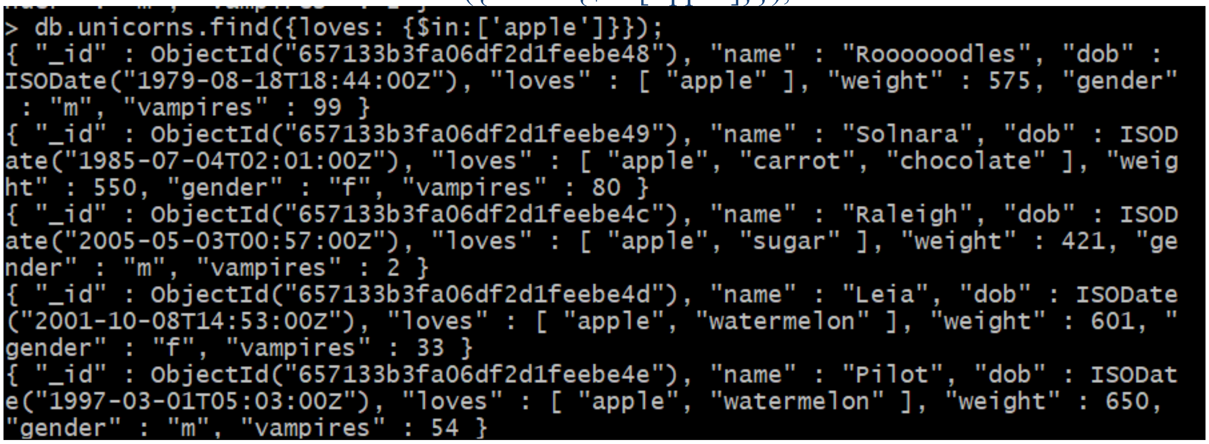
Command used:- “db.unicorns.find({weight : {$lt : 500}});”



```
> db.unicorns.find({weight : {$lt : 500}})
{ "_id" : ObjectId("657133b3fa06df2d1feebe46"), "name" : "Aurora", "dob" : ISODa
te("1991-01-24T13:00:00Z"), "loves" : [ "carrot", "grape" ], "weight" : 450, "ge
nder" : "f", "vampires" : 43 }
{ "_id" : ObjectId("657133b3fa06df2d1feebe4c"), "name" : "Raleigh", "dob" : ISOD
ate("2005-05-03T00:57:00Z"), "loves" : [ "apple", "sugar" ], "weight" : 421, "ge
nder" : "m", "vampires" : 2 }
```

Exercise 2) (1 point)
Write a command that finds all unicorns who love apples. Hint, search for “apple”. Include the code you executed and some sample output as the result of this exercise.

Command used: - “db.unicorns.find({loves: {$in:['apple']}});”



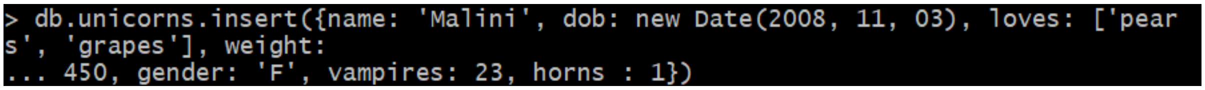
```
> db.unicorns.find({loves: {$in:['apple']}});
{ "_id" : ObjectId("657133b3fa06df2d1feebe48"), "name" : "Roooooodles", "dob" :
ISODate("1979-08-18T18:44:00Z"), "loves" : [ "apple" ], "weight" : 575, "gender"
 : "m", "vampires" : 99 }
{ "_id" : ObjectId("657133b3fa06df2d1feebe49"), "name" : "Solnara", "dob" : ISOD
ate("1985-07-04T02:01:00Z"), "loves" : [ "apple", "carrot", "chocolate" ], "weig
ht" : 550, "gender" : "f", "vampires" : 80 }
{ "_id" : ObjectId("657133b3fa06df2d1feebe4c"), "name" : "Raleigh", "dob" : ISOD
ate("2005-05-03T00:57:00Z"), "loves" : [ "apple", "sugar" ], "weight" : 421, "ge
nder" : "m", "vampires" : 2 }
{ "_id" : ObjectId("657133b3fa06df2d1feebe4d"), "name" : "Leia", "dob" : ISODate
("2001-10-08T14:53:00Z"), "loves" : [ "apple", "watermelon" ], "weight" : 601, "
gender" : "f", "vampires" : 33 }
{ "_id" : ObjectId("657133b3fa06df2d1feebe4e"), "name" : "Pilot", "dob" : ISODat
e("1997-03-01T05:03:00Z"), "loves" : [ "apple", "watermelon" ], "weight" : 650,
"gender" : "m", "vampires" : 54 }
```

Exercise 3) (1 point)
Write a command that adds a unicorn with the following attributes to the collection. Note dob means “Date of Birth.”

Attribute	Value(s)
name	Malini
dob	11/03/2008
loves	pears, grapes
weight	450
gender	F
vampires	23
horns	1

Include the code you executed to insert this unicorn into the collection along with the output of a find command showing it is in the collection.

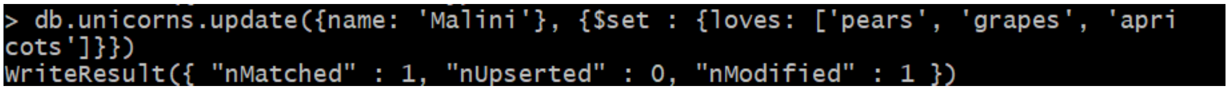
Command used: - “db.unicorns.insert({name: 'Malini', dob: new Date(2008, 11, 03), loves: ['pears', 'grapes'], weight:450, gender: 'F', vampires: 23, horns : 1});



```
> db.unicorns.insert({name: 'Malini', dob: new Date(2008, 11, 03), loves: ['pear
s', 'grapes'], weight:
... 450, gender: 'F', vampires: 23, horns : 1})
```

Exercise 4) (1 point)
Write a command that updates the above record to add apricots to the list of things Malini loves. Include the code you executed and some sample output showing the addition.

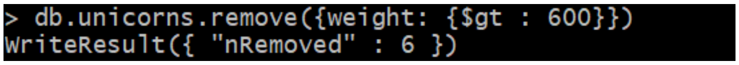
Command used: - “db.unicorns.update({name: 'Malini'}, {$set : {loves: ['pears', 'grapes', 'apricots']}});”



```
> db.unicorns.update({name: 'Malini'}, {$set : {loves: ['pears', 'grapes', 'apri
cots']}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
```

Exercise 5) (1 point)
Write a command that deletes all unicorns with weight more than 600 pounds. Include the code you executed and some sample output as the result of this exercise.

Command used: - “db.unicorns.remove({weight: {$gt : 600}});



```
> db.unicorns.remove({weight: {$gt : 600}})
WriteResult({ "nRemoved" : 6 })
```