Mohammed Hannan Desai - mmd76

This project contains two applications:

1) car-recognizition
2) text-recognizition

Both projects are written in Java using maven project management tool to manage dependencies for AWS SDK.
To run this project, you will need Java – JDK & JRE and maven installed on the system, links to download both given below:

https://www.java.com/en/download/help/download_options.html

To check JAVA installed, run this command: **java --version**

https://maven.apache.org/download.cgi

To check maven installed, run this command: **mvn –version**

To run these both applications for development first go into that project directory, then run following commands:

1) **mvn clean package**

2) **mvn exec:java -Dexec.mainClass="org.example.App"**

To run this application is production environment in our case in EC2, we will compile it into single jar file and run that jar file on EC2, to create the jar run following command:

1) **mvn clean package**

So, it will create the jar file in target forlder with name "**text-recognizition-1.0-SNAPSHOT-jar-with-dependencies.jar**" for text-recognizition and "**car-recognizition-1.0-SNAPSHOT-jar-with-dependencies.jar**" for car-recognizition".

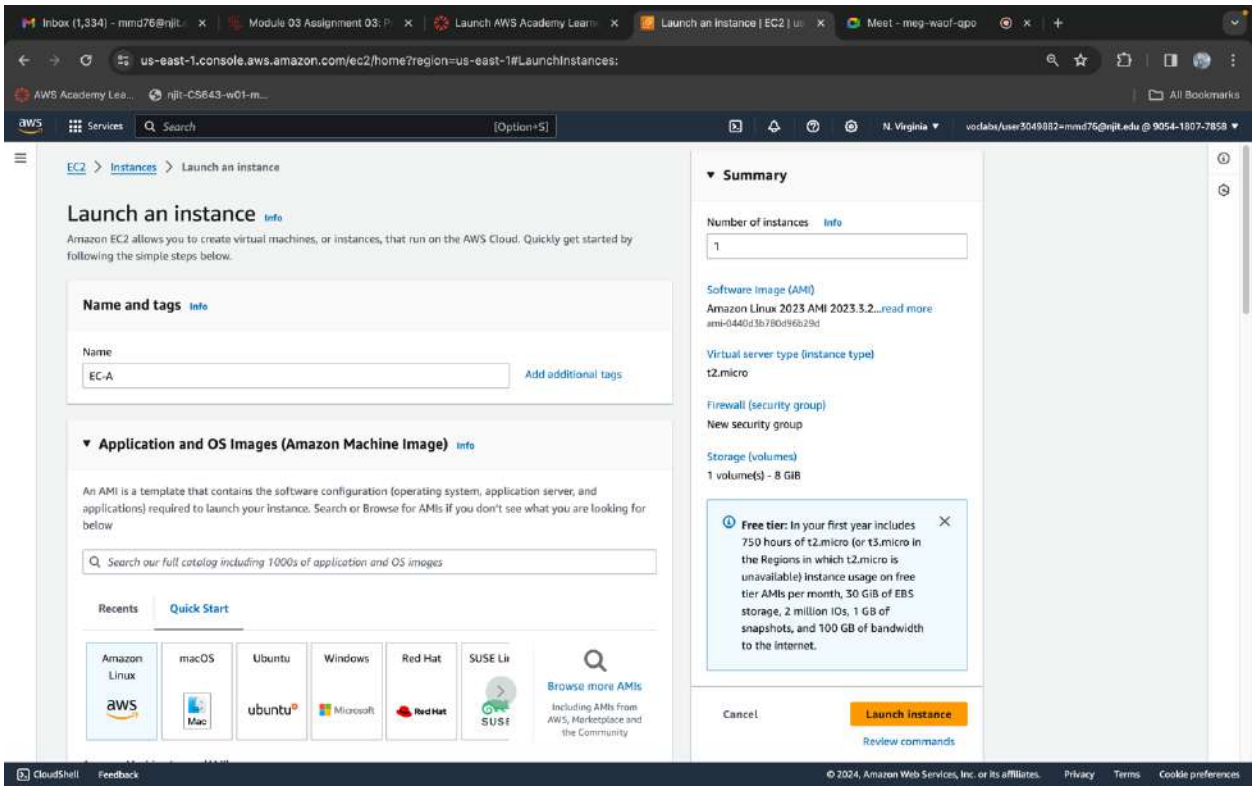
Now, we have to host & run car-recognizition jar file to EC-A & copy text-recongnizition jar file to EC-B, process to do this given in following pages.

To run this jar file run following command:

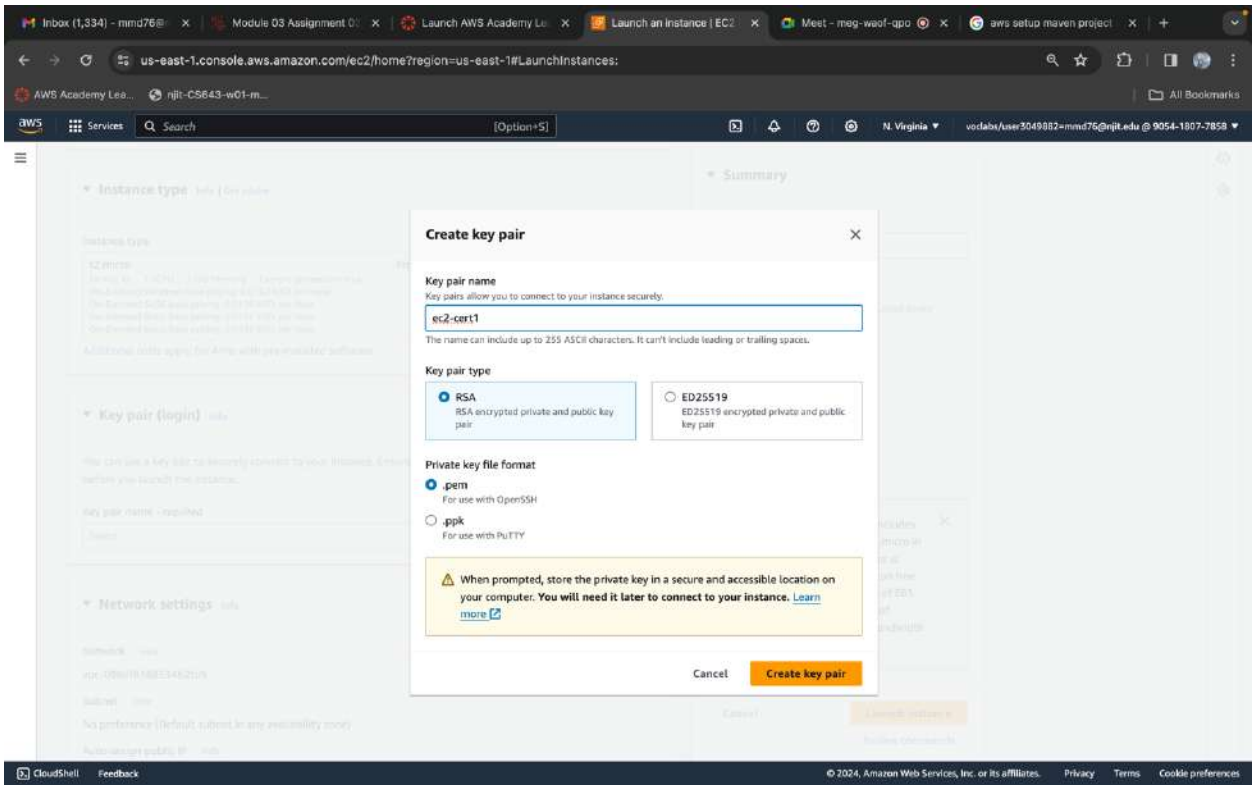
java -jar text-recognizition-1.0-SNAPSHOT-jar-with-dependencies.jar
java -jar car-recognizition-1.0-SNAPSHOT-jar-with-dependencies.jar

Cloud Setup

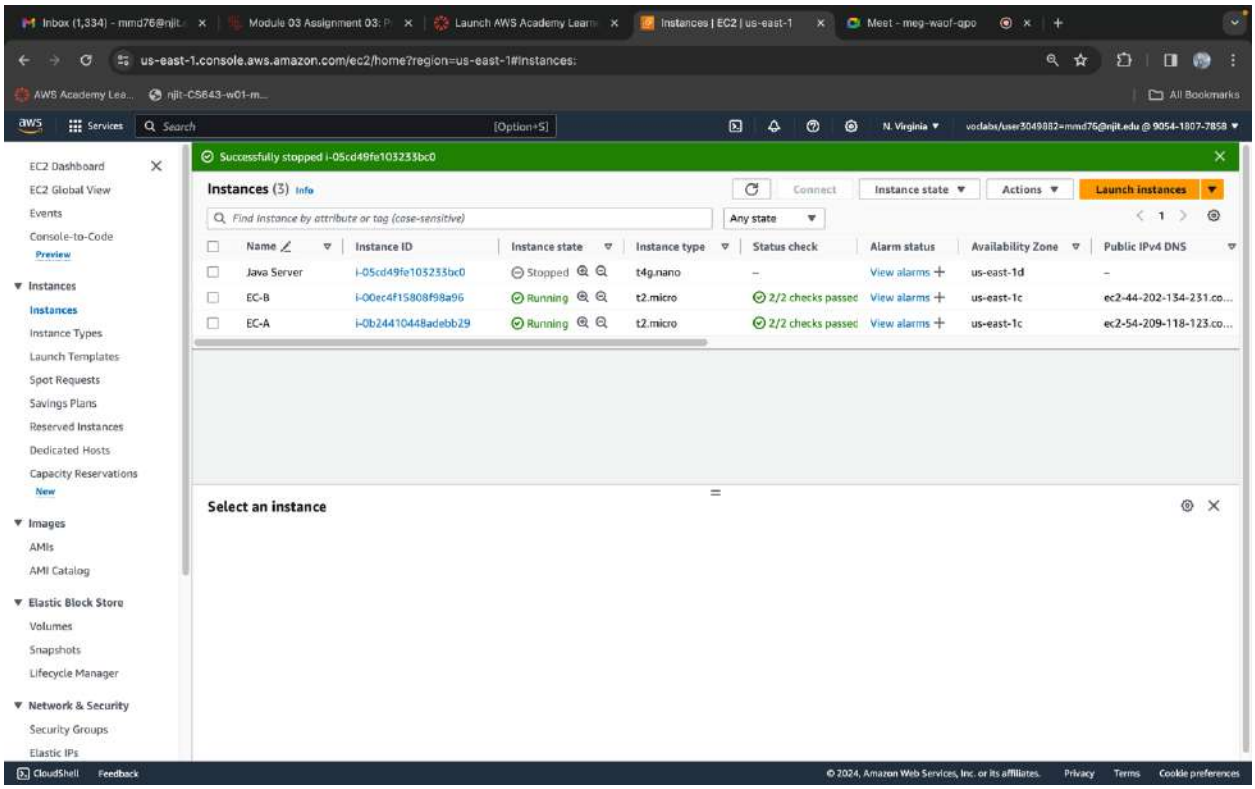
Create two instances: EC-A & EC-B
Go to AWS management console in that go to EC2 instances screen and click “Launch Instance” button there



During instance creation it will ask to select Key pair (login), this will be used for connecting with instances securely.
We need create one key pair and same key pair will be used for both the instances.
It will be used whenever we want to connect to this instances from our local machine.



Once both the instances have been created it will show in instances screen, wait until instance state is “Running”.



Java installation on EC2 to run java code

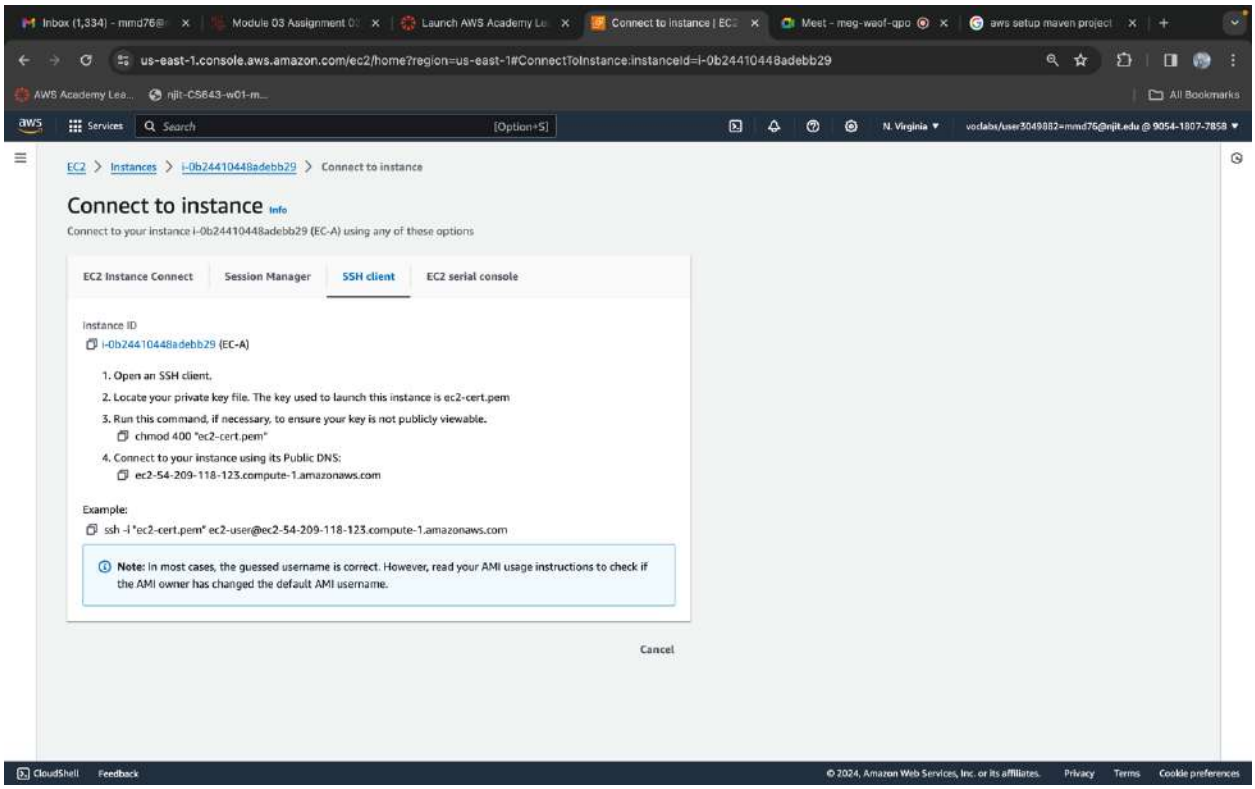
First connect to EC2 from local machine using SSH:

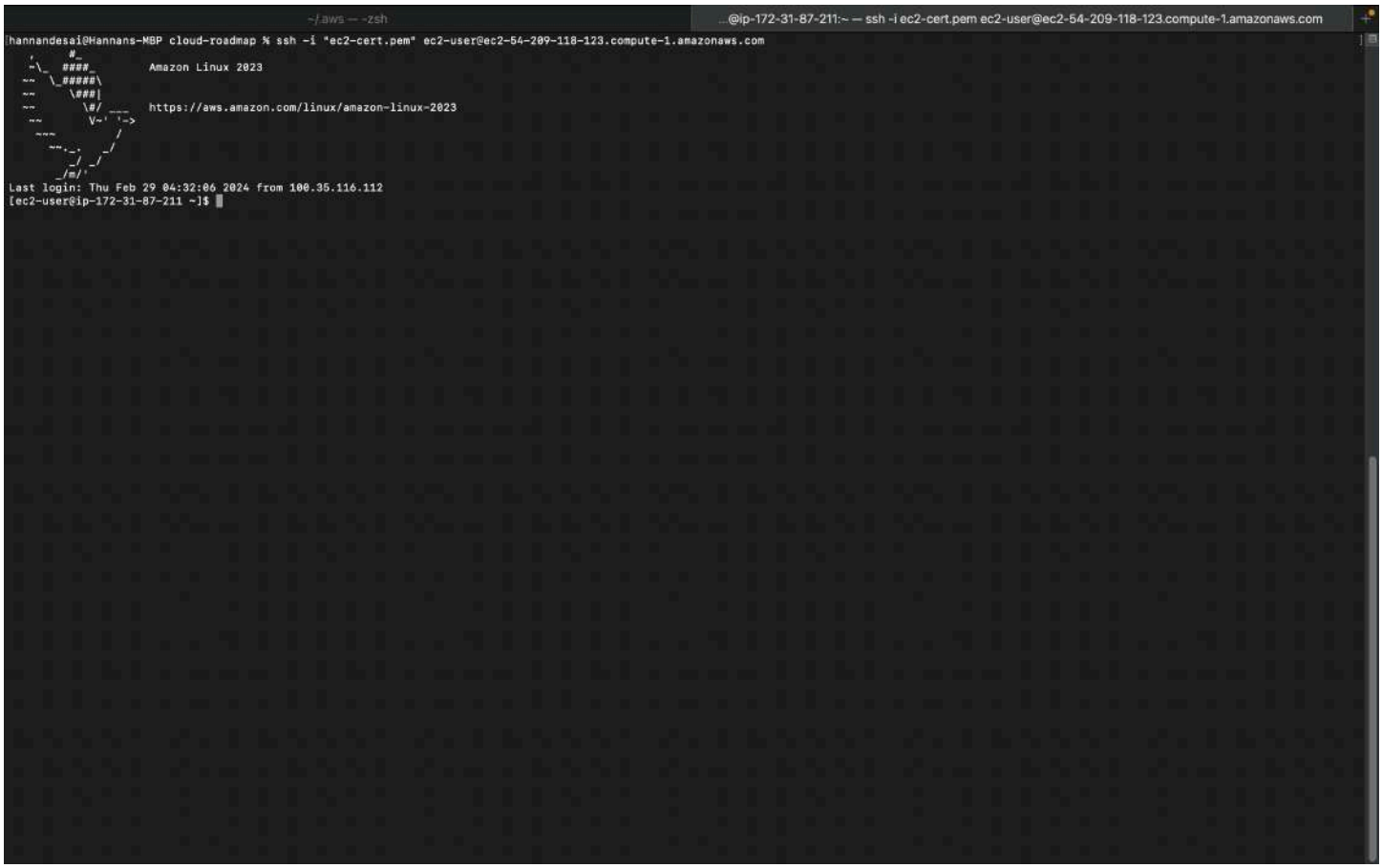
We can connect to EC2 using Public IPv4 DNS that we can find in when we click on particular EC2 from instances screen:

Form there click on “Connect” button, and it will show the command to connect EC2:

ssh -i "ec2-cert.pem" ec2-user@ec2-54-209-118-123.compute-1.amazonaws.com

In this command ec2-cert.pem is the certificate file that we created while creating an instance.


Inbox (1,334) - mmd76@
Module 03 Assignment 0
Launch AWS Academy Le
Connect to instance | EC2
Meet - meg-woof-qpo
aws setup maven project
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#ConnectToInstance:instanceId=i-0b24410448adebb29
AWS Academy Lea...
njit-CS643-w01-m...
All Bookmarks
Services
Search
[Option+S]
N. Virginia
voclabs/user3049882=mmd76@njit.edu @ 9054-1807-7858
EC2 > Instances > i-0b24410448adebb29 > Connect to instance
Connect to instance Info
Connect to your instance i-0b24410448adebb29 (EC-A) using any of these options
EC2 Instance Connect
Session Manager
SSH client
EC2 serial console
Instance ID
i-0b24410448adebb29 (EC-A)
1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is ec2-cert.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
chmod 400 "ec2-cert.pem"
4. Connect to your instance using its Public DNS:
ec2-54-209-118-123.compute-1.amazonaws.com
Example:
ssh -i "ec2-cert.pem" ec2-user@ec2-54-209-118-123.compute-1.amazonaws.com
Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.
Cancel
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences



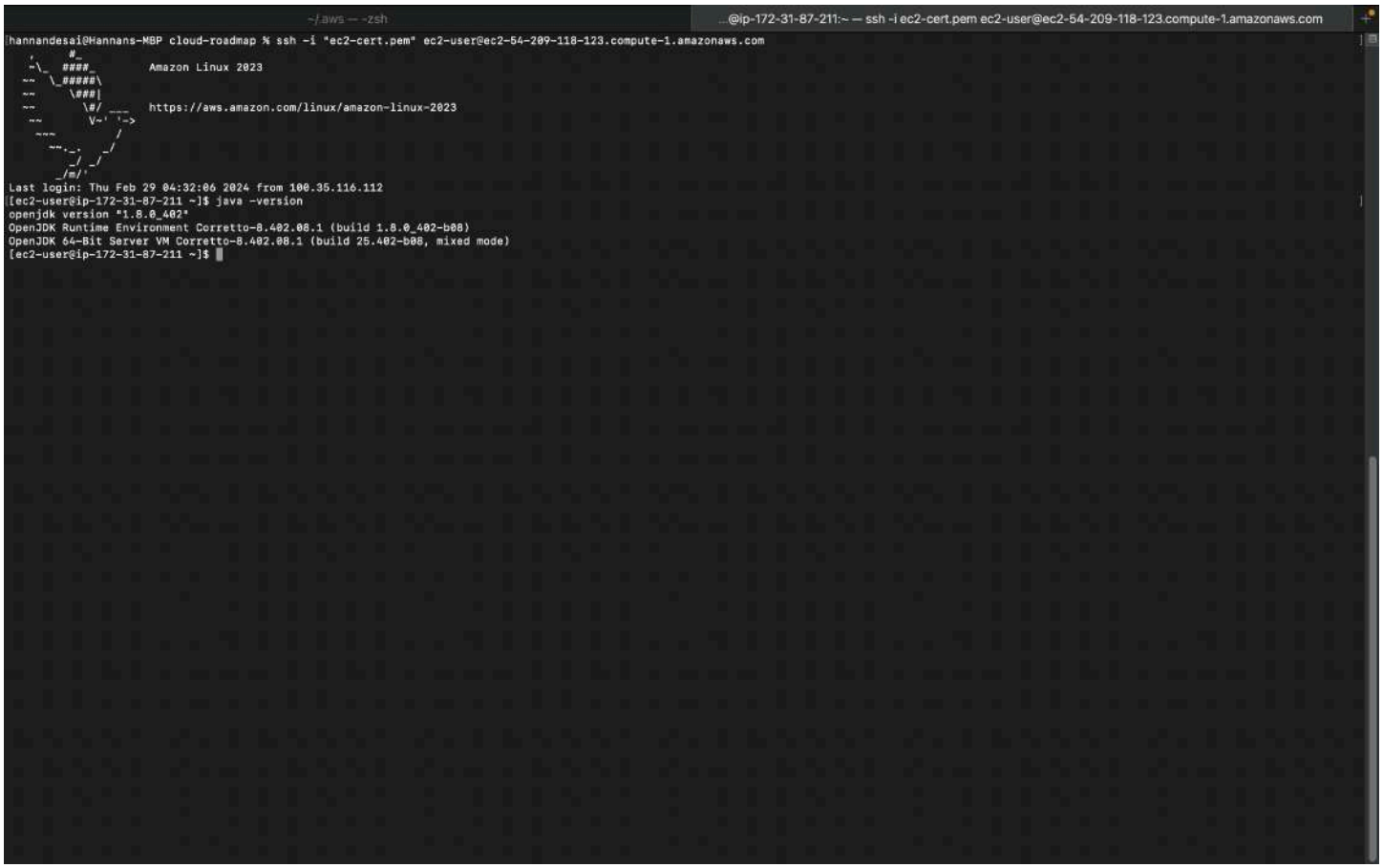
Once connected run below commands to install Java on EC2

```
sudo yum clean metadata
```

```
sudo yum install -y java-1.8.0-amazon-corretto.
```

After successfully running both of these commands it will intall Java on EC2 which can be used to run Java applications there, we can check if Java is installed or not using below command:

```
java -version
```



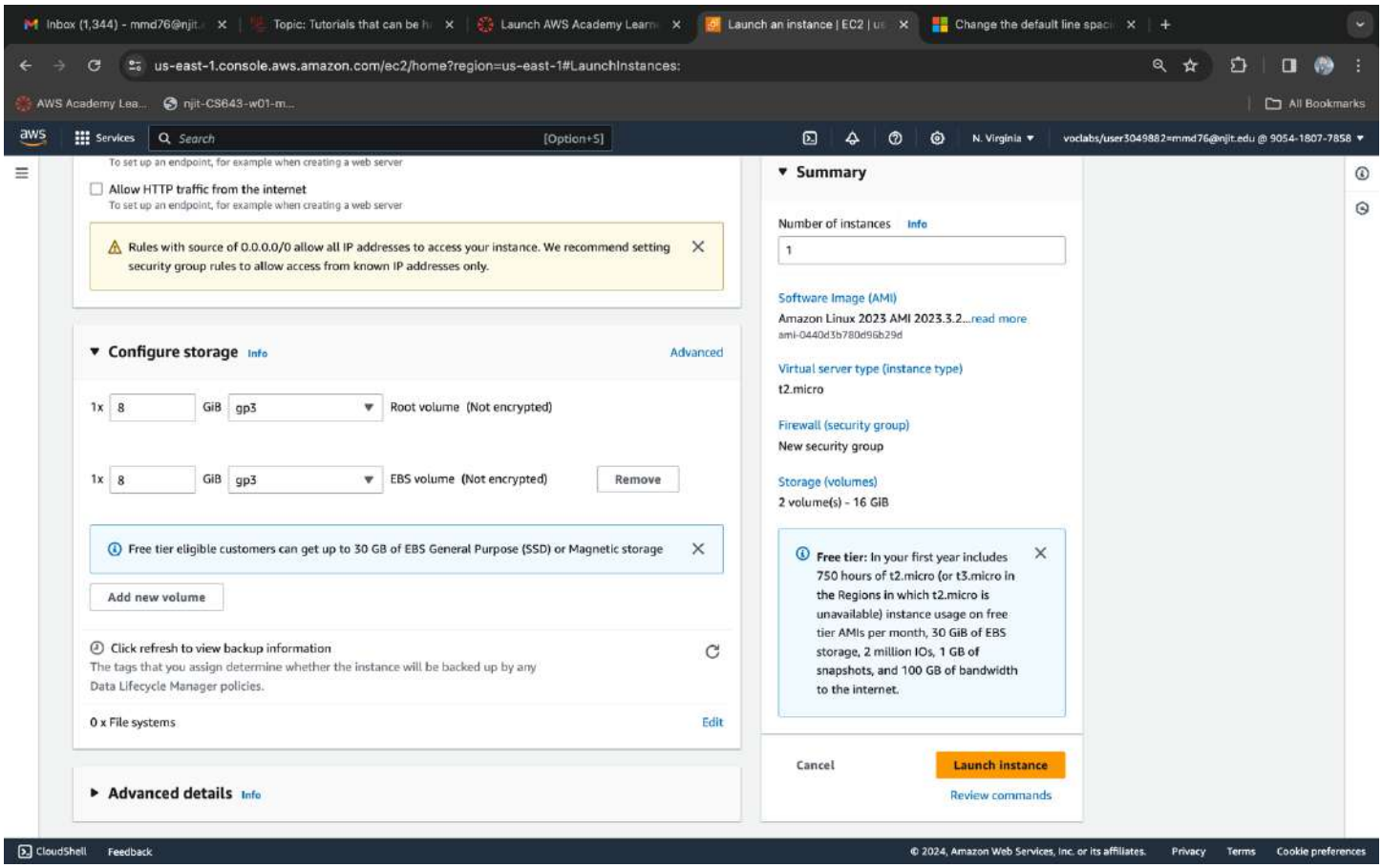
So, we can see that Java is installed!
We have to repeat this Java installation process for both the instances we have created:

Setup EBS (Elastic Block Storage) on EC-B for storing process output:

So, first when creating an EC-B instance, there will be an option to mount EBS to this instance, as given in following screenshot.

In this click on Add Volume option, modify the size to desired size, so in this instance AWS will allocate EBS storage as well.

So this is an extra step we have to follow while creating EC-B.



Now, we have to mount this EBS value in EC-B to make it available for use, to do this we have to run following commands on EC-B:

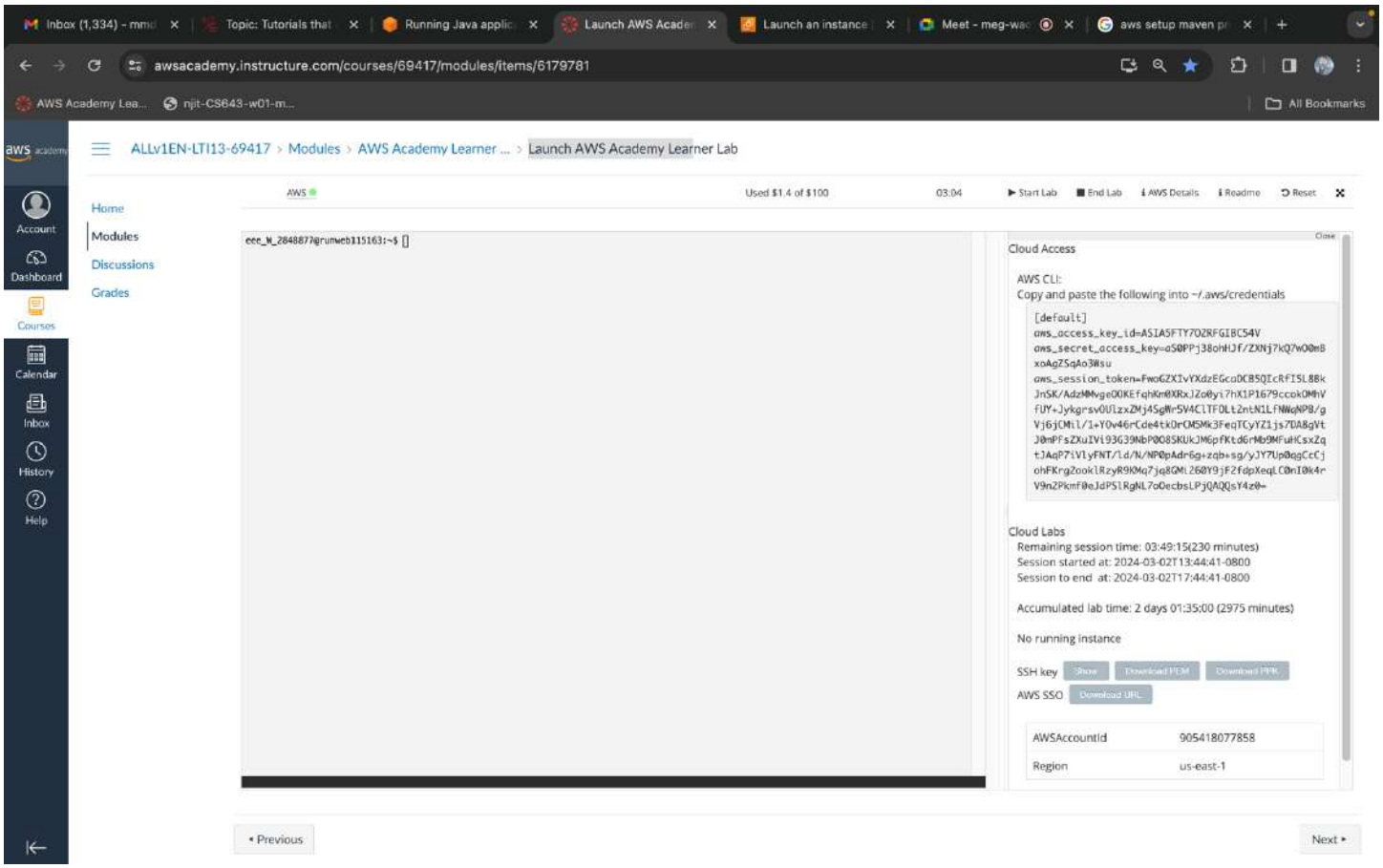
1. lsblk
2. sudo file -s /dev/xvdb
3. sudo mkfs -t xfs /dev/xvdb
4. sudo mkdir /ebs-data
5. sudo mount /dev/xvdb /ebs-data
6. file -s /dev/xvdb

Followed this link for more detailed explanation:

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-using-volumes.html

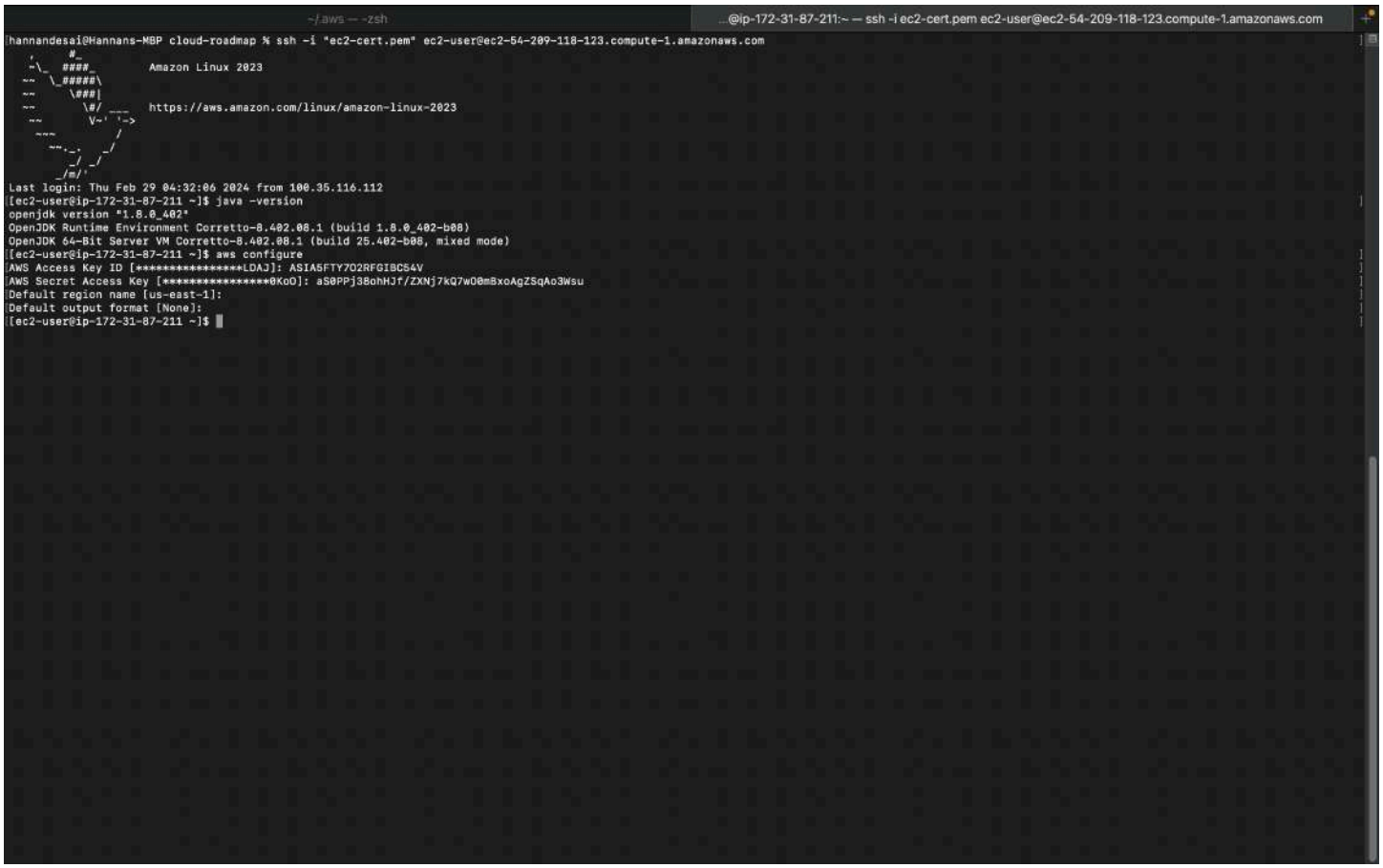
Store AWS credentials in EC2 to access AWS services like S3, SQS and AWS Rekognition:

We can get these credentials from Learner Lab Academy Launch Page right panel, refer below screenshot:



Now run following command in EC2 terminal to store these credentials:

`aws configure`

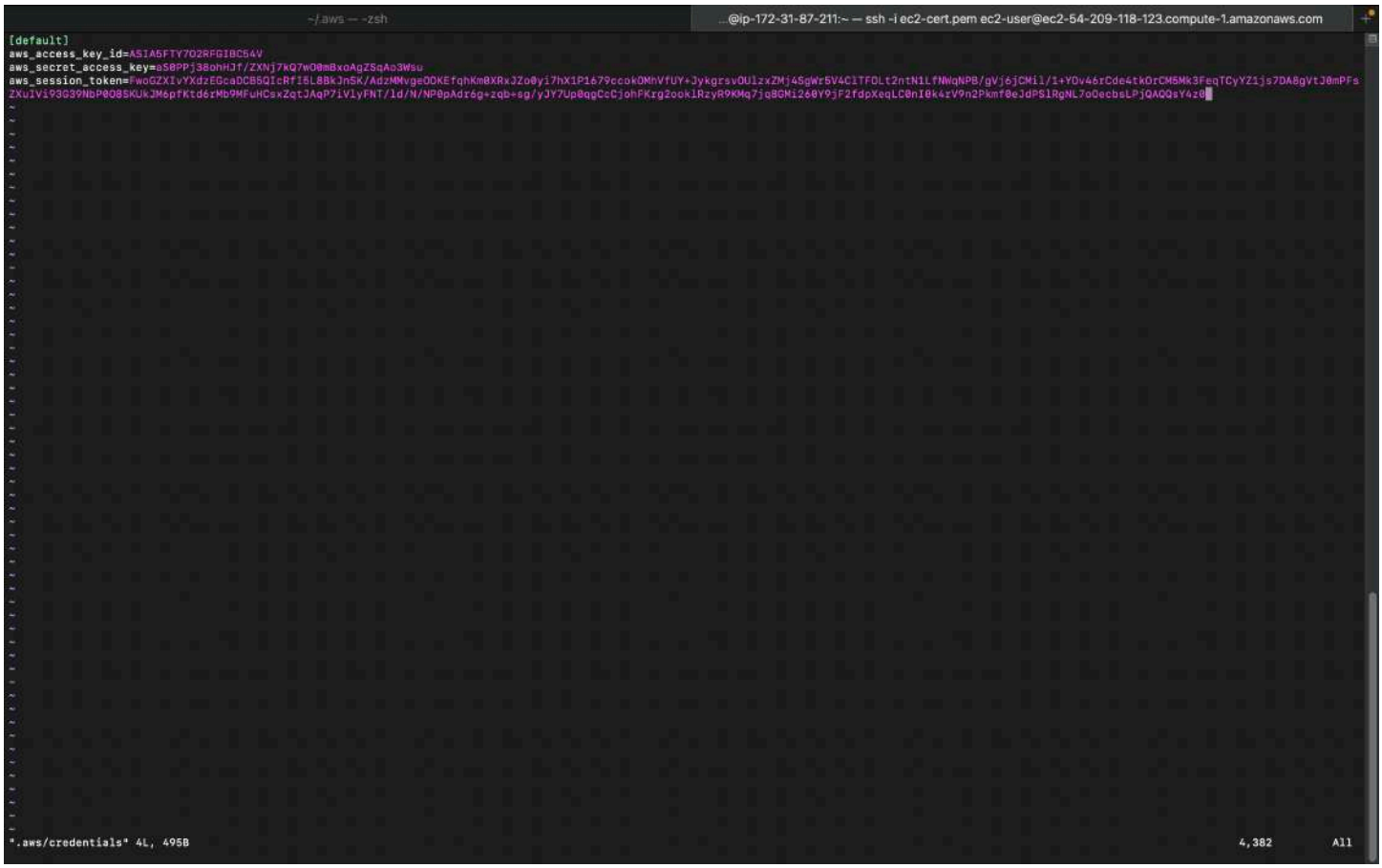


After running above command, it will create ".aws" directory in EC2 instance, in that directory we have to put our credentials in "credentials" file using vi command:

sudo vi ./aws/credentials

and copy paste the entire credentials into this file

To save this file: **esc ➔ : ➔ wq! ➔ enter**



Repeat this process as well for both the instances.

Upload jar file to server:
Next step is to upload project jar file onto EC2, car-rekognizition to EC-A and text-rekognizition to EC-B

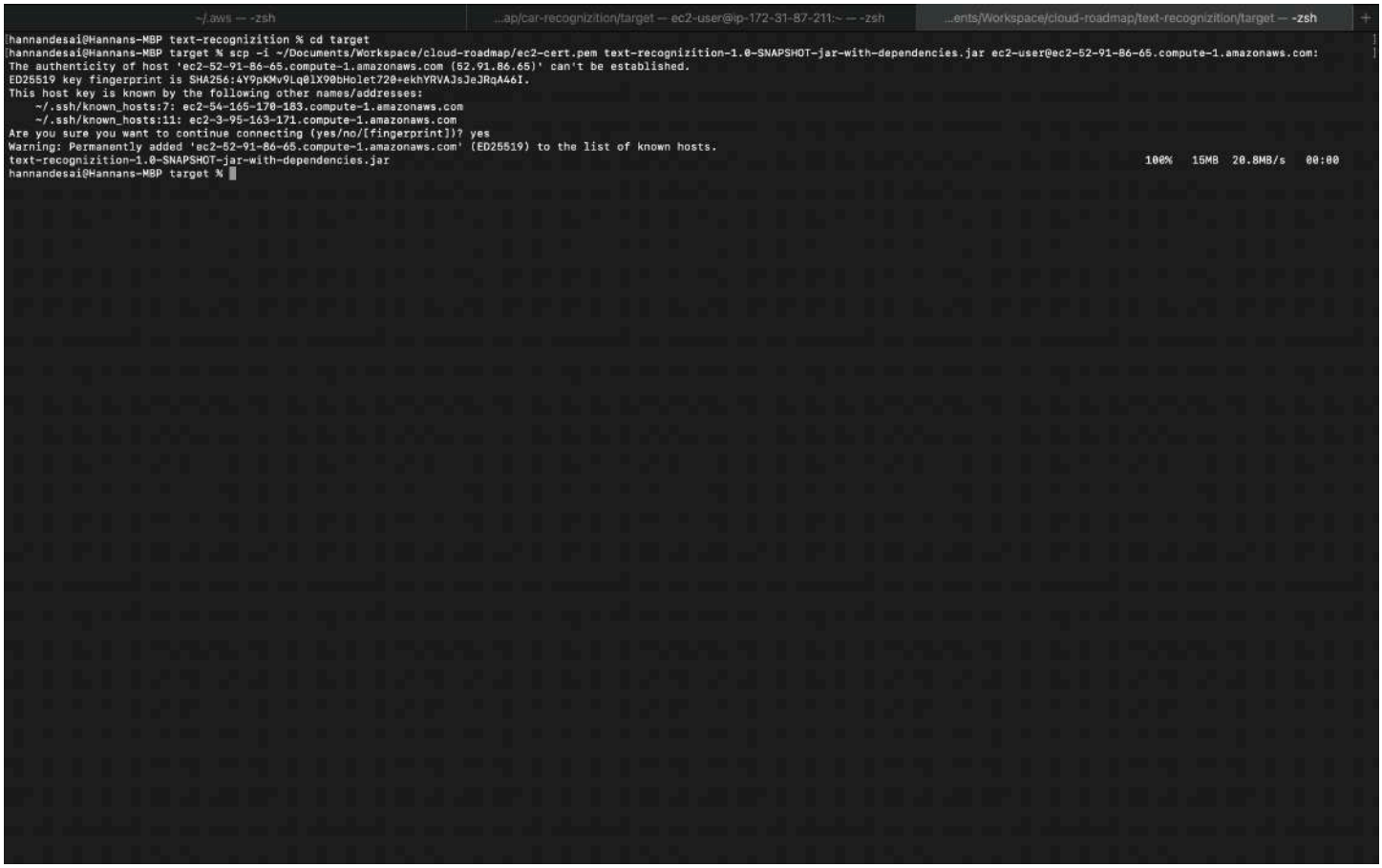
1) Go into directory where jar file is located, how to create jar file is shown in first page.

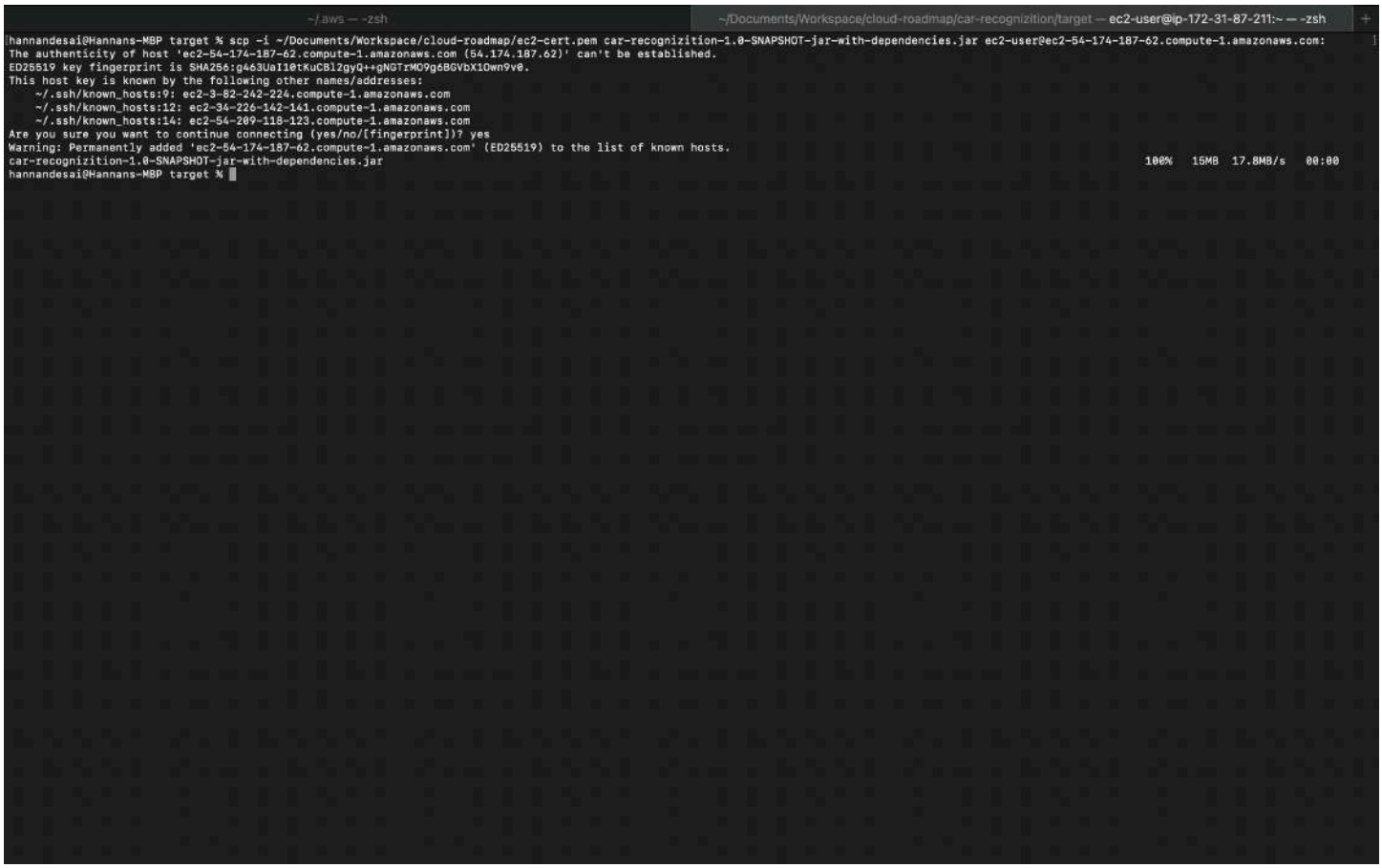
2) Use scp command to upload jar file from your local machine to EC2, run this command in terminal where we have connected EC2

   scp -i "ec2-cert.pem" car-recognizition-1.0-SNAPSHOT-jar-with-dependencies.jar ec2-user@ec2-54-165-170-183.compute-1.amazonaws.com:

   ec2-cert.pem is the cert file that we created while creating instance, if jar and .pem is not in same directory then give full path where is you .pem file is located in your machine.

   So run this scp command for both projects in their respective EC2 connected terminal.


~/.aws — -zsh
...ap/car-recognizition/target — ec2-user@ip-172-31-87-211:~ — -zsh
...ents/Workspace/cloud-roadmap/text-recognizition/target — -zsh
hannandesai@Hannans-MBP text-recognizition % cd target
hannandesai@Hannans-MBP target % scp -i ~/Documents/Workspace/cloud-roadmap/ec2-cert.pem text-recognizition-1.0-SNAPSHOT-jar-with-dependencies.jar ec2-user@ec2-52-91-86-65.compute-1.amazonaws.com:
The authenticity of host 'ec2-52-91-86-65.compute-1.amazonaws.com (52.91.86.65)' can't be established.
ED25519 key fingerprint is SHA256:4Y9pKMv9Lq0lX90bHolet720+ekhYRVAJsJeJRqA46I.
This host key is known by the following other names/addresses:
~/.ssh/known_hosts:7: ec2-54-165-170-183.compute-1.amazonaws.com
~/.ssh/known_hosts:11: ec2-3-95-163-171.compute-1.amazonaws.com
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-52-91-86-65.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
text-recognizition-1.0-SNAPSHOT-jar-with-dependencies.jar
100% 15MB 20.8MB/s 00:00
hannandesai@Hannans-MBP target %

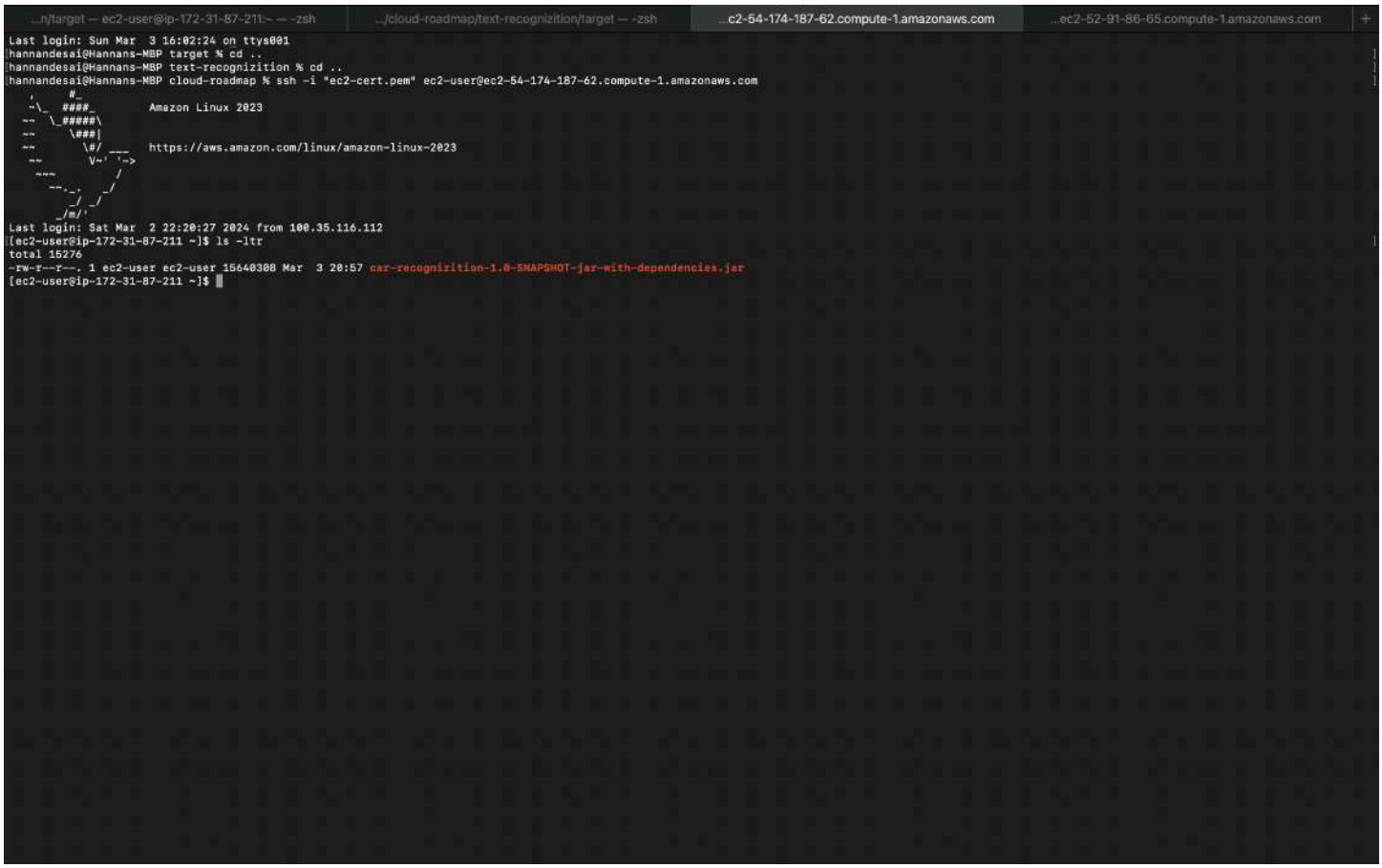


Now connect to EC-A & EC-B using SSH command as mentioned previously:

ssh -i "ec2-cert.pem" <Public IPv4 DNS>

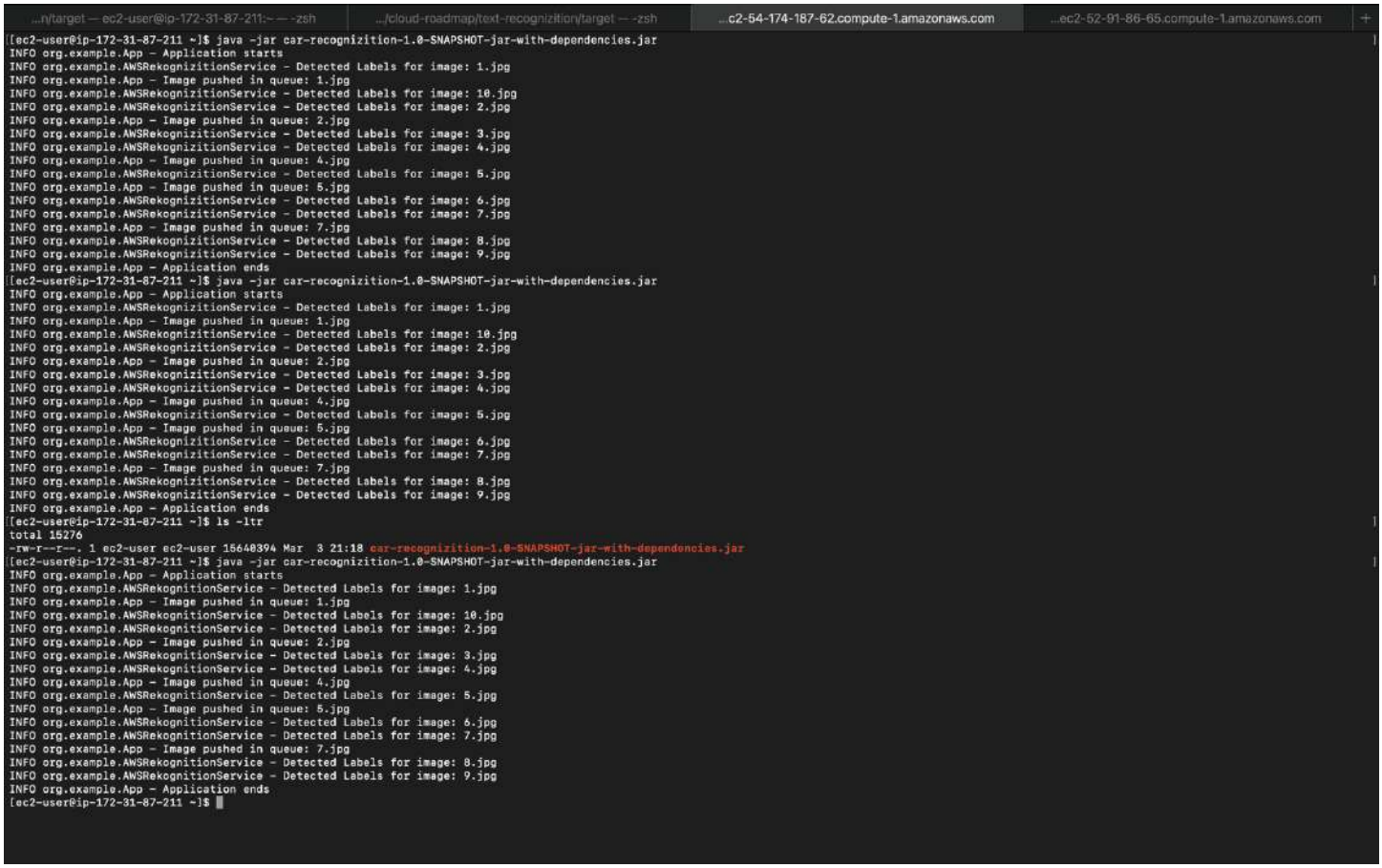
Run following command in both instances to see that file has been uploaded:

ls -ltr

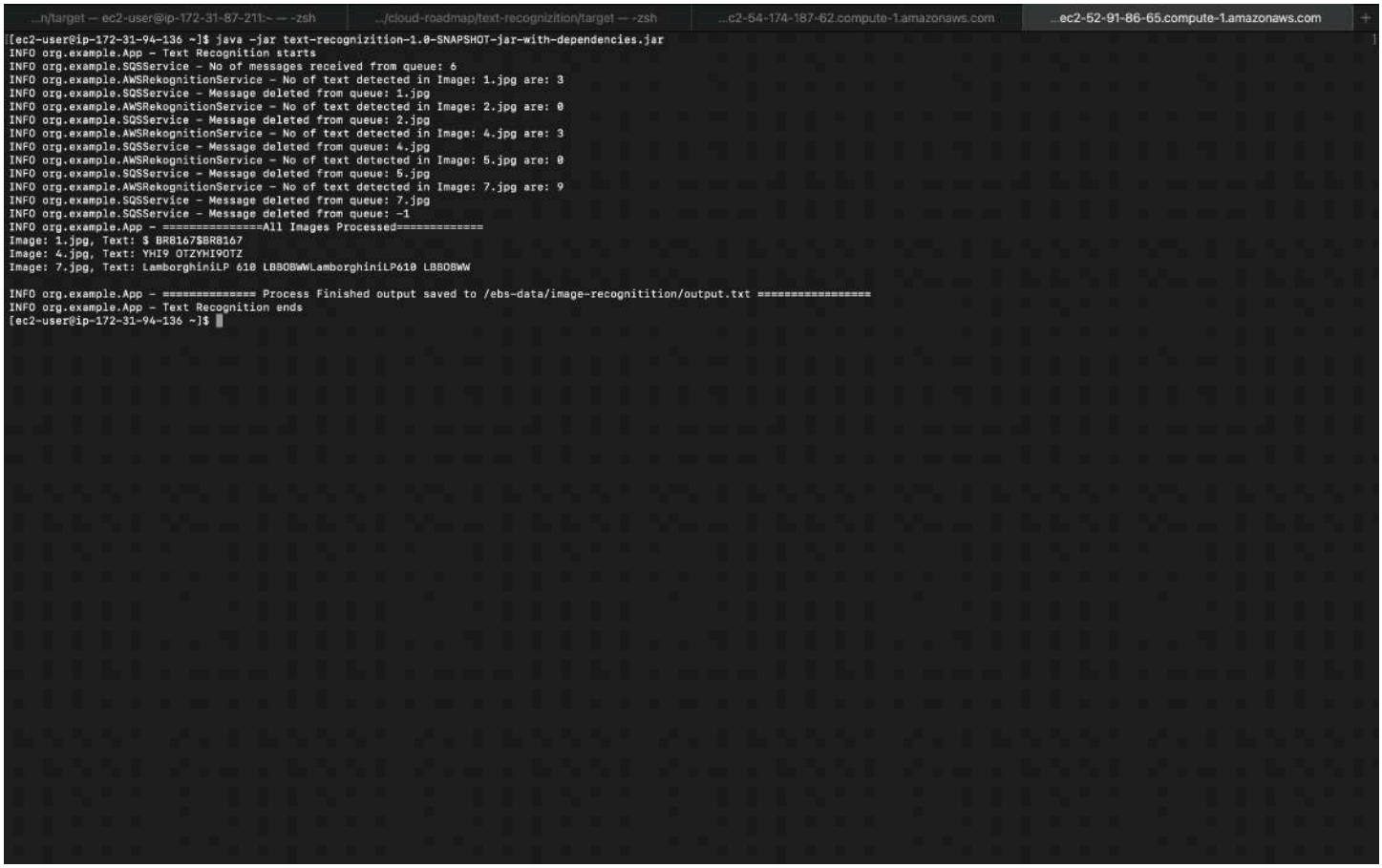

.../target — ec2-user@ip-172-31-87-211:~ — -zsh
.../cloud-roadmap/text-recognizition/target — -zsh
...c2-54-174-187-62.compute-1.amazonaws.com
...ec2-52-91-86-65.compute-1.amazonaws.com
Last login: Sun Mar 3 16:02:24 on ttys001
hannandesai@Hannans-MBP target % cd ..
hannandesai@Hannans-MBP text-recognizition % cd ..
hannandesai@Hannans-MBP cloud-roadmap % ssh -i "ec2-cert.pem" ec2-user@ec2-54-174-187-62.compute-1.amazonaws.com
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023
Last login: Sat Mar 2 22:20:27 2024 from 100.35.116.112
[ec2-user@ip-172-31-87-211 ~]$ ls -ltr
total 15276
-rw-r--r--. 1 ec2-user ec2-user 15640308 Mar 3 20:57 car-recognizition-1.0-SNAPSHOT-jar-with-dependencies.jar
[ec2-user@ip-172-31-87-211 ~]$

Run the jar file:

1) Run the car-recognizition first on EC-A, using command:
   java -jar car-recognizition-1.0-SNAPSHOT-jar-with-dependencies.jar



2) Then run text-recognizition on EC-B, using command:
   java -jar text-recognizition-1.0-SNAPSHOT-jar-with-dependencies.jar



So, now both the process finished and from the output image 1,4 & 7 have both car and text in it, output has been written in file format (.txt) on associated EBS directory “ebs-data/image-recognitition.output.txt”, this directory we have created previously.


...n/target1 — ec2-user@ip-172-31-87-211:~ — -zsh
.../cloud-roadmap/text-recognition/target — -zsh
...c2-54-174-187-62.compute-1.amazonaws.com
...ec2-52-91-86-65.compute-1.amazonaws.com
Image: 1.jpg, Text: $ BRB167$BRB167
Image: 4.jpg, Text: YHI9 OTZYHI9OTZ
Image: 7.jpg, Text: LamborghiniLP 610 LBBOBWWLamborghiniLP610 LBBOBWW
"output.txt" 3L, 146B
1,1
All