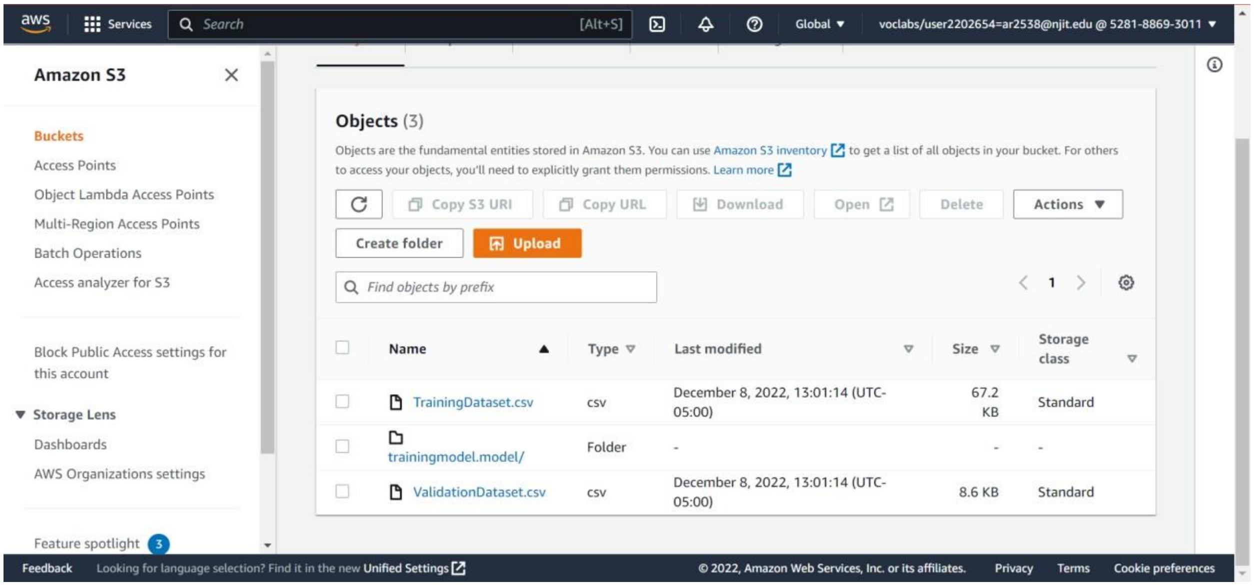
Cloud Computing Programming Assignment – 2

Prerequisite:

Created an S3 Bucket, and uploaded the training and validation data sets.

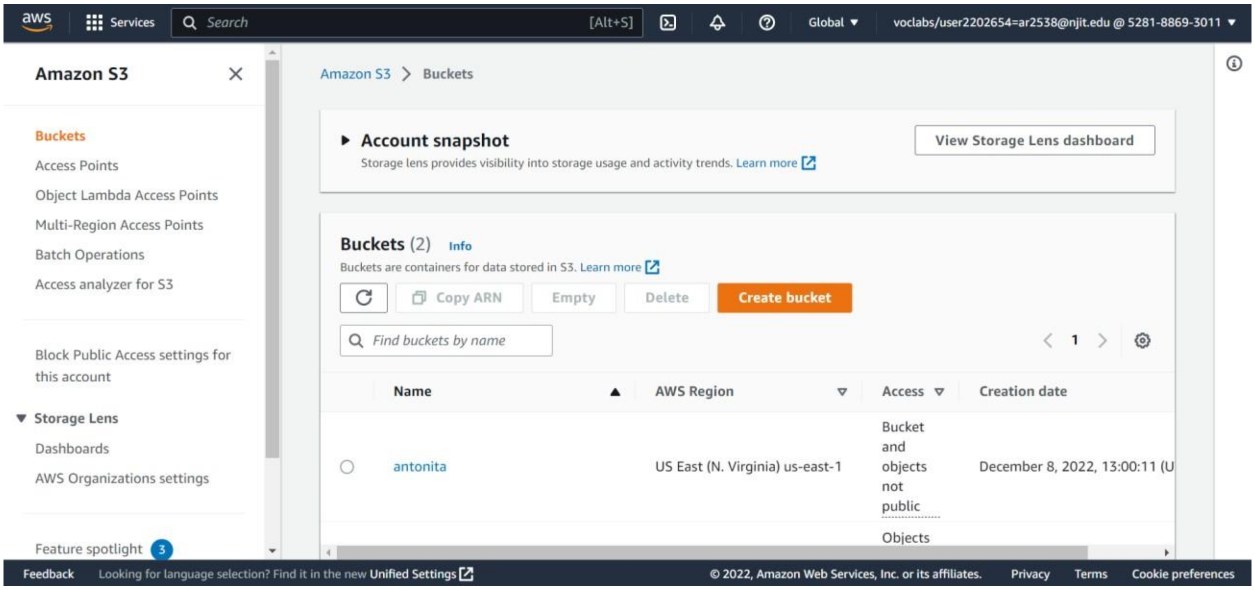

aws
Services
Search
[Alt+S]
Global
voclabs/user2202654=ar2538@njit.edu @ 5281-8869-3011
Amazon S3
Buckets
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
Access analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
AWS Organizations settings
Feature spotlight 3
Copy S3 URI
Copy URL
Download
Open
Delete
Actions
Create folder
Upload
Find objects by prefix
1
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Objects (3)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

	Name	Type	Last modified	Size	Storage class
□	TrainingDataset.csv	csv	December 8, 2022, 13:01:14 (UTC-05:00)	67.2 KB	Standard
□	trainingmodel.model/	Folder	-	-	-
□	ValidationDataset.csv	csv	December 8, 2022, 13:01:14 (UTC-05:00)	8.6 KB	Standard

The name of the bucket is antonita


aws
Services
Search
[Alt+S]
Global
voclabs/user2202654=ar2538@njit.edu @ 5281-8869-3011
Amazon S3
Buckets
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
Access analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
AWS Organizations settings
Feature spotlight 3
Amazon S3 > Buckets
View Storage Lens dashboard
Info
Copy ARN
Empty
Delete
Create bucket
Find buckets by name
1
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Account snapshot

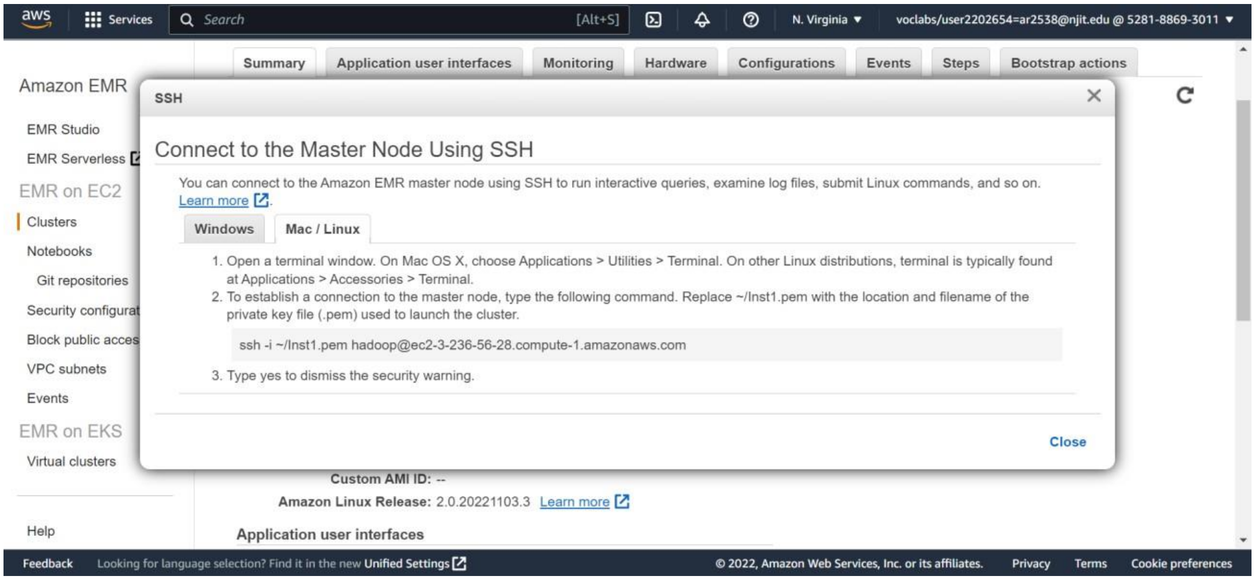
Storage lens provides visibility into storage usage and activity trends. Learn more

Buckets (2)

Buckets are containers for data stored in S3. Learn more

	Name	AWS Region	Access	Creation date
○	antonita	US East (N. Virginia) us-east-1	Bucket and objects not public	December 8, 2022, 13:00:11 (U
			Objects	

Task 1: Parallel Training on four EC2 Instances, with software configuration as Spark, and EC2 key pair as Inst1, it is my existing key pair from Programming Assignment 1.



Connect to the Master Node Using SSH

You can connect to the Amazon EMR master node using SSH to run interactive queries, examine log files, submit Linux commands, and so on. Learn more.

1. Open a terminal window. On Mac OS X, choose Applications > Utilities > Terminal. On other Linux distributions, terminal is typically found at Applications > Accessories > Terminal.
2. To establish a connection to the master node, type the following command. Replace ~/Inst1.pem with the location and filename of the private key file (.pem) used to launch the cluster.

```
ssh -i ~/Inst1.pem hadoop@ec2-3-236-56-28.compute-1.amazonaws.com
```

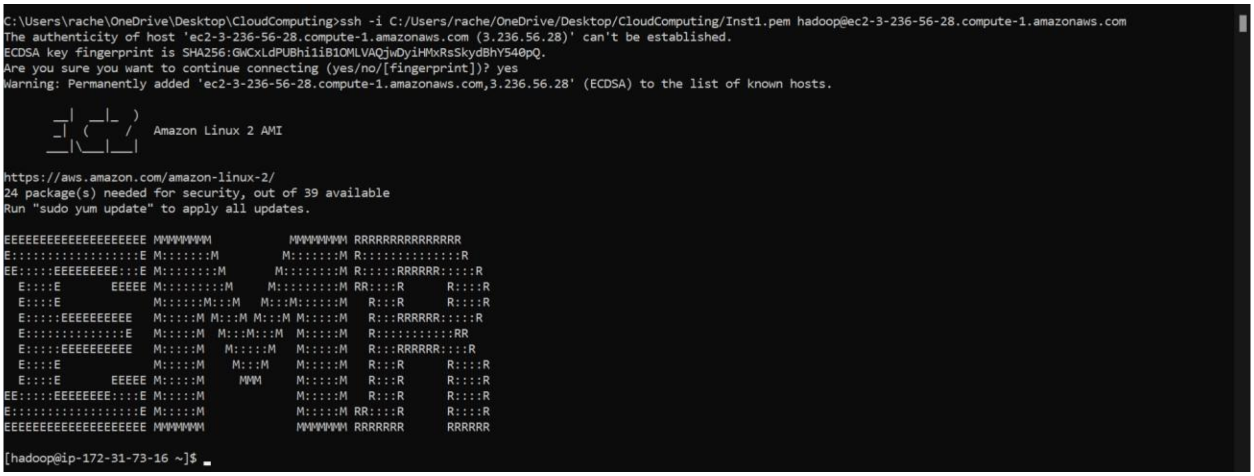
3. Type yes to dismiss the security warning.

Custom AMI ID: --

Amazon Linux Release: 2.0.20221103.3 Learn more

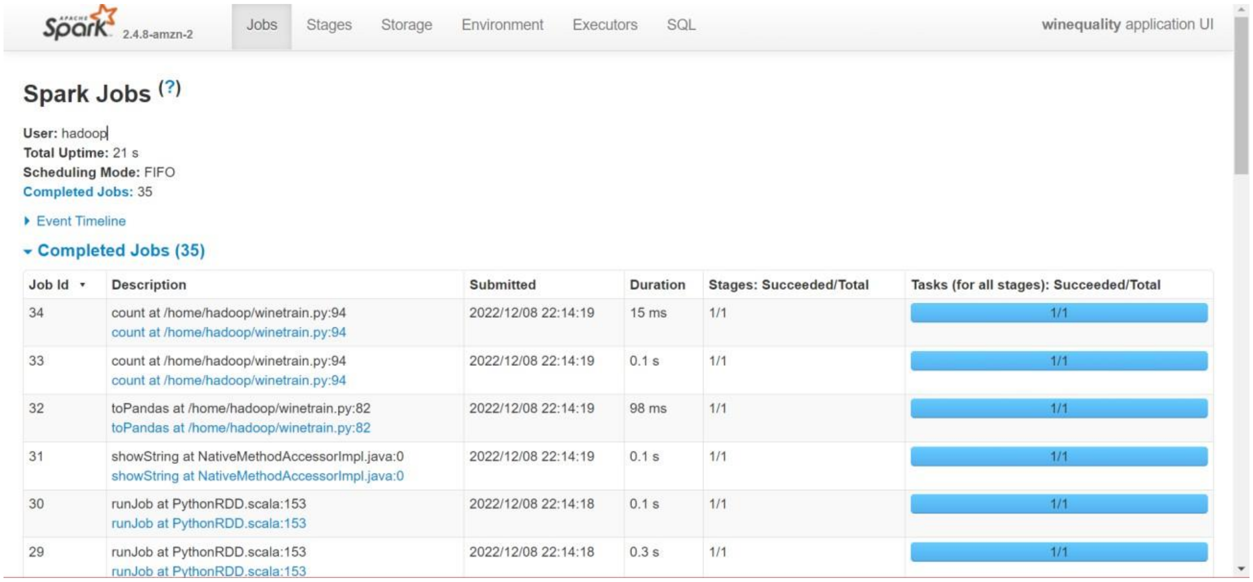
Application user interfaces

Copied the IP address and added the location of my Inst1.pem file to build my EMR connection.



```
C:\Users\rache\OneDrive\Desktop\CloudComputing>ssh -i C:/Users/rache/OneDrive/Desktop/CloudComputing/Inst1.pem hadoop@ec2-3-236-56-28.compute-1.amazonaws.com
The authenticity of host 'ec2-3-236-56-28.compute-1.amazonaws.com (3.236.56.28)' can't be established.
ECDSA key fingerprint is SHA256:GWCxLdPUBhi1iB1OMLVAQjwDyiHMxRsSkydBhY540pQ.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-3-236-56-28.compute-1.amazonaws.com,3.236.56.28' (ECDSA) to the list of known hosts.

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
24 package(s) needed for security, out of 39 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-172-31-73-16 ~]$ _
```

I created a python file and ran pip install find spark, and ran my code with spark-submit. The history server of my spark is displayed below:



Spark Jobs (?)

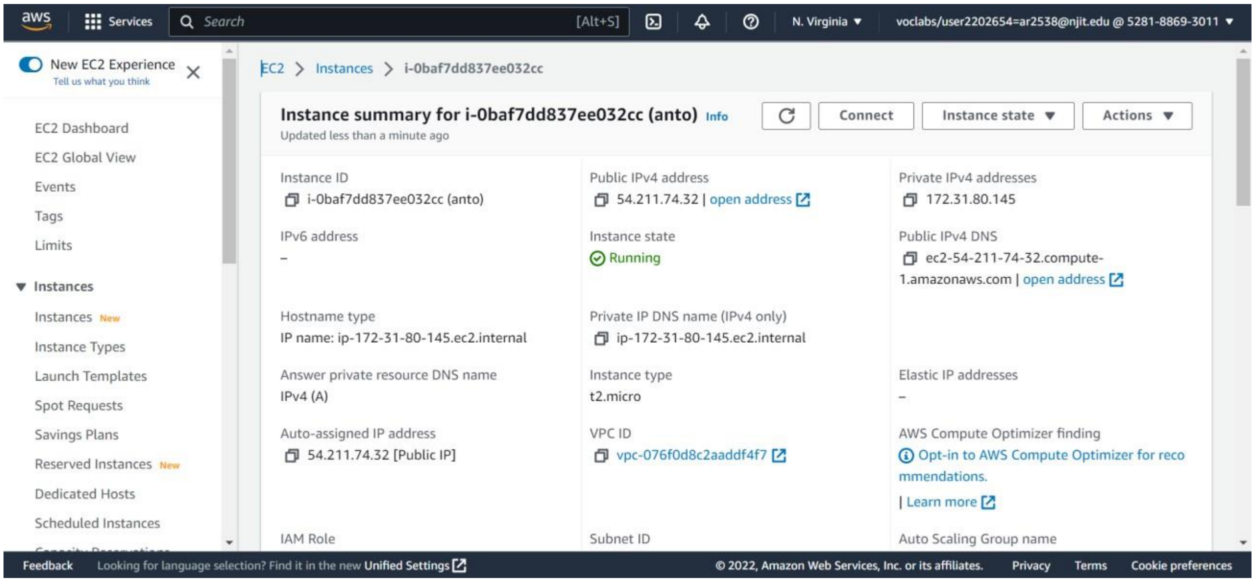
User: hadoop
Total Uptime: 21 s
Scheduling Mode: FIFO
Completed Jobs: 35

▸ Event Timeline

▾ Completed Jobs (35)

Job Id ▾	Description	Submitted	Duration	Stages: Succeeded/Total	Tasks (for all stages): Succeeded/Total
34	count at /home/hadoop/winetrain.py:94 count at /home/hadoop/winetrain.py:94	2022/12/08 22:14:19	15 ms	1/1	1/1
33	count at /home/hadoop/winetrain.py:94 count at /home/hadoop/winetrain.py:94	2022/12/08 22:14:19	0.1 s	1/1	1/1
32	toPandas at /home/hadoop/winetrain.py:82 toPandas at /home/hadoop/winetrain.py:82	2022/12/08 22:14:19	98 ms	1/1	1/1
31	showString at NativeMethodAccessorImpl.java:0 showString at NativeMethodAccessorImpl.java:0	2022/12/08 22:14:19	0.1 s	1/1	1/1
30	runJob at PythonRDD.scala:153 runJob at PythonRDD.scala:153	2022/12/08 22:14:18	0.1 s	1/1	1/1
29	runJob at PythonRDD.scala:153 runJob at PythonRDD.scala:153	2022/12/08 22:14:18	0.3 s	1/1	1/1

Task 2: Initiated a new Instance called anto



Instance summary for i-0baf7dd837ee032cc (anto) Info
Updated less than a minute ago

Instance ID
i-0baf7dd837ee032cc (anto)

Public IPv4 address
54.211.74.32 | open address

Private IPv4 addresses
172.31.80.145

IPv6 address
–

Instance state
Running

Public IPv4 DNS
ec2-54-211-74-32.compute-1.amazonaws.com | open address

Hostname type
IP name: ip-172-31-80-145.ec2.internal

Private IP DNS name (IPv4 only)
ip-172-31-80-145.ec2.internal

Answer private resource DNS name
IPv4 (A)

Instance type
t2.micro

Elastic IP addresses
–

Auto-assigned IP address
54.211.74.32 [Public IP]

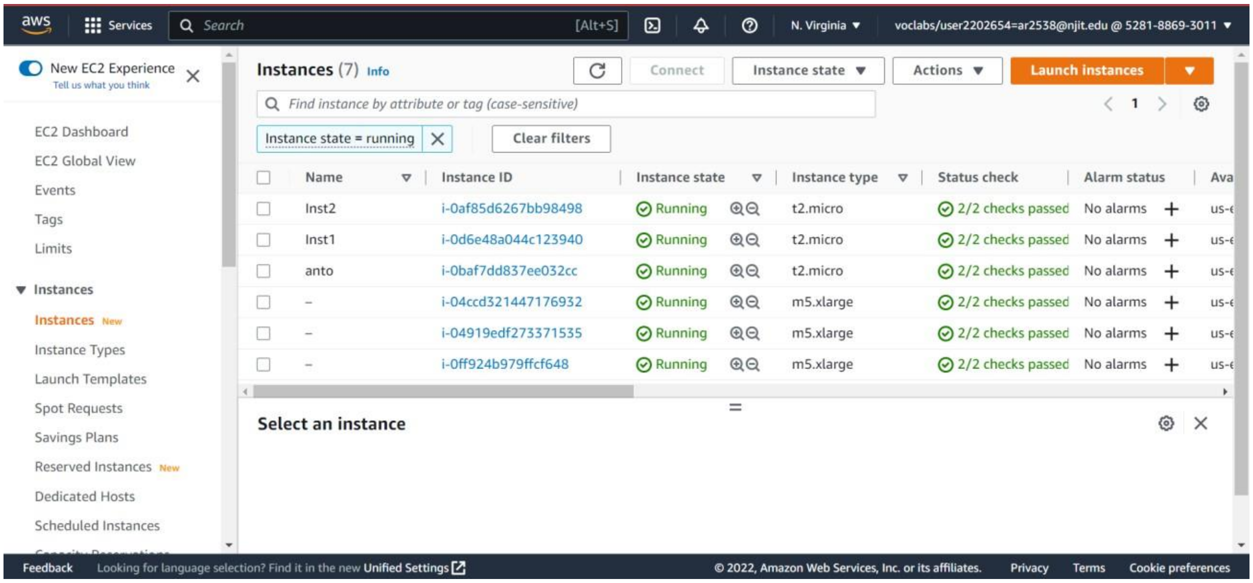
VPC ID
vpc-076f0d8c2aaddf4f7

AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations.
| Learn more

IAM Role

Subnet ID

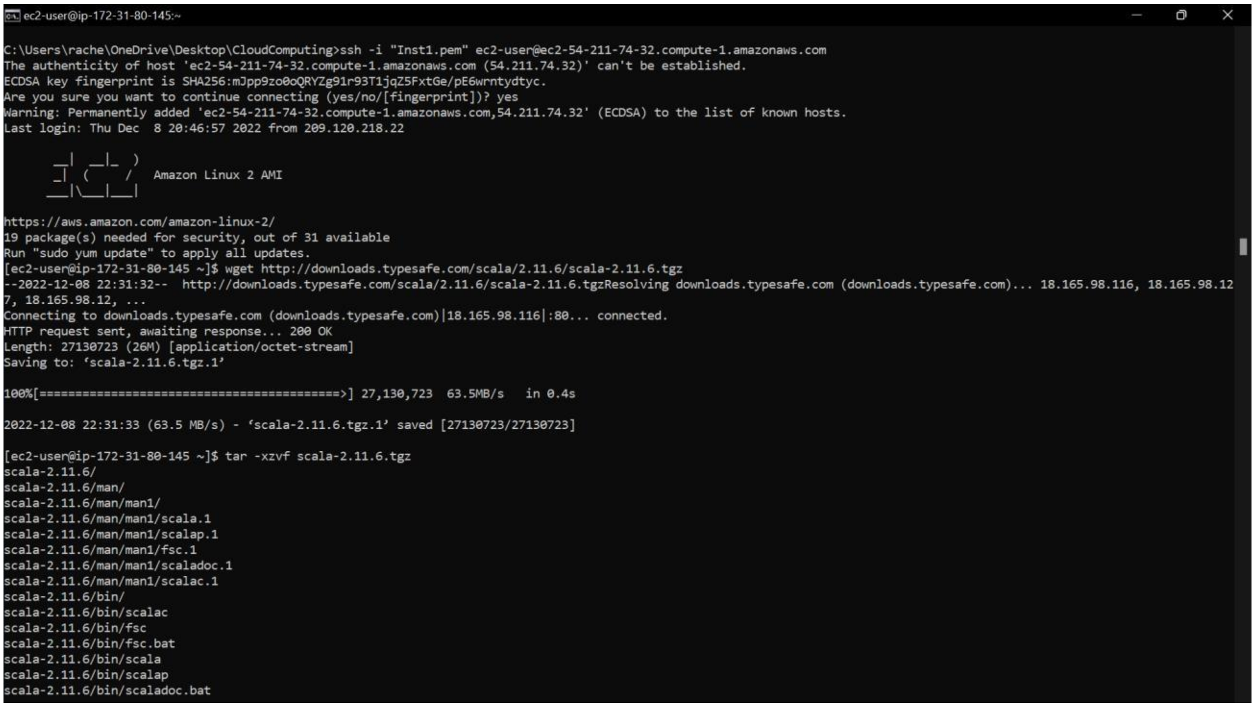
Auto Scaling Group name



Instances (7) Info

□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Ava
□	Inst2	i-0af85d6267bb98498	✓ Running	t2.micro	✓ 2/2 checks passed	No alarms +	us-e
□	Inst1	i-0d6e48a044c123940	✓ Running	t2.micro	✓ 2/2 checks passed	No alarms +	us-e
□	anto	i-0baf7dd837ee032cc	✓ Running	t2.micro	✓ 2/2 checks passed	No alarms +	us-e
□	–	i-04ccd321447176932	✓ Running	m5.xlarge	✓ 2/2 checks passed	No alarms +	us-e
□	–	i-04919edf273371535	✓ Running	m5.xlarge	✓ 2/2 checks passed	No alarms +	us-e
□	–	i-0ff924b979ffcf648	✓ Running	m5.xlarge	✓ 2/2 checks passed	No alarms +	us-e

Built the connection with my local terminal for the EC2 Instance



```
C:\Users\rache\OneDrive\Desktop\CloudComputing>ssh -i "Inst1.pem" ec2-user@ec2-54-211-74-32.compute-1.amazonaws.com
The authenticity of host 'ec2-54-211-74-32.compute-1.amazonaws.com (54.211.74.32)' can't be established.
ECDSA key fingerprint is SHA256:mJpp9zo0oQRYZg91r93T1jqZ5FxtGe/pE6wrntydtyc.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-211-74-32.compute-1.amazonaws.com,54.211.74.32' (ECDSA) to the list of known hosts.
Last login: Thu Dec  8 20:46:57 2022 from 209.120.218.22

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
19 package(s) needed for security, out of 31 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-80-145 ~]$ wget http://downloads.typesafe.com/scala/2.11.6/scala-2.11.6.tgz
--2022-12-08 22:31:32--  http://downloads.typesafe.com/scala/2.11.6/scala-2.11.6.tgzResolving downloads.typesafe.com (downloads.typesafe.com)... 18.165.98.116, 18.165.98.12
7, 18.165.98.12, ...
Connecting to downloads.typesafe.com (downloads.typesafe.com)|18.165.98.116|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 27130723 (26M) [application/octet-stream]
Saving to: ‘scala-2.11.6.tgz.1’

100%[==========================================>] 27,130,723  63.5MB/s   in 0.4s

2022-12-08 22:31:33 (63.5 MB/s) - ‘scala-2.11.6.tgz.1’ saved [27130723/27130723]

[ec2-user@ip-172-31-80-145 ~]$ tar -xzvf scala-2.11.6.tgz
scala-2.11.6/
scala-2.11.6/man/
scala-2.11.6/man/man1/
scala-2.11.6/man/man1/scala.1
scala-2.11.6/man/man1/scalap.1
scala-2.11.6/man/man1/fsc.1
scala-2.11.6/man/man1/scaladoc.1
scala-2.11.6/man/man1/scalac.1
scala-2.11.6/bin/
scala-2.11.6/bin/scalac
scala-2.11.6/bin/fsc
scala-2.11.6/bin/fsc.bat
scala-2.11.6/bin/scala
scala-2.11.6/bin/scalap
scala-2.11.6/bin/scaladoc.bat
```

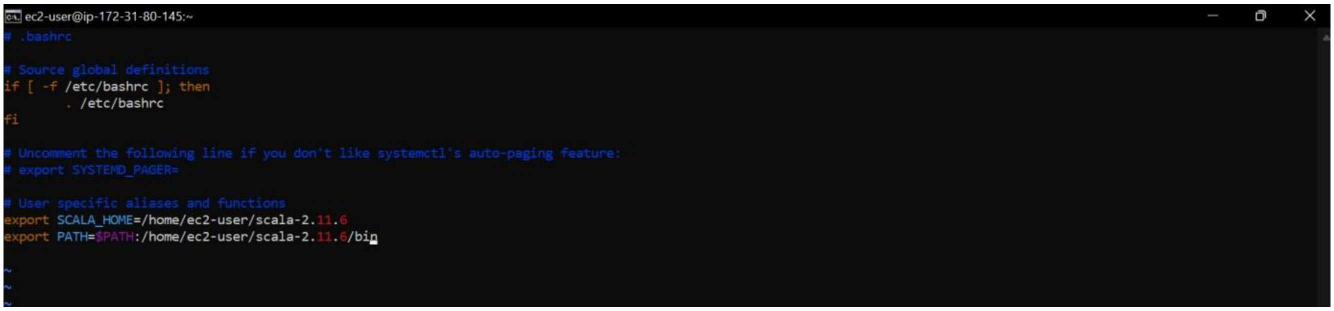
Install scala.

wget http://downloads.typesafe.com/scala/2.11.6/scala-2.11.6.tgz

tar -xzvf scala-2.11.6.tgz

- ✓ Update PATH environment variable:
  - vim ~/.bashrc
  - copy following lines into file and then save it
    - export SCALA_HOME=/home/ec2-user/scala-2.11.6
    - export PATH=$PATH:/home/ec2-user/scala-2.11.6/bin

source ~/.bashrc



```
# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
        . /etc/bashrc
fi

# Uncomment the following line if you don't like systemctl's auto-paging feature:
# export SYSTEMD_PAGER=

# User specific aliases and functions
export SCALA_HOME=/home/ec2-user/scala-2.11.6
export PATH=$PATH:/home/ec2-user/scala-2.11.6/bin

~
~
~
```

Installed Spark using following commands:

- ✓ wget https://archive.apache.org/dist/spark/spark-2.4.5/spark-2.4.5-bin-hadoop2.7.tgz
- ✓ sudo tar xvf spark-2.4.5-bin-hadoop2.7.tgz -C /opt
- ✓ sudo chown -R ec2-user:ec2-user /opt/spark-2.4.5-bin-hadoop2.7
- ✓ sudo ln -fs spark-2.4.5-bin-hadoop2.7 /opt/spark
- ✓ Update PATH Environment

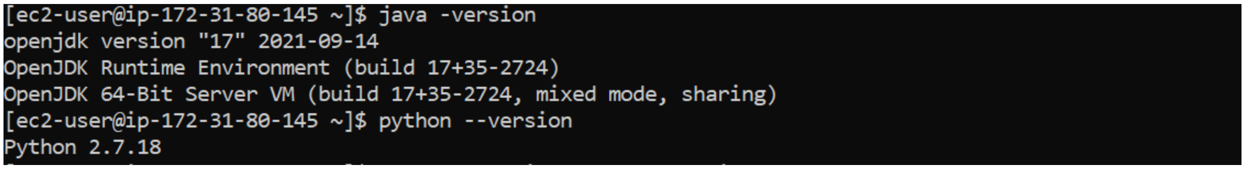
  $ vim ~/.bash_profile

  copy following lines into file and then save it

  ```
  export SPARK_HOME=/opt/spark
  PATH=$PATH:$SPARK_HOME/bin
  export  PATH
  ```

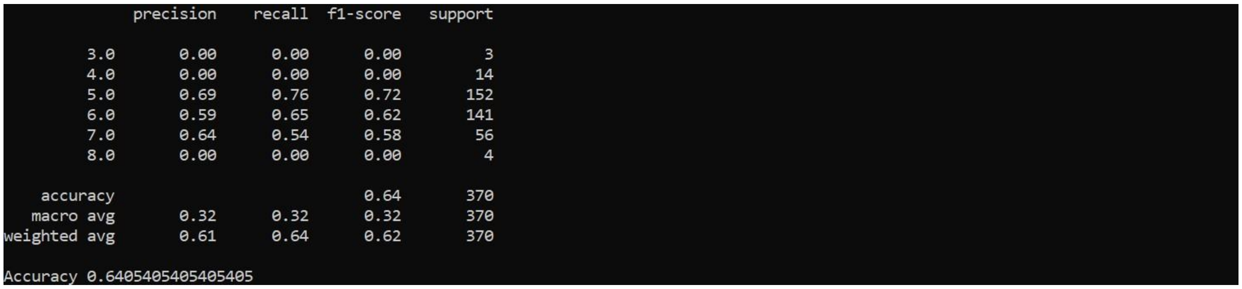
$ source ~/.bash_profile

Checked java –version/ Checked python version



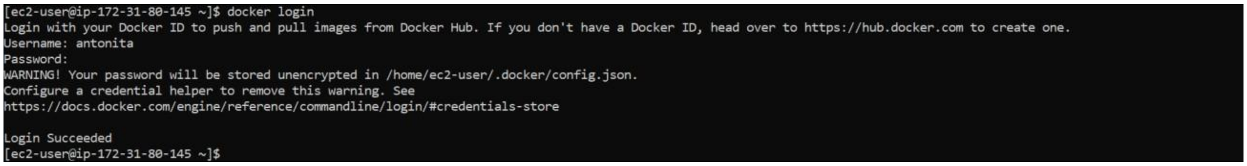
```
[ec2-user@ip-172-31-80-145 ~]$ java -version
openjdk version "17" 2021-09-14
OpenJDK Runtime Environment (build 17+35-2724)
OpenJDK 64-Bit Server VM (build 17+35-2724, mixed mode, sharing)
[ec2-user@ip-172-31-80-145 ~]$ python --version
Python 2.7.18
```

Installed libraries like pandas, scikit-learn, and NumPy and delivered the accuracy shown below:

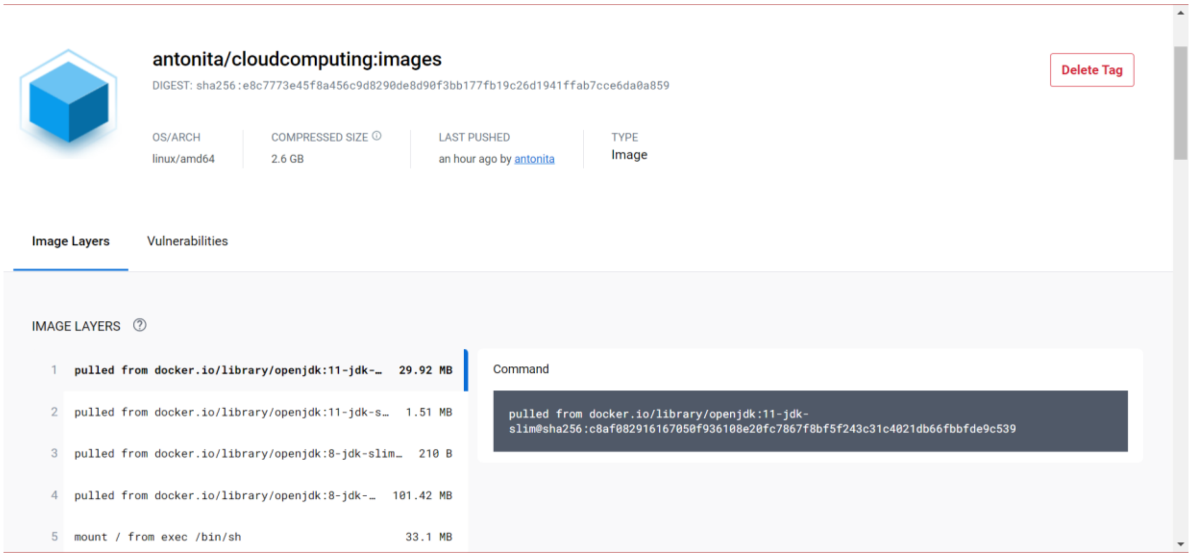


```
              precision    recall  f1-score   support

         3.0       0.00      0.00      0.00         3
         4.0       0.00      0.00      0.00        14
         5.0       0.69      0.76      0.72       152
         6.0       0.59      0.65      0.62       141
         7.0       0.64      0.54      0.58        56
         8.0       0.00      0.00      0.00         4

    accuracy                           0.64       370
   macro avg       0.32      0.32      0.32       370
weighted avg       0.61      0.64      0.62       370

Accuracy 0.6405405405405405
```

Task 3: Created a Docker account and built a repository called cloud computing to push and pull commands, also logged in to Docker via the EC2 instance in my local terminal.



```
[ec2-user@ip-172-31-80-145 ~]$ docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com to create one.
Username: antonita
Password:
WARNING! Your password will be stored unencrypted in /home/ec2-user/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
[ec2-user@ip-172-31-80-145 ~]$
```



antonita/cloudcomputing:images

DIGEST: sha256:e8c7773e45f8a456c9d8290de8d90f3bb177fb19c26d1941ffab7cce6da0a859

OS/ARCH: linux/amd64

COMPRESSED SIZE: 2.6 GB

LAST PUSHED: an hour ago by antonita

TYPE: Image

IMAGE LAYERS

```
1  pulled from docker.io/library/openjdk:11-jdk-…    29.92 MB
2  pulled from docker.io/library/openjdk:11-jdk-s…    1.51 MB
3  pulled from docker.io/library/openjdk:8-jdk-slim…   210 B
4  pulled from docker.io/library/openjdk:8-jdk-…     101.42 MB
5  mount / from exec /bin/sh                          33.1 MB
```

Command

```
pulled from docker.io/library/openjdk:11-jdk-
slim@sha256:c8af082916167050f936108e20fc7867f8bf5f243c31c4021db66fbbfde9c539
```

Links:

GitHub - https://github.com/ARacheal/Cloud-Computing-

Docker - https://hub.docker.com/repository/docker/antonita/cloudcomputing