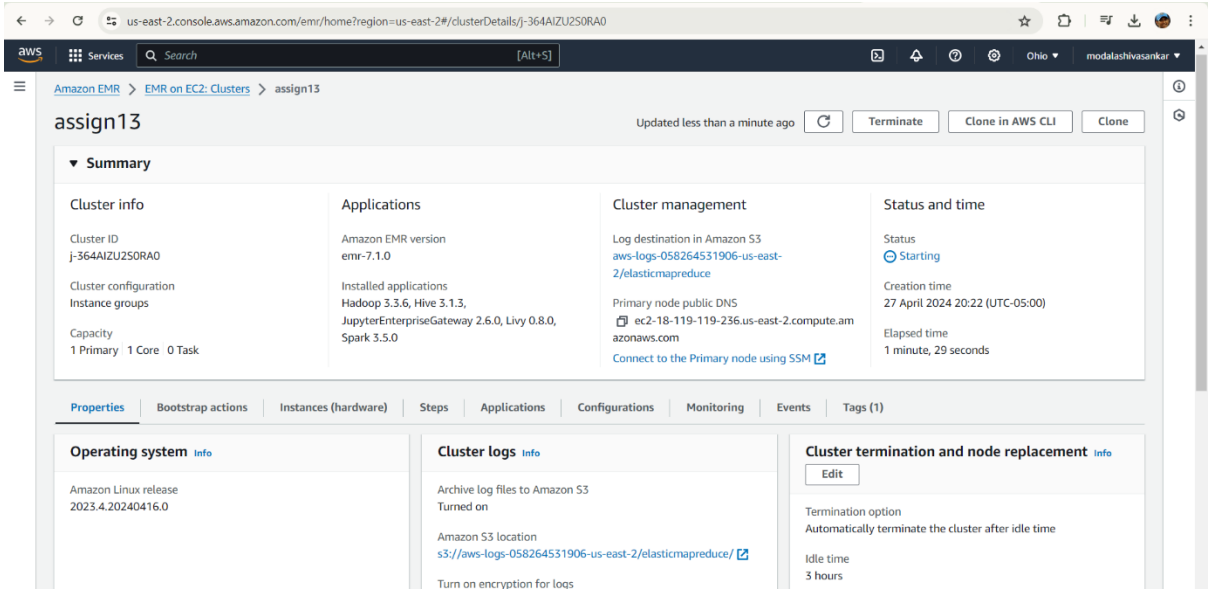
CSP 554 Big Data Technologies

Assignment – #13

Shiva Sankar Modala(A20517528)

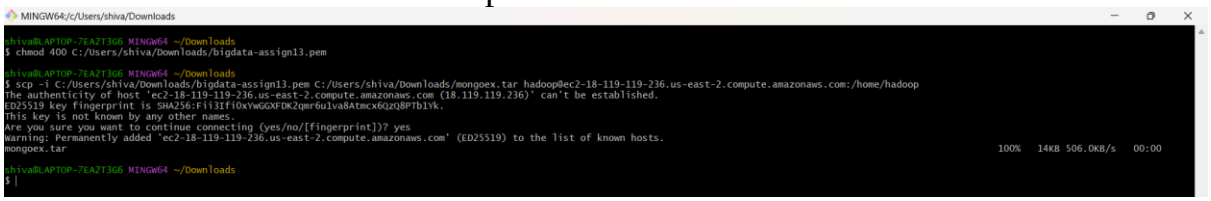
Step A – Start an EMR cluster

Start up a EMR cluster as previously, but instead of choosing the “Core Hadoop” configuration chose the “Spark Interactive” configuration (see below), otherwise proceed as before.



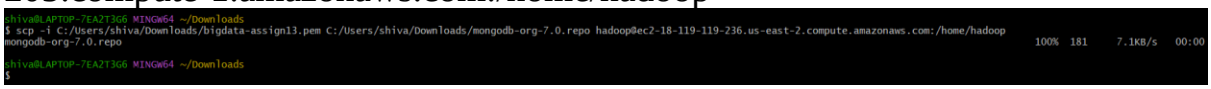
Step B – Download the assignment software (mongoex.tar, mongodb-org-7.0.repo) to master node

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, but note the target location is /home/hadoop).
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-7.0.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, but note the target location is /home/hadoop):

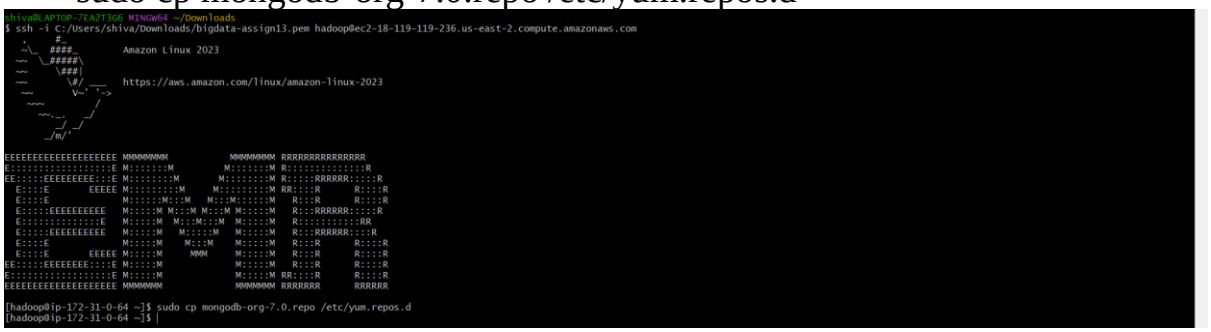
scp -i ./emr-key-pair-2.cer /Users/nachdaph/csp554-fall-2021/assignments/mongodb-org-7.0.repo hadoop@ec2-44-199-215-205.compute-1.amazonaws.com:/home/hadoop



Step C – Install assignment software (mongoex.zip, mongodb-org-7.0.repo)

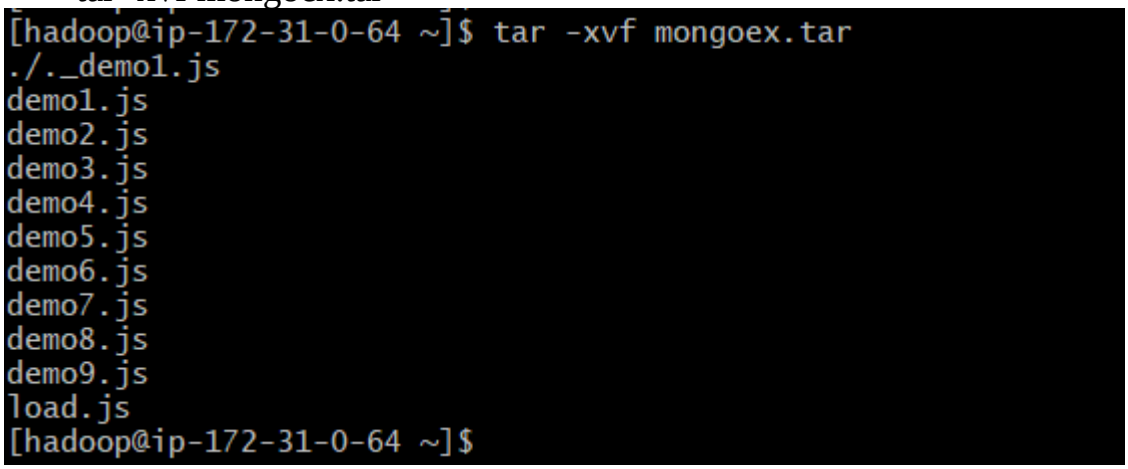
Enter the following into a terminal window which you have connected to the EMR master node. Going forward we will call this terminal connection Init-Term (don't cut and paste this, type it in manually):

sudo cp mongodb-org-7.0.repo /etc/yum.repos.d



Then enter this into Init-Term to unzip mongoex.tar (don't cut and paste this, type it in manually):

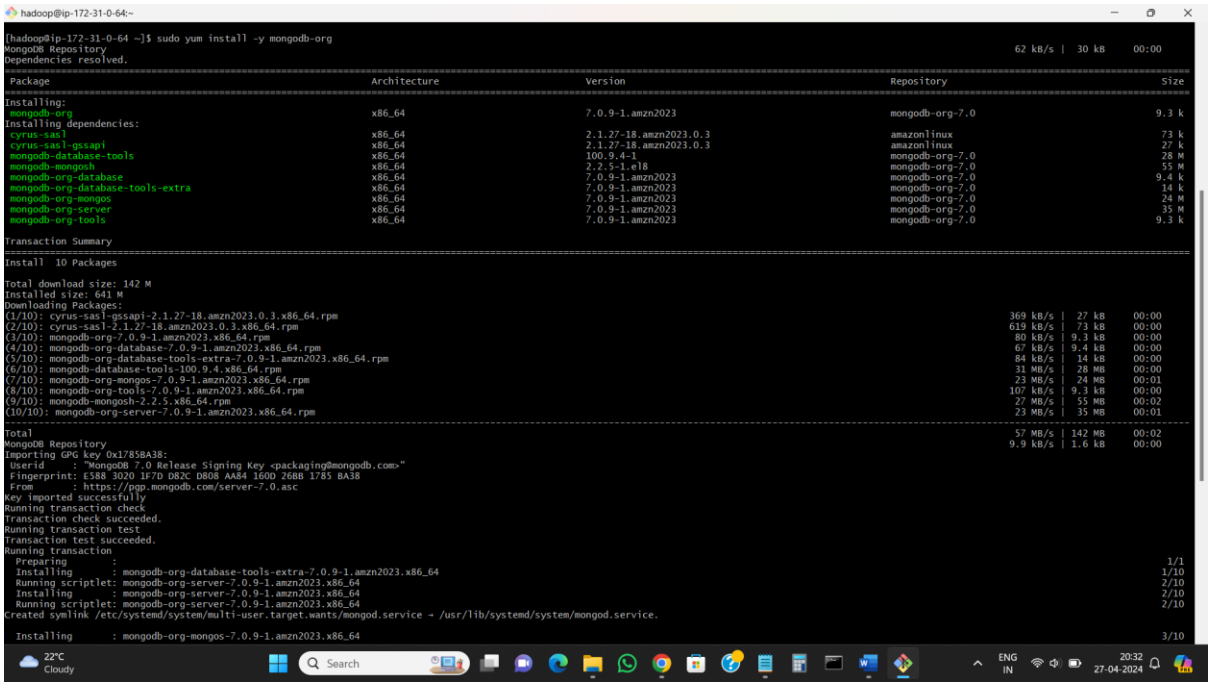
tar -xvf mongoex.tar



Step D – Install and start MongoDB

Enter the following into Init-Term to install MongoDB (don't cut and paste this, type it in manually):

sudo yum install -y mongodb-org



Now enter this into Init-Term to start mongodb (don't cut and paste this, type it in manually):

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.

To set up to use the MongoDB shell enter the following (don't cut and paste this, type it in manually):

sudo dnf erase -qy mongodb-mongosh

sudo dnf install -qy mongodb-mongosh-shared-openssl3

```
[hadoop@ip-172-31-0-64 ~]$ sudo dnf erase -qy mongodb-mongosh

Removed:
  cyrus-sasl-2.1.27-18.amzn2023.0.3.x86_64
  cyrus-sasl-gssapi-2.1.27-18.amzn2023.0.3.x86_64
  mongodb-database-tools-100.9.4-1.x86_64
  mongodb-mongosh-2.2.5-1.el8.x86_64
  mongodb-org-7.0.9-1.amzn2023.x86_64
  mongodb-org-database-7.0.9-1.amzn2023.x86_64
  mongodb-org-database-tools-extra-7.0.9-1.amzn2023.x86_64
  mongodb-org-mongos-7.0.9-1.amzn2023.x86_64
  mongodb-org-server-7.0.9-1.amzn2023.x86_64
  mongodb-org-tools-7.0.9-1.amzn2023.x86_64

[hadoop@ip-172-31-0-64 ~]$ sudo dnf install -qy mongodb-mongosh-shared-openssl3

Installed:
  mongodb-mongosh-shared-openssl3-2.2.5-1.el8.x86_64

[hadoop@ip-172-31-0-64 ~]$
```

Now start and run the mongodb shell as follows:

Mongosh

```
[hadoop@ip-172-31-0-64 ~]$ mongosh
Current Mongosh Log ID: 662da83aba7c17f4e45400a9
Connecting to:          mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.2.5
Using MongoDB:          7.0.9
Using Mongosh:          2.2.5

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


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

------
   The server generated these startup warnings when booting
   2024-04-28T01:33:23.833+00:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
   2024-04-28T01:33:23.833+00:00: vm.max_map_count is too low
------

test>
```

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 (don't cut and paste this, type it in manually):

use assignment;

```
test> use assignment;
switched to db assignment
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, you might see a warning message. This is ok. I use the insert() rather than the insertOne() command just to illustrate that the former is deprecated and in the future you should use the other.

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.

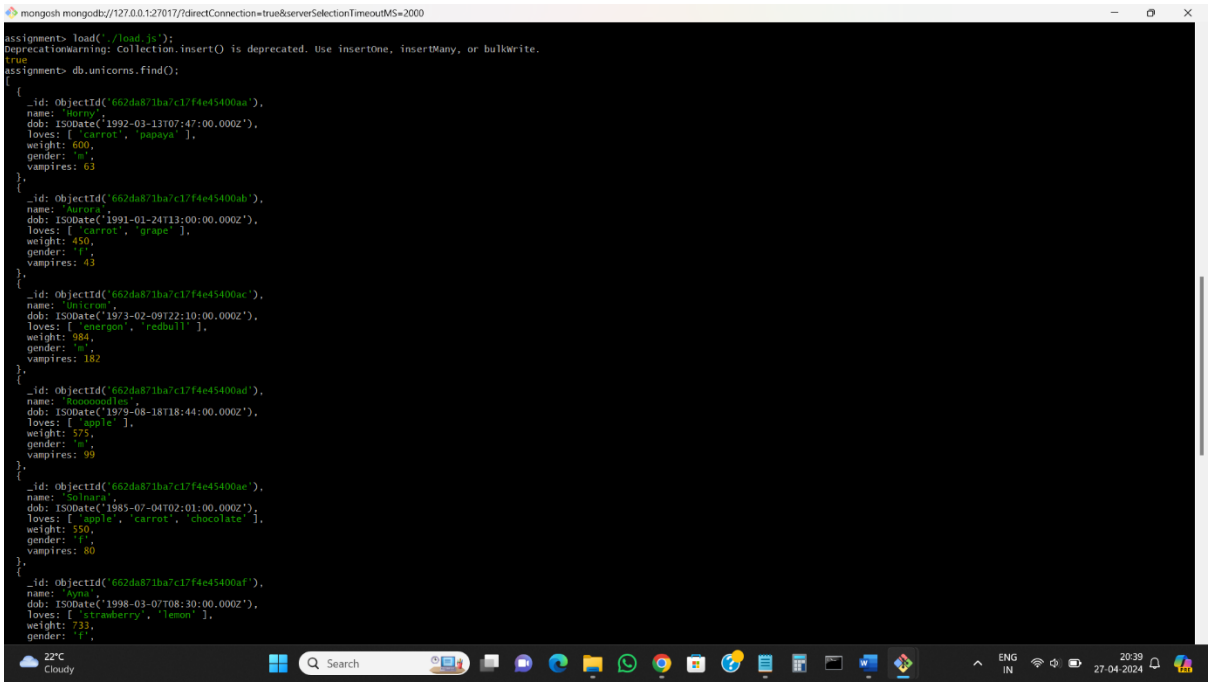
Confirm this has all worked by executing the following command in the MongoDB shell (don't cut and paste this, type it in manually):

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 (don't cut and paste this, type it in manually):

load('./demo1.js');

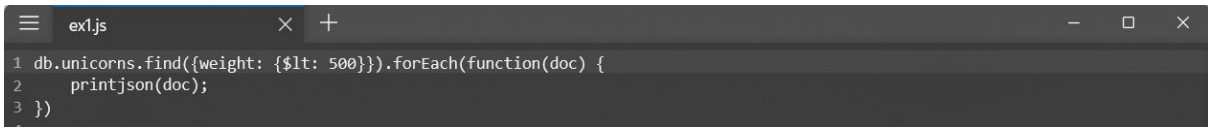
Note,as you execute some of the demo code, you might see a warning message. This is ok. I use the update() rather than the updateOne() command just to illustrate that the former is deprecated and in the future you should use the other.



Exercises:

Exercise 1) (1 point)

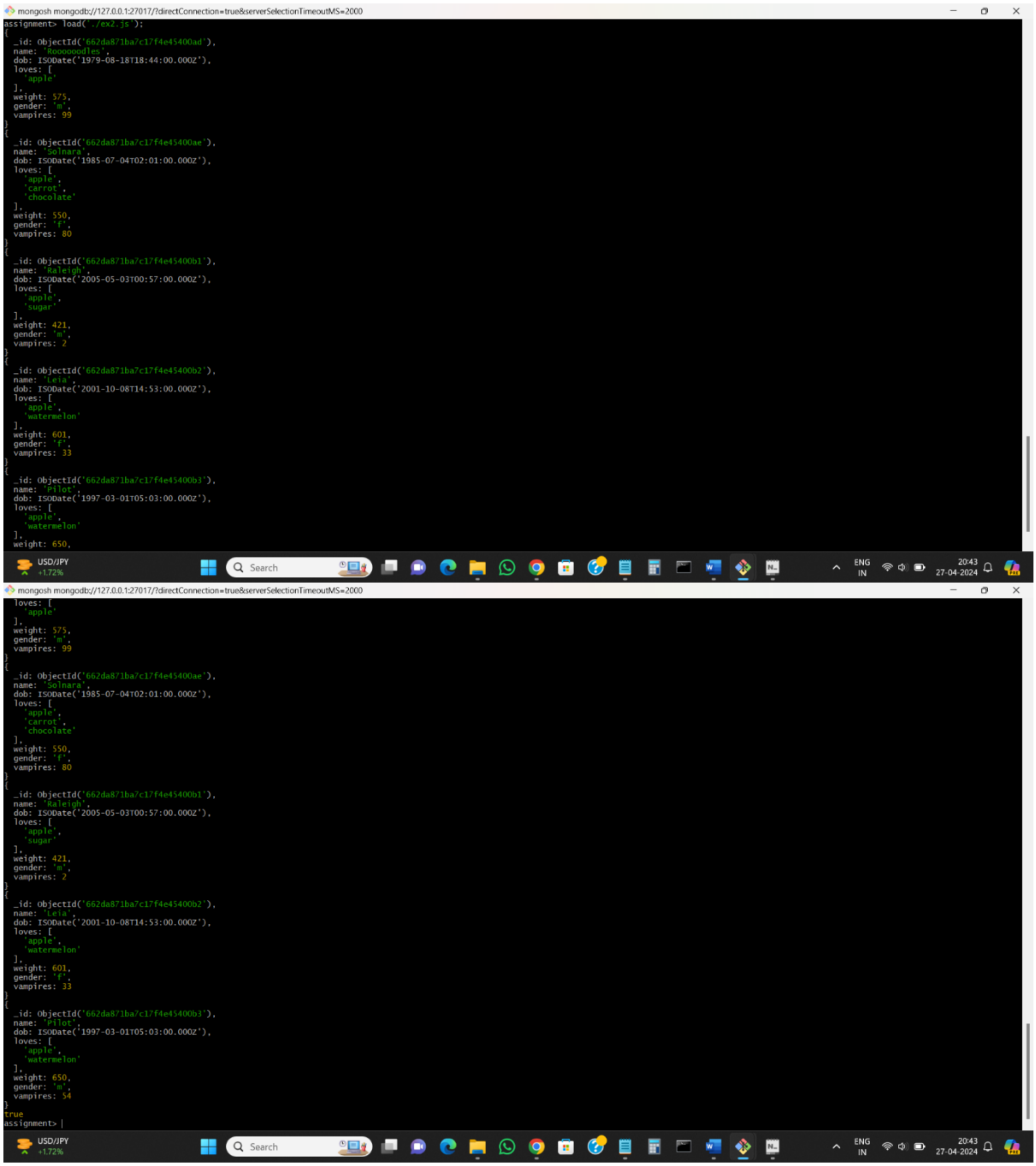
Write a command that finds all unicorns having weight less than 500 pounds. Include the code you executed and some sample output as the result of this exercise. Recall you can place the command, if you choose, into a file, say ‘ex1.js’ and execute it with the load command as above and similarly for the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



```
assignment> load('./ex1.js');
{
  _id: ObjectId('662da871ba7c17f4e45400ab'),
  name: 'Aurora',
  dob: ISODate('1991-01-24T13:00:00.000Z'),
  loves: [
    'carrot',
    'grape'
  ],
  weight: 450,
  gender: 'f',
  vampires: 43
}
{
  _id: ObjectId('662da871ba7c17f4e45400b1'),
  name: 'Raleigh',
  dob: ISODate('2005-05-03T00:57:00.000Z'),
  loves: [
    'apple',
    'sugar'
  ],
  weight: 421,
  gender: 'm',
  vampires: 2
}
true
assignment>
```

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.



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.

ex3.js

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

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
assignment> load('./ex3.js');
true
assignment> db.unicorns.find();
[
  {
    _id: ObjectId('662da871ba7c17f4e45400aa'),
    name: 'Horny',
    dob: ISODate('1992-03-13T07:47:00.000Z'),
    loves: [ 'carrot', 'papaya' ],
    weight: 600,
    gender: 'm',
    vampires: 63
  },
  {
    _id: ObjectId('662da871ba7c17f4e45400ab'),
    name: 'Aurora',
    dob: ISODate('1991-01-24T13:00:00.000Z'),
    loves: [ 'carrot', 'grape' ],
    weight: 450,
    gender: 'f',
    vampires: 43
  },
  {
    _id: ObjectId('662da871ba7c17f4e45400ac'),
    name: 'Unicrom',
    dob: ISODate('1973-02-09T22:10:00.000Z'),
    loves: [ 'energon', 'redbull' ],
    weight: 984,
    gender: 'm',
    vampires: 182
  },
  {
    _id: ObjectId('662da871ba7c17f4e45400ad'),
    name: 'Roooooodles',
    dob: ISODate('1979-08-18T18:44:00.000Z'),
    loves: [ 'apple' ],
    weight: 575,
    gender: 'm',
    vampires: 99
  },
  {
    _id: ObjectId('662da871ba7c17f4e45400ae'),
    name: 'Solnara',
    dob: ISODate('1985-07-04T02:01:00.000Z'),
    loves: [ 'apple', 'carrot', 'chocolate' ],
    weight: 550,
    gender: 'f',
    vampires: 80
  },
  {
    _id: ObjectId('662da871ba7c17f4e45400af'),
    name: 'Ayna',
    dob: ISODate('1998-03-07T08:30:00.000Z'),
    loves: [ 'strawberry', 'lemon' ],
    weight: 733,
    gender: 'f',
    vampires: 40
  },
```

Tomorrow's high
To break record

Search

ENG
IN

20:44
27-04-2024

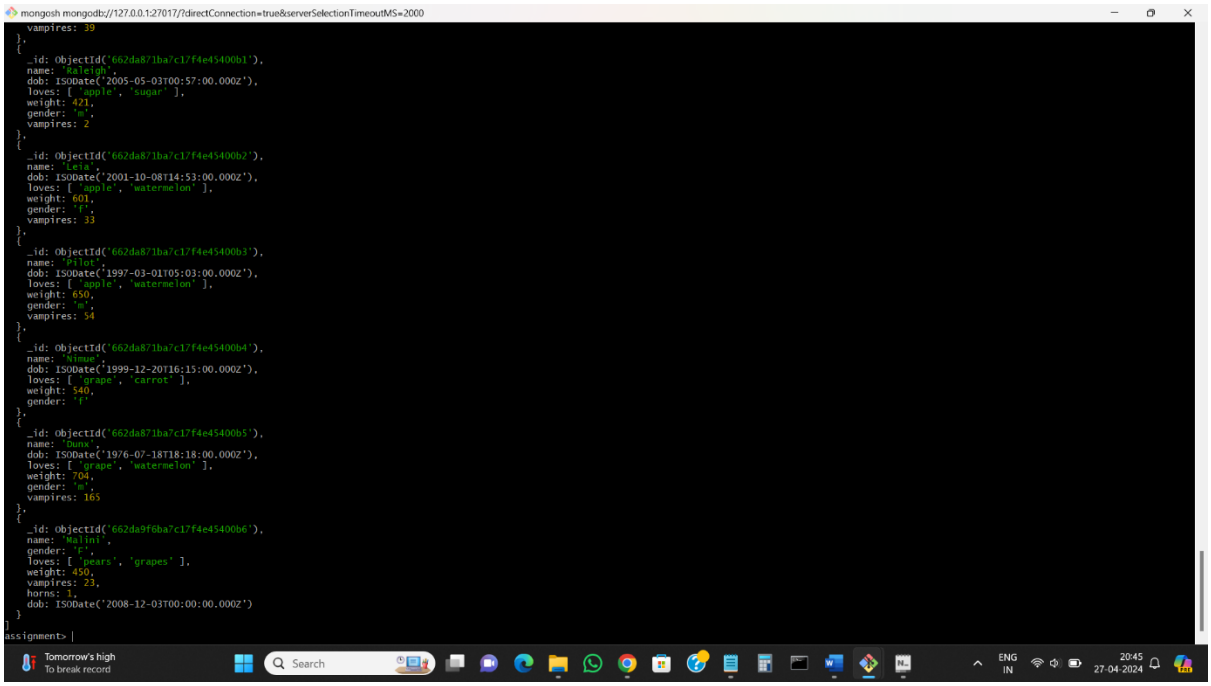
```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
  },
  {
    _id: ObjectId('662da871ba7c17f4e45400b0'),
    name: 'Kenny',
    dob: ISODate('1997-07-01T10:42:00.000Z'),
    loves: [ 'grape', 'lemon' ],
    weight: 690,
    gender: 'm',
    vampires: 39
  },
  {
    _id: ObjectId('662da871ba7c17f4e45400b1'),
    name: 'Raleigh',
    dob: ISODate('2005-05-03T00:57:00.000Z'),
    loves: [ 'apple', 'sugar' ],
    weight: 421,
    gender: 'm',
    vampires: 2
  },
  {
    _id: ObjectId('662da871ba7c17f4e45400b2'),
    name: 'Leia',
    dob: ISODate('2001-10-08T14:53:00.000Z'),
    loves: [ 'apple', 'watermelon' ],
    weight: 601,
    gender: 'f',
    vampires: 33
  },
  {
    _id: ObjectId('662da871ba7c17f4e45400b3'),
    name: 'Pilot',
    dob: ISODate('1997-03-01T05:03:00.000Z'),
    loves: [ 'apple', 'watermelon' ],
    weight: 650,
    gender: 'm',
    vampires: 54
  },
  {
    _id: ObjectId('662da871ba7c17f4e45400b4'),
    name: 'Nimue',
    dob: ISODate('1999-12-20T16:15:00.000Z'),
    loves: [ 'grape', 'carrot' ],
    weight: 540,
    gender: 'f'
  },
  {
    _id: ObjectId('662da871ba7c17f4e45400b5'),
    name: 'Dunx',
    dob: ISODate('1976-07-18T18:18:00.000Z'),
    loves: [ 'grape', 'watermelon' ],
    weight: 704,
    gender: 'm',
    vampires: 165
  },
  {
    _id: ObjectId('662da9f6ba7c17f4e45400b6'),
    name: 'Malini',
    gender: 'F',
```

Tomorrow's high
To break record

Search

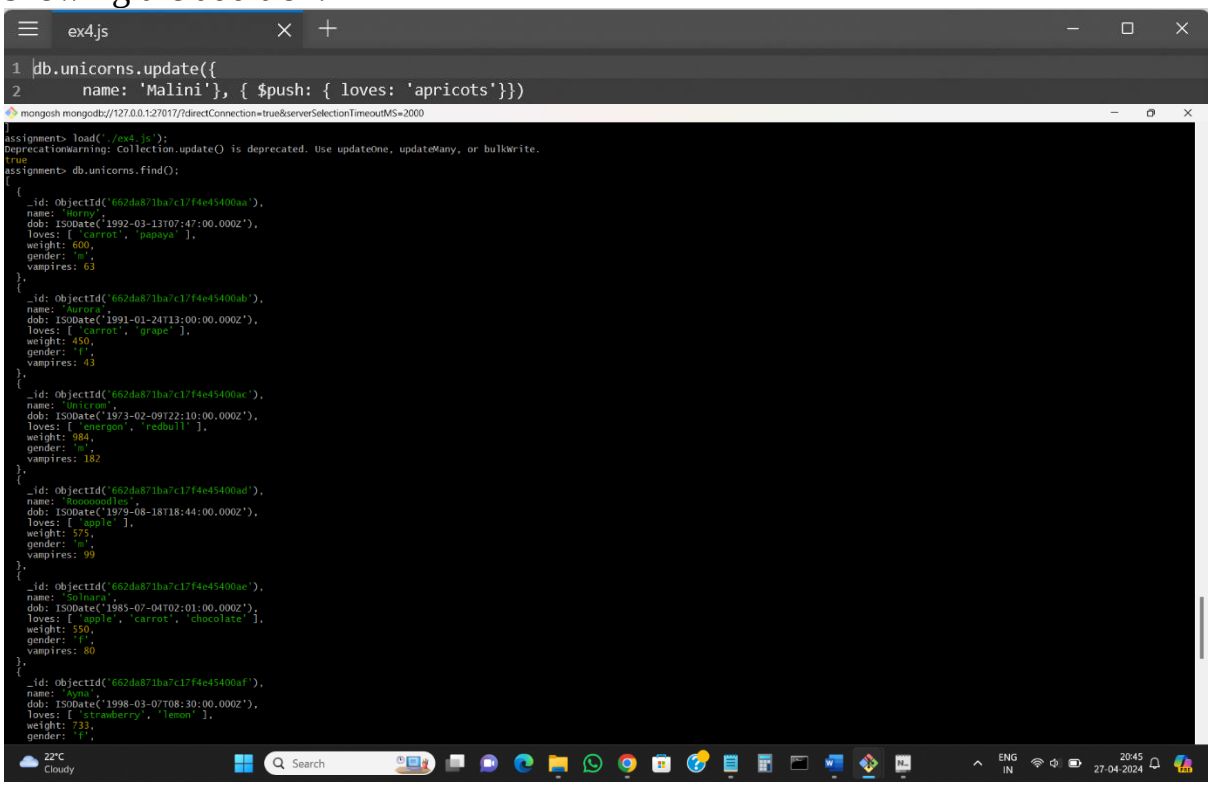
ENG
IN

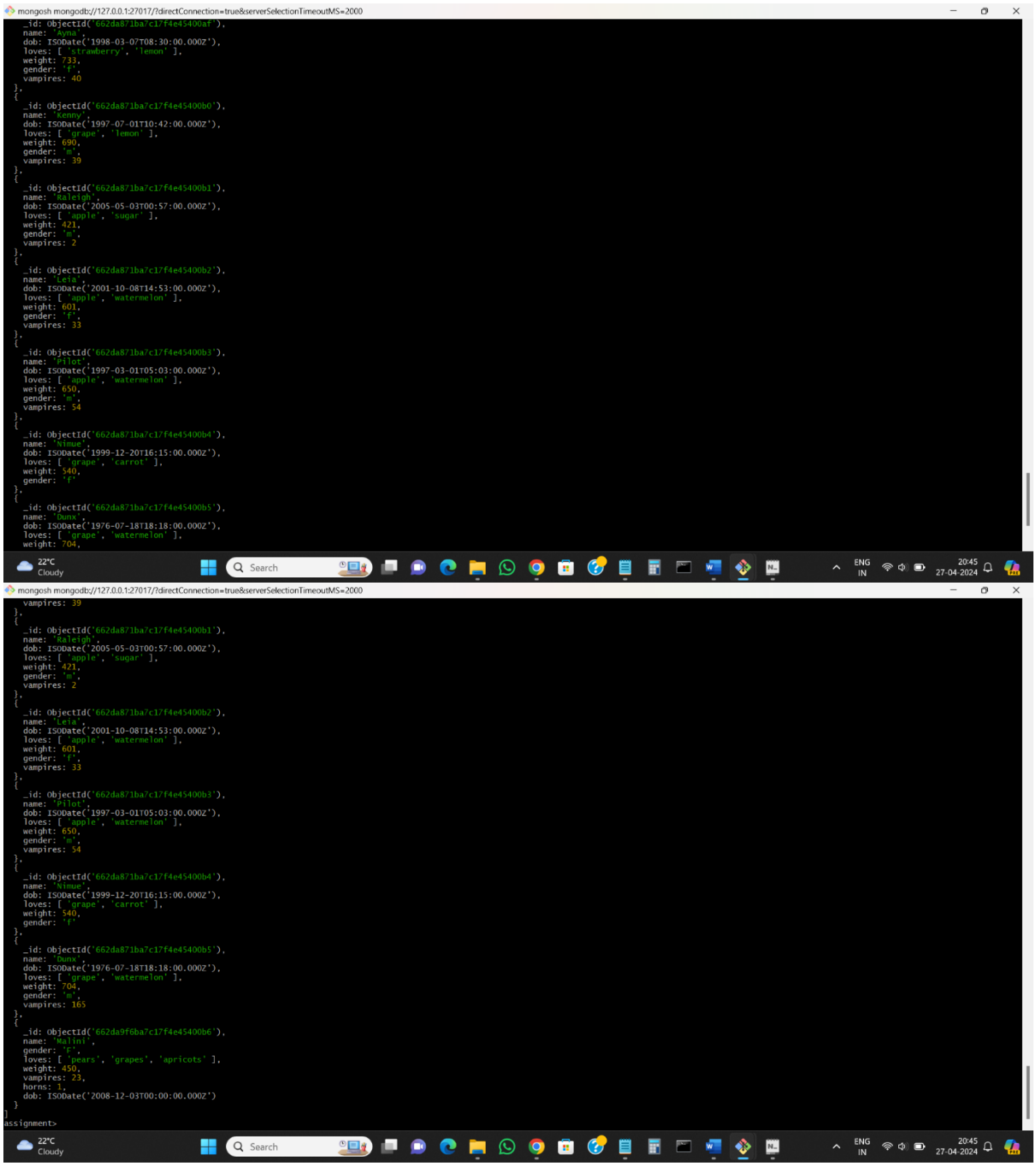
20:45
27-04-2024



Exercise 4) (1 point)

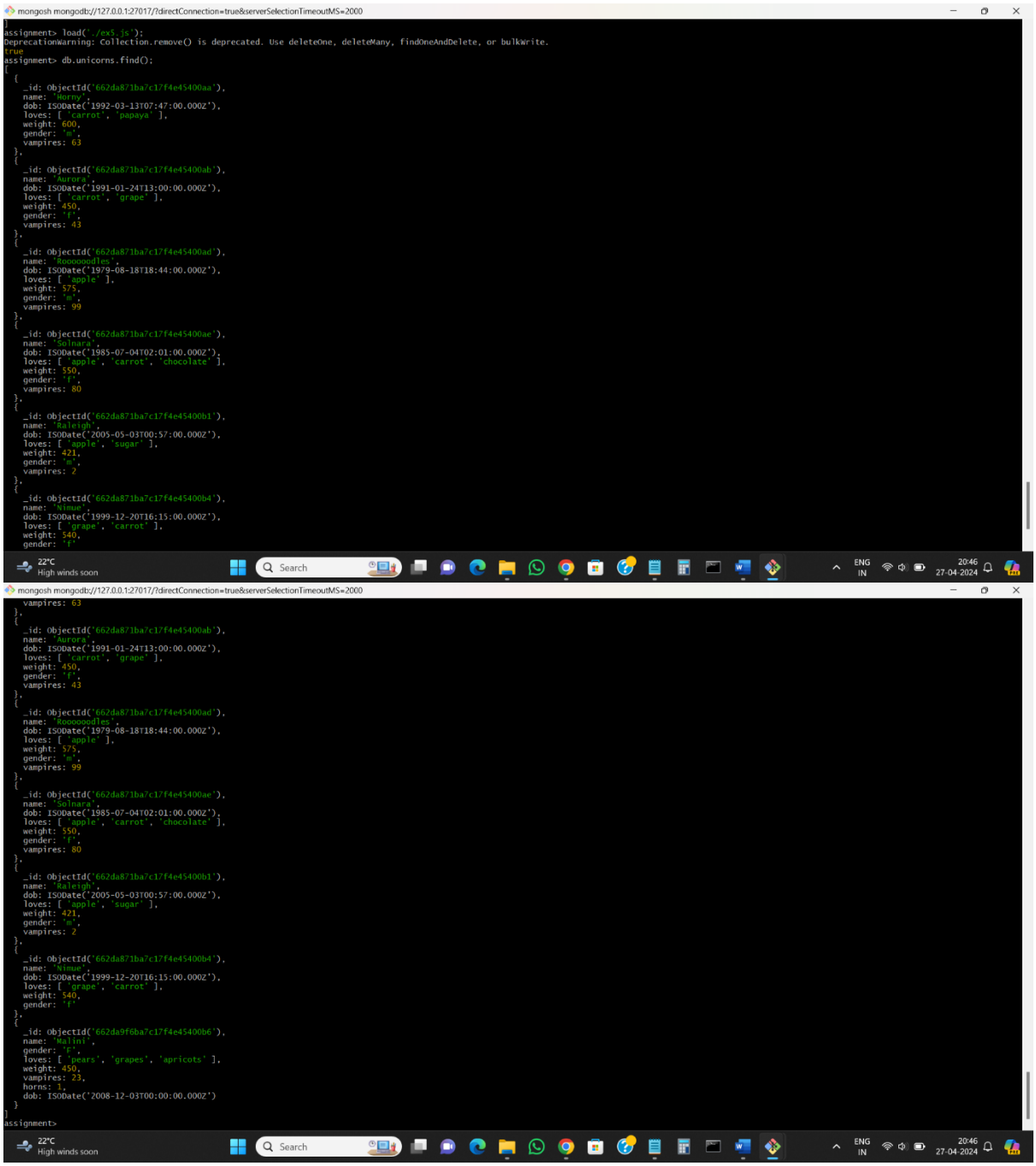
Write a command that updates the above record to add apricots to the list of things Malini loves. Include the code you executed and some sample output showing the addition.



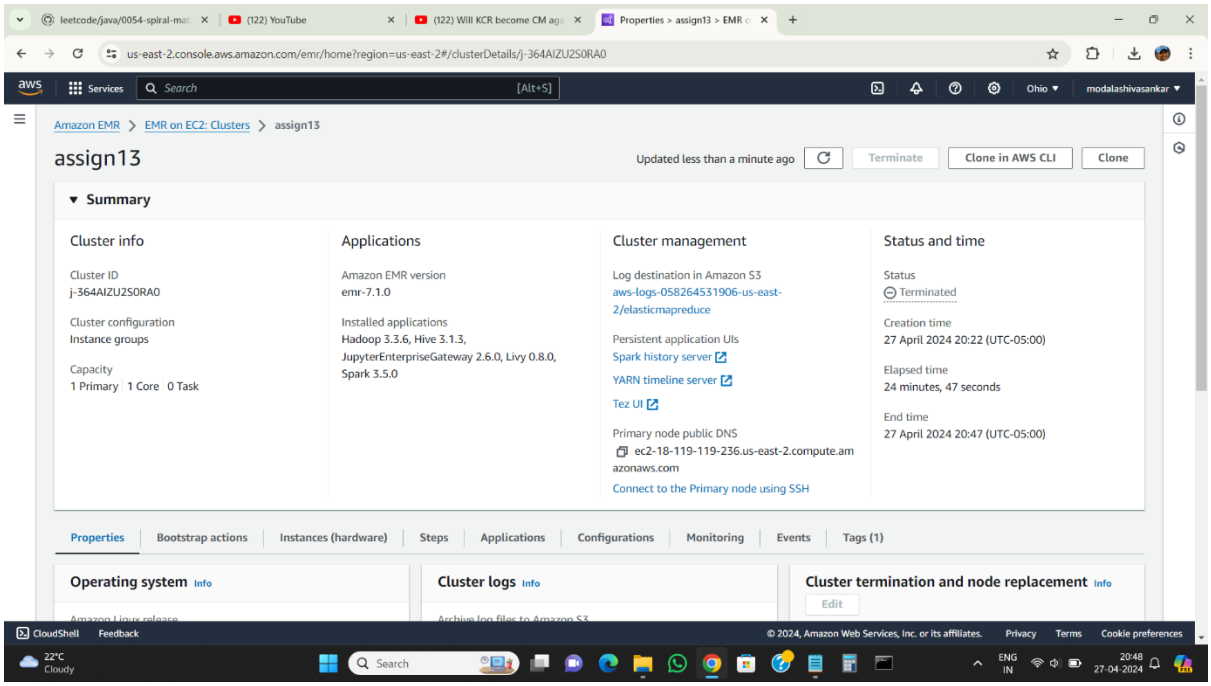


Exercise 5) (1 point)

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.



Terminated the cluster


leetcode/java/0054-spiral-mat
(122) YouTube
(122) Will KCR become CM aga
Properties > assign13 > EMR c
us-east-2.console.aws.amazon.com/emr/home?region=us-east-2#/clusterDetails/j-364AIZU2S0RA0
aws
Services
Search
[Alt+S]
Ohio
modalashivasankar
Amazon EMR > EMR on EC2: Clusters > assign13
assign13
Updated less than a minute ago
Terminate
Clone in AWS CLI
Clone
Summary
Cluster info
Cluster ID
j-364AIZU2S0RA0
Cluster configuration
Instance groups
Capacity
1 Primary | 1 Core | 0 Task
Applications
Amazon EMR version
emr-7.1.0
Installed applications
Hadoop 3.3.6, Hive 3.1.3, JupyterEnterpriseGateway 2.6.0, Livy 0.8.0, Spark 3.5.0
Cluster management
Log destination in Amazon S3
aws-logs-058264531906-us-east-2/elasticmapreduce
Persistent application UIs
Spark history server
YARN timeline server
Tez UI
Primary node public DNS
ec2-18-119-119-236.us-east-2.compute.amazonaws.com
Connect to the Primary node using SSH
Status and time
Status
Terminated
Creation time
27 April 2024 20:22 (UTC-05:00)
Elapsed time
24 minutes, 47 seconds
End time
27 April 2024 20:47 (UTC-05:00)
Properties
Bootstrap actions
Instances (hardware)
Steps
Applications
Configurations
Monitoring
Events
Tags (1)
Operating system Info
Cluster logs Info
Cluster termination and node replacement Info
Edit
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
22°C
Cloudy
Search
ENG
IN
20:48
27-04-2024