Programming Assignment 2

Name : Himaniben Ashokbhai Patel

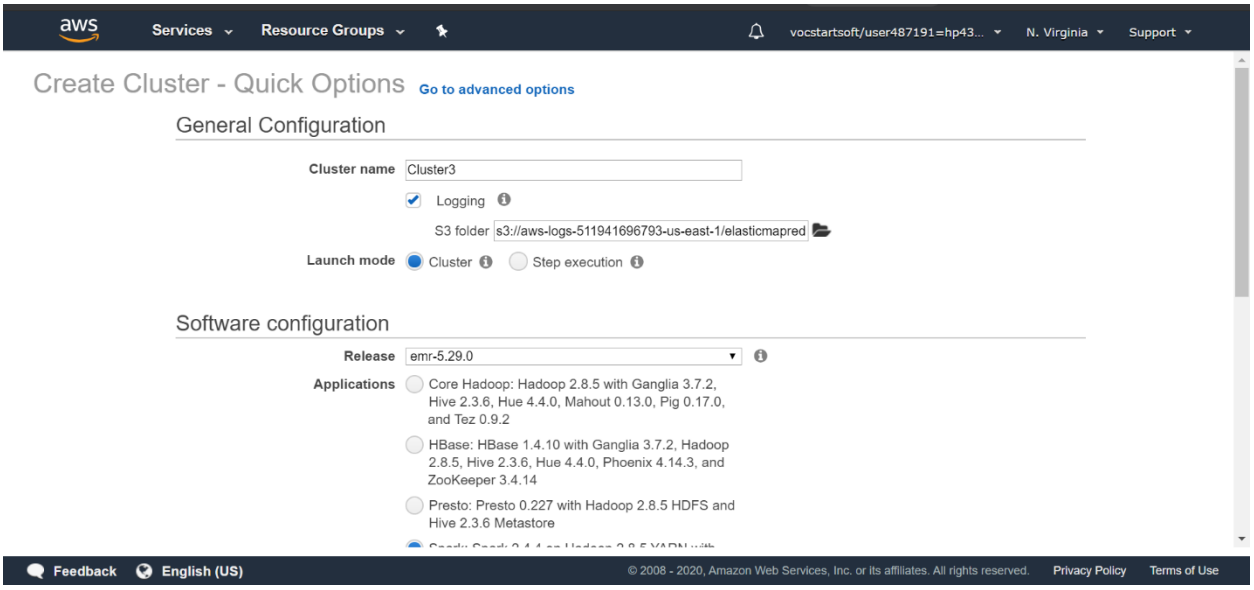
UCID : Hp437

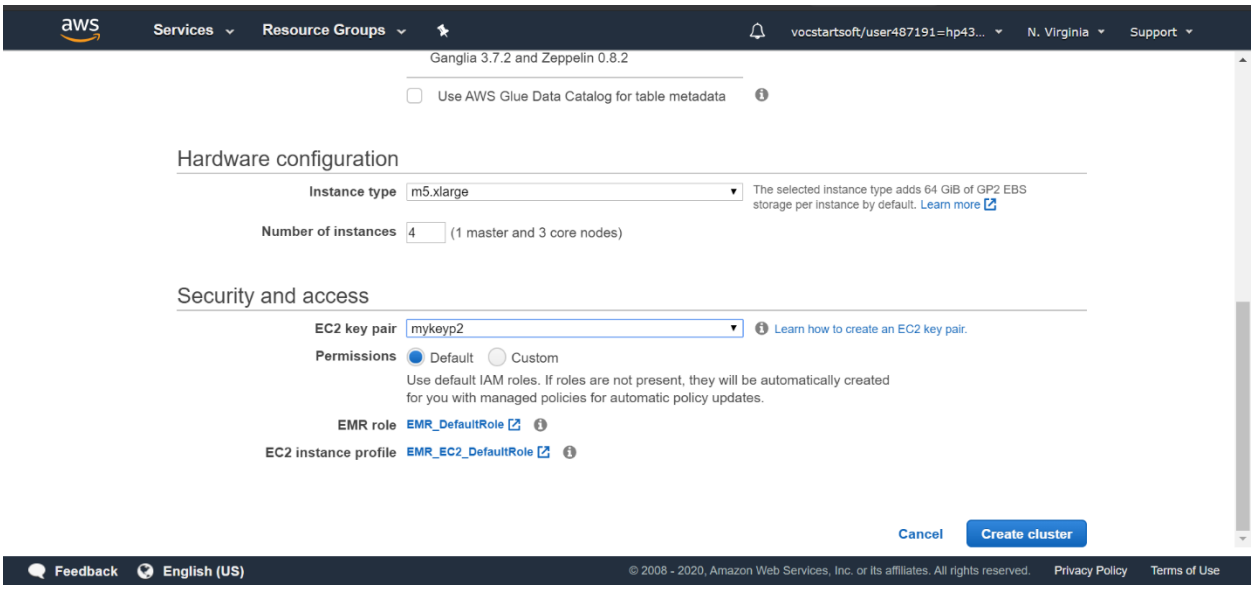
Email Id : hp437@njit.edu

Goal: The purpose of this individual assignment is to learn how to develop parallel machine learning (ML) applications in Amazon AWS cloud platform.

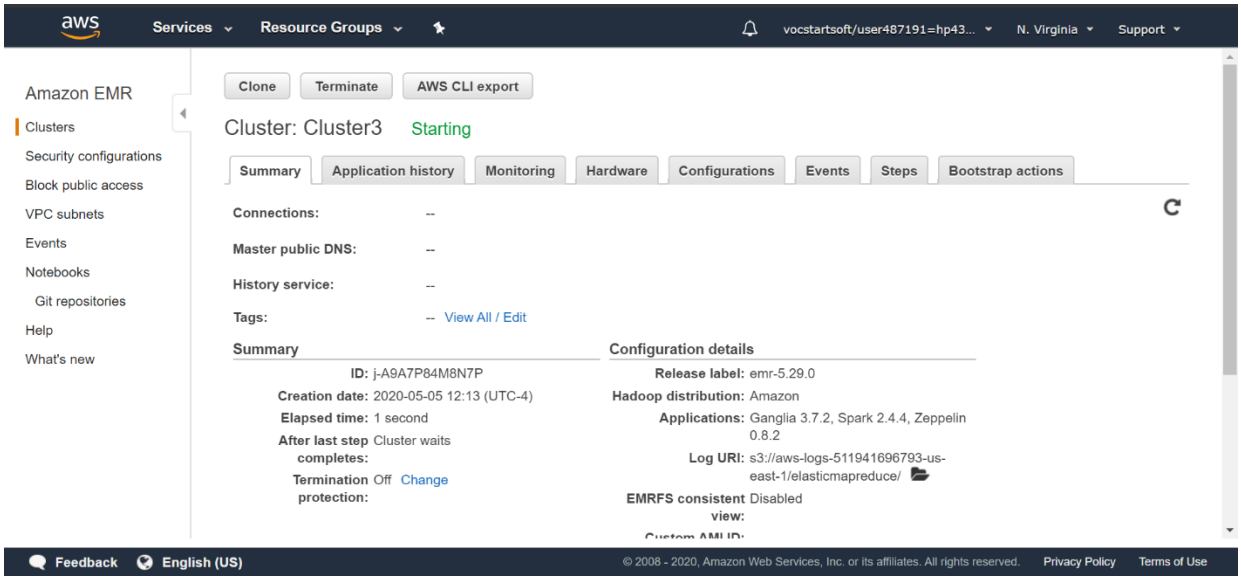
A wine quality prediction ML model in Spark over AWS. Here spark cluster is created by EMR. Amazon EMR is the industry-leading cloud big data platform for processing vast amounts of data using open source tools such as Apace Spark. Here **Decision Tree Model** is used to predict the model. Here I couldn't implement docker. This describes step by step guide to run without docker.

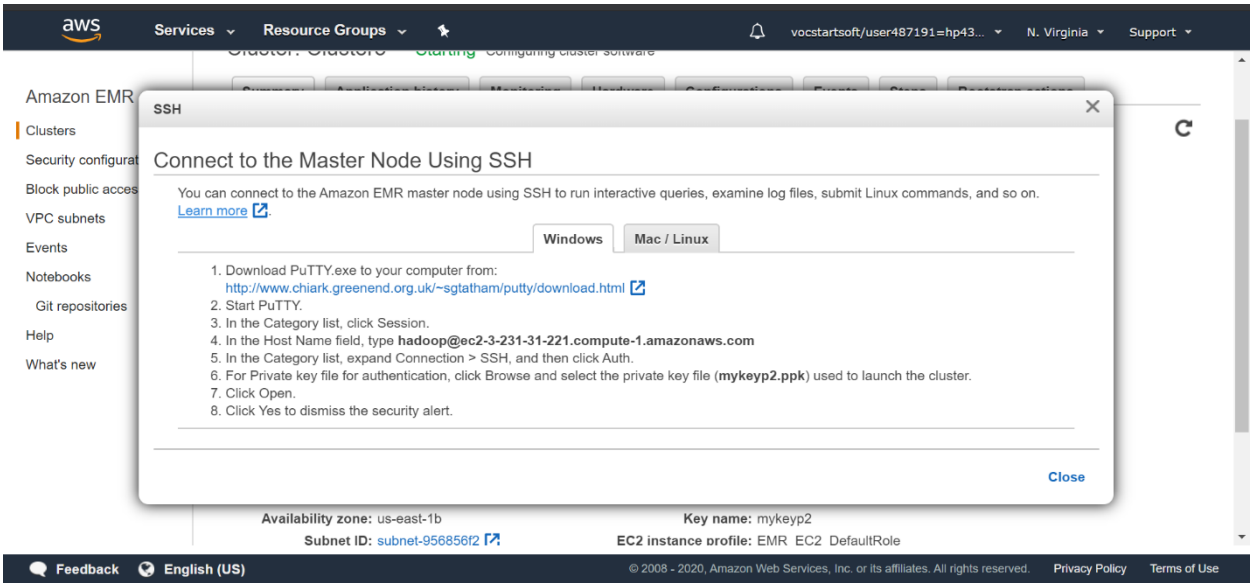
- Create spark cluster having **4 instances** in that and with KeyPair named **mykeyp2.pem**.

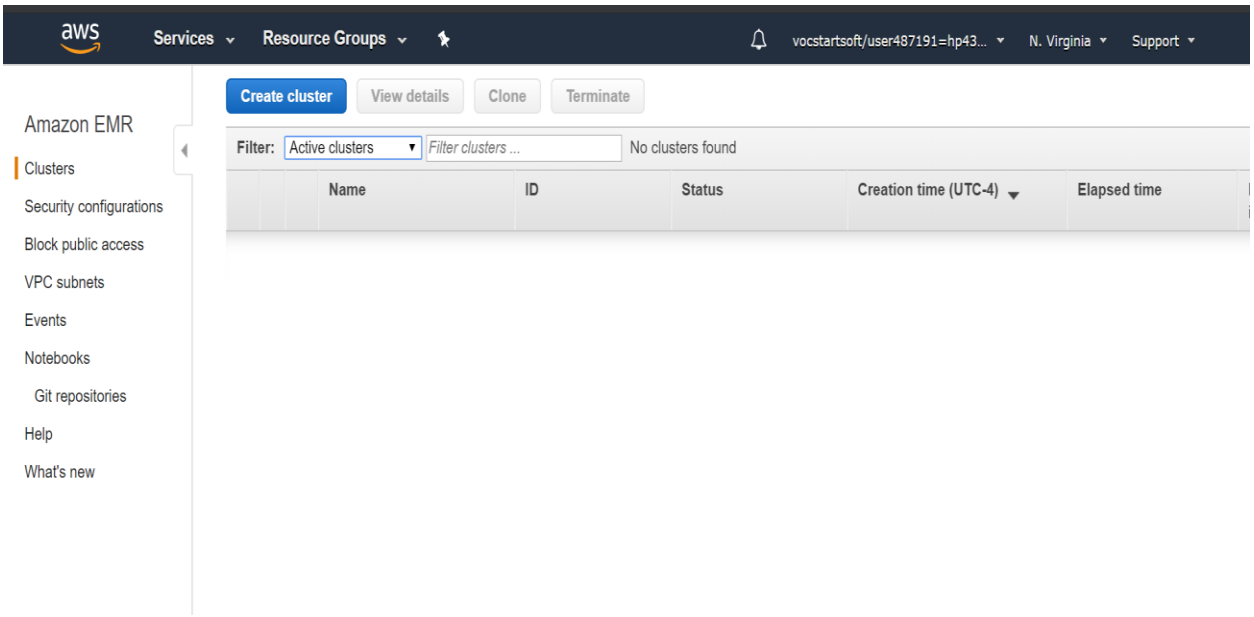


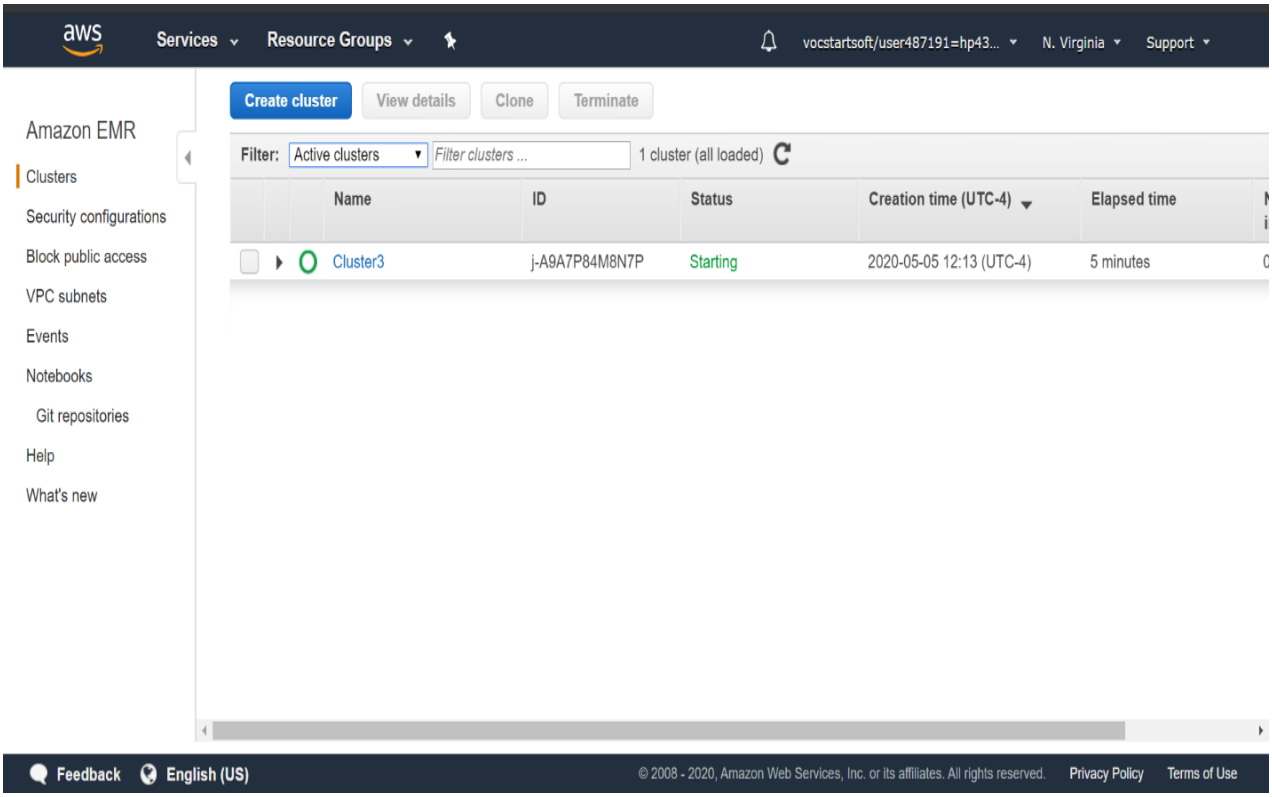


- After creating cluster it will look like as shown below. It has host name with that we can log in.

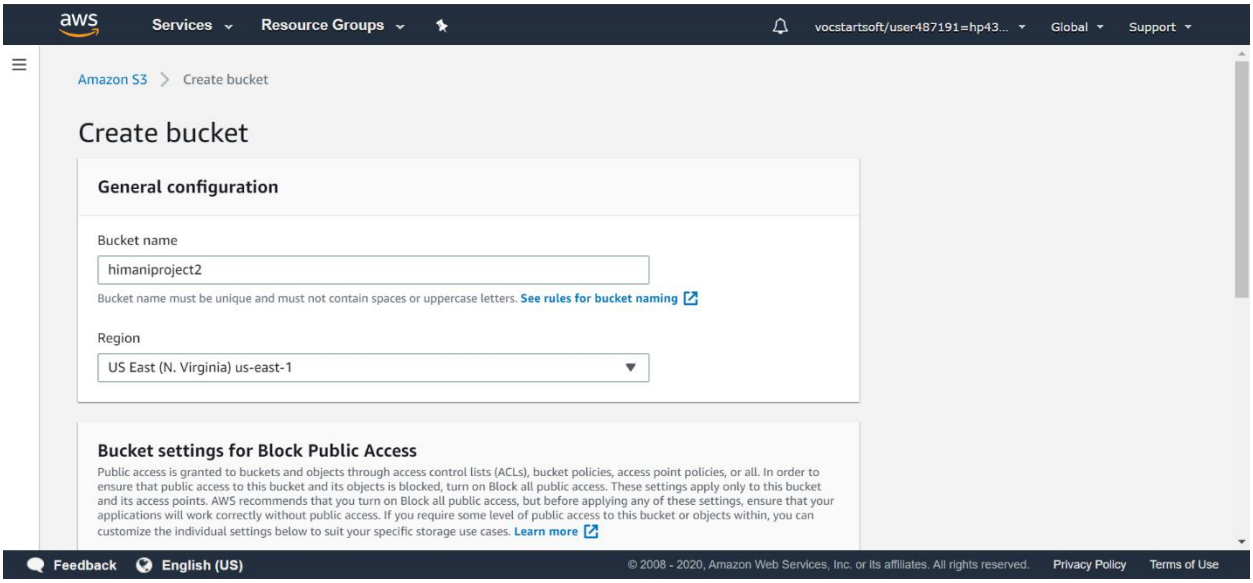


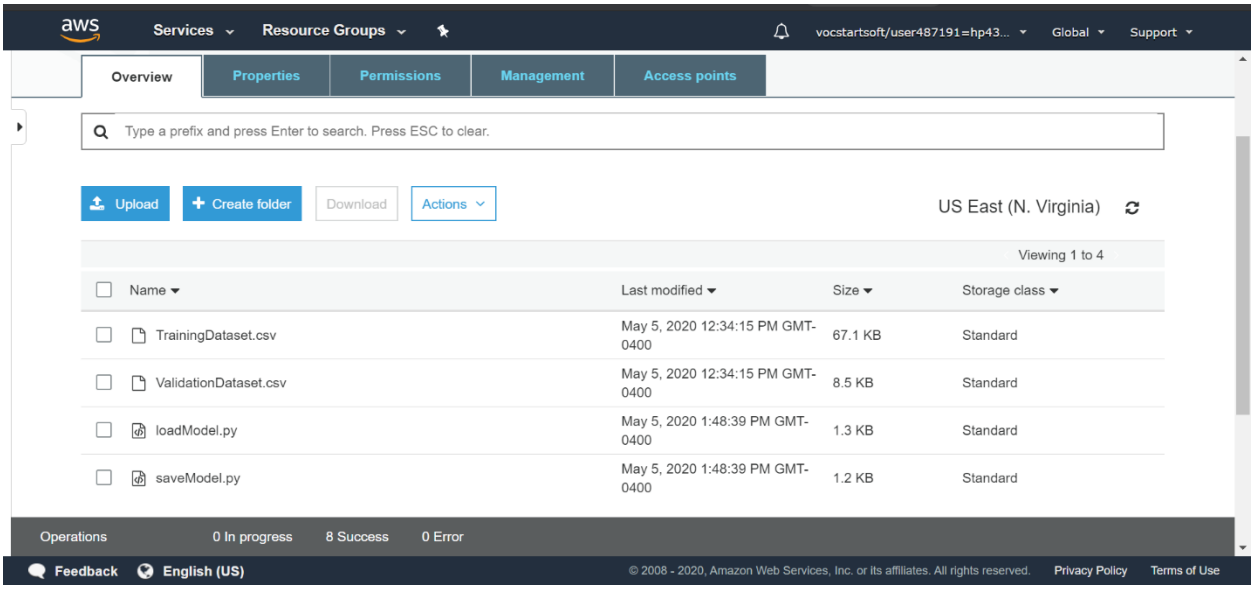



aws
Services
Resource Groups
vocstartsoft/user487191=hp43...
N. Virginia
Support
Amazon EMR
Clusters
Security configurations
Block public access
VPC subnets
Events
Notebooks
Git repositories
Help
What's new
Create cluster
View details
Clone
Terminate
Filter:
Active clusters
Filter clusters ...
No clusters found
Name
ID
Status
Creation time (UTC-4)
Elapsed time


aws
Services
Resource Groups
vocstartsoft/user487191=hp43...
N. Virginia
Support
Amazon EMR
Clusters
Security configurations
Block public access
VPC subnets
Events
Notebooks
Git repositories
Help
What's new
Create cluster
View details
Clone
Terminate
Filter:
Active clusters
Filter clusters ...
1 cluster (all loaded)
Name
ID
Status
Creation time (UTC-4)
Elapsed time
Cluster3
j-A9A7P84M8N7P
Starting
2020-05-05 12:13 (UTC-4)
5 minutes
Feedback
English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

- Create bucket named **himaniProject2** for storing dataset and python program.

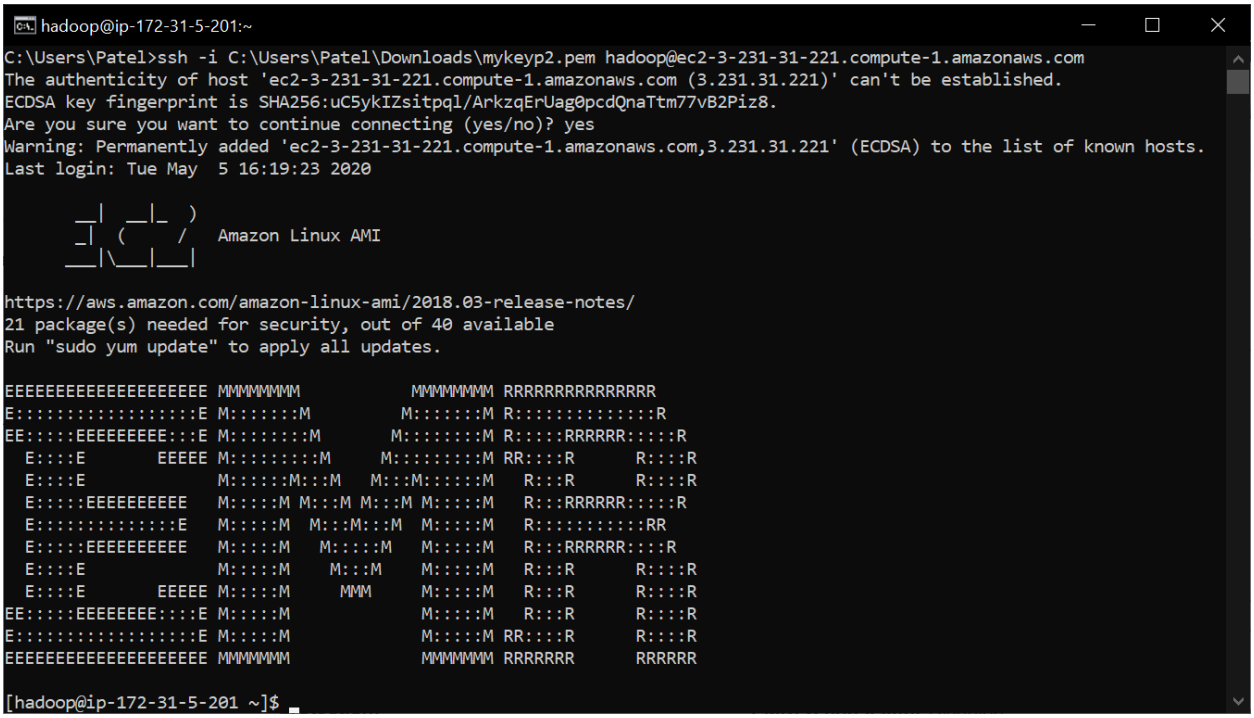




- Now login with cluster host name

  ssh – i path_to_keypair host_name

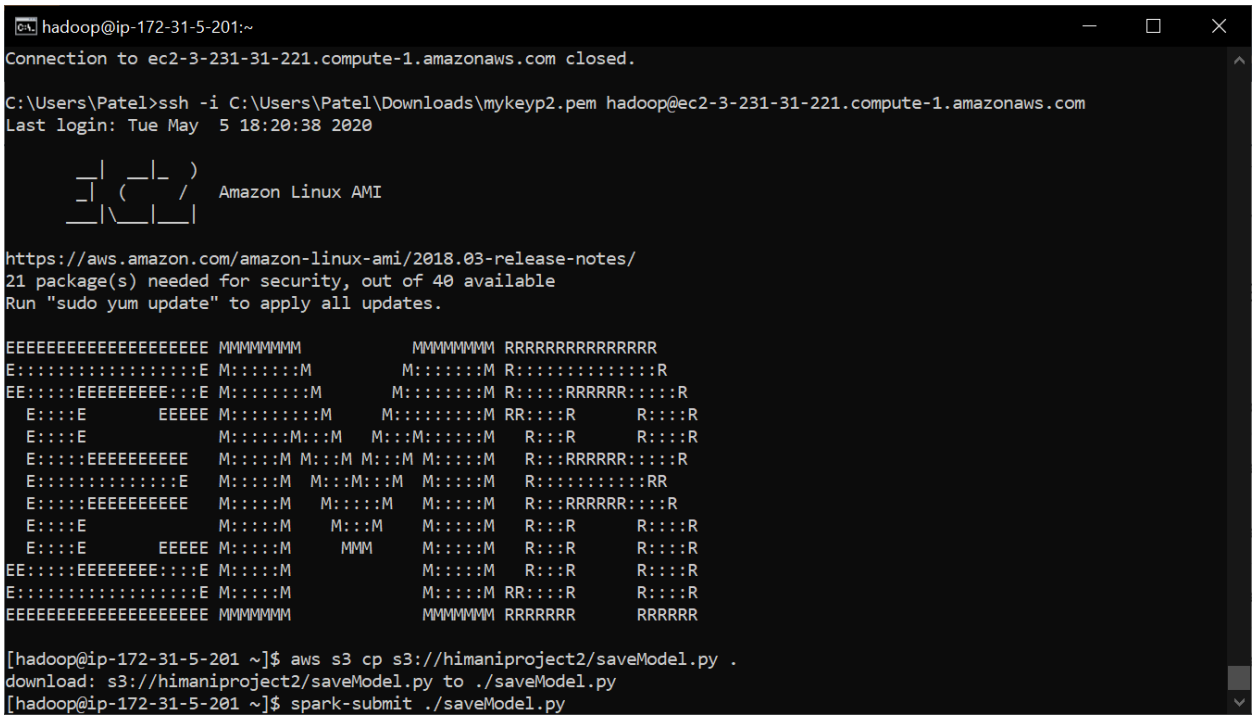
  ssh – i C:\Users\Patel\Downloads\mykeyp2.pem hadoop@ec2-34-239-157-134.compute-1.amazonaws.com

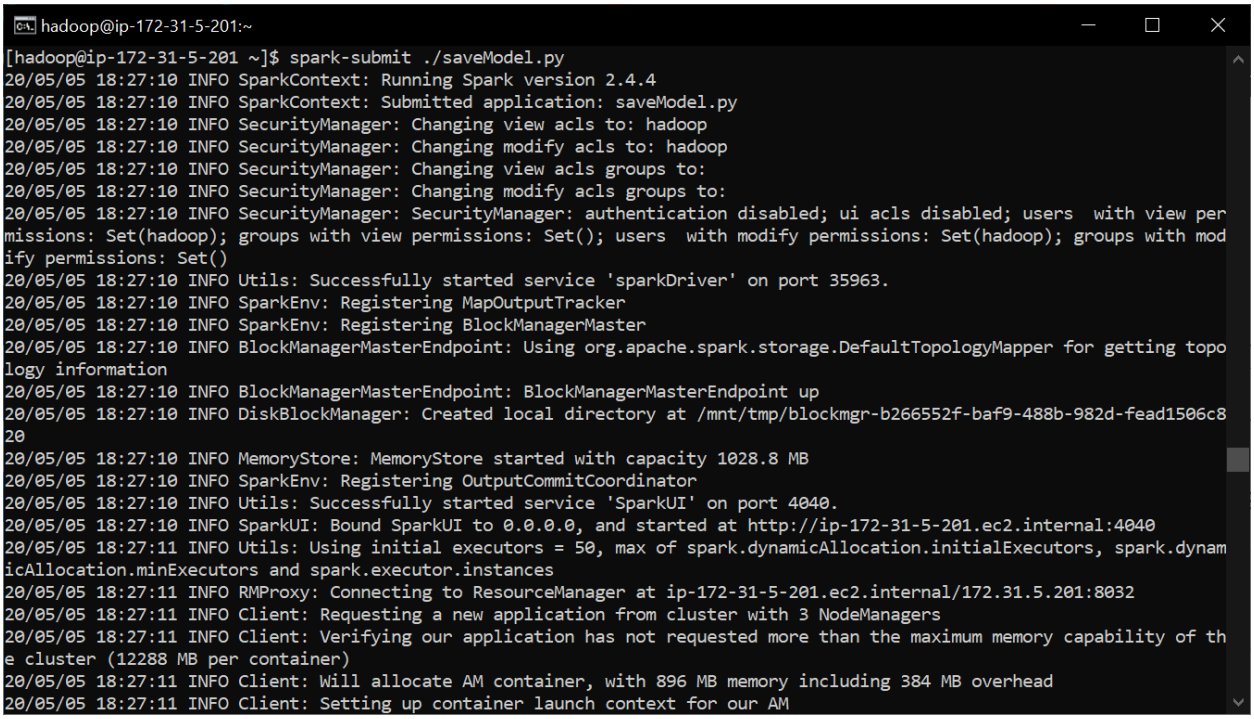


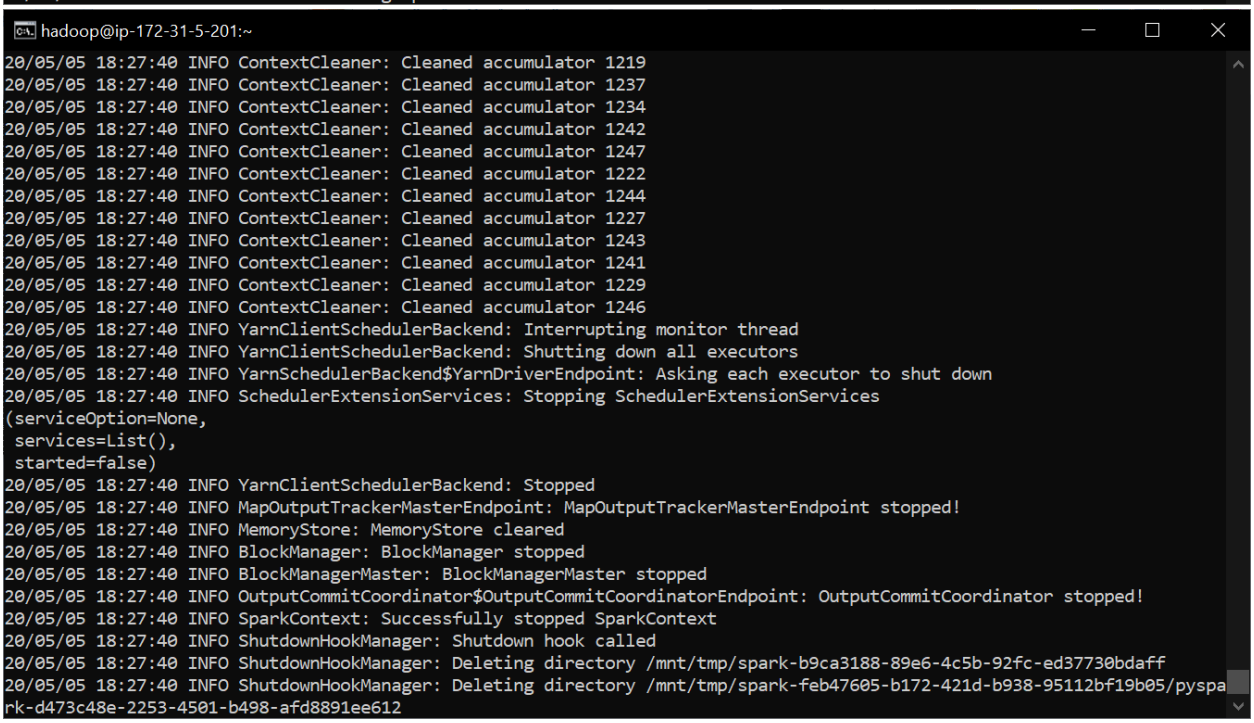
- Save **saveModel.py** from s3 bucket named **himaniproject2** to aws cluster. And the run that with spark-submit command

aws s3 cp s3://himaniproject2/saveModel.py .

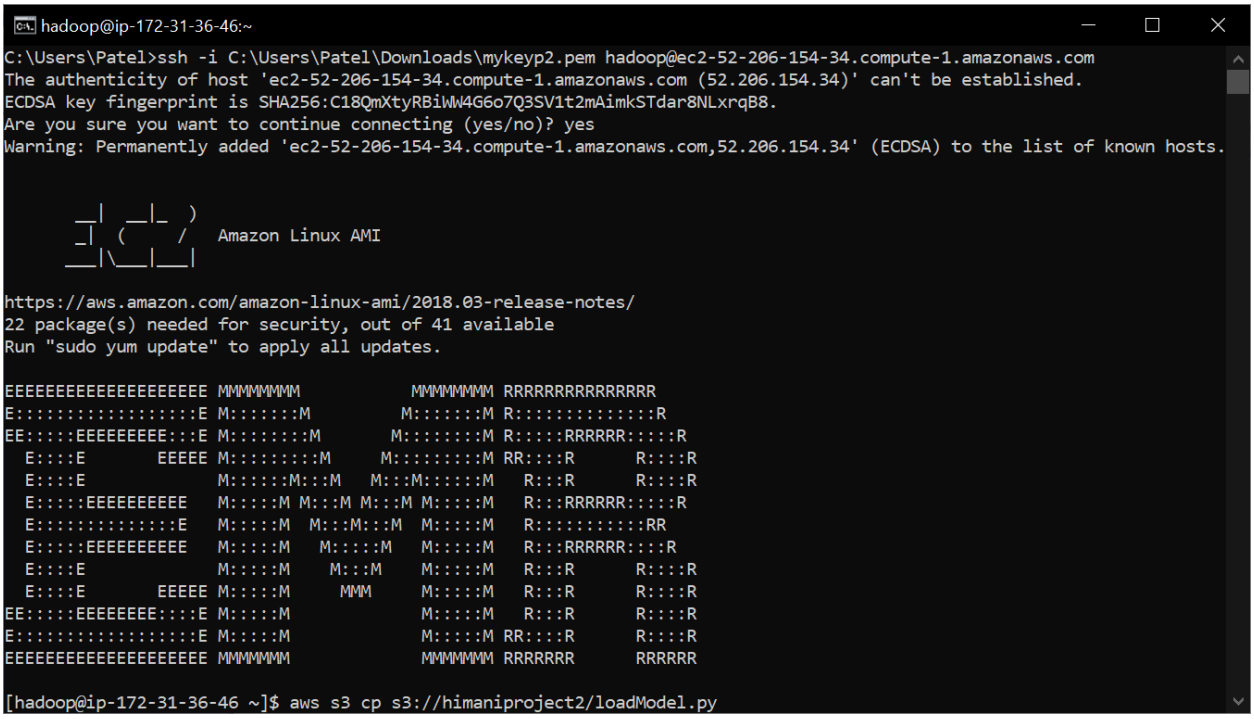
spark-submit ./saveModel.py

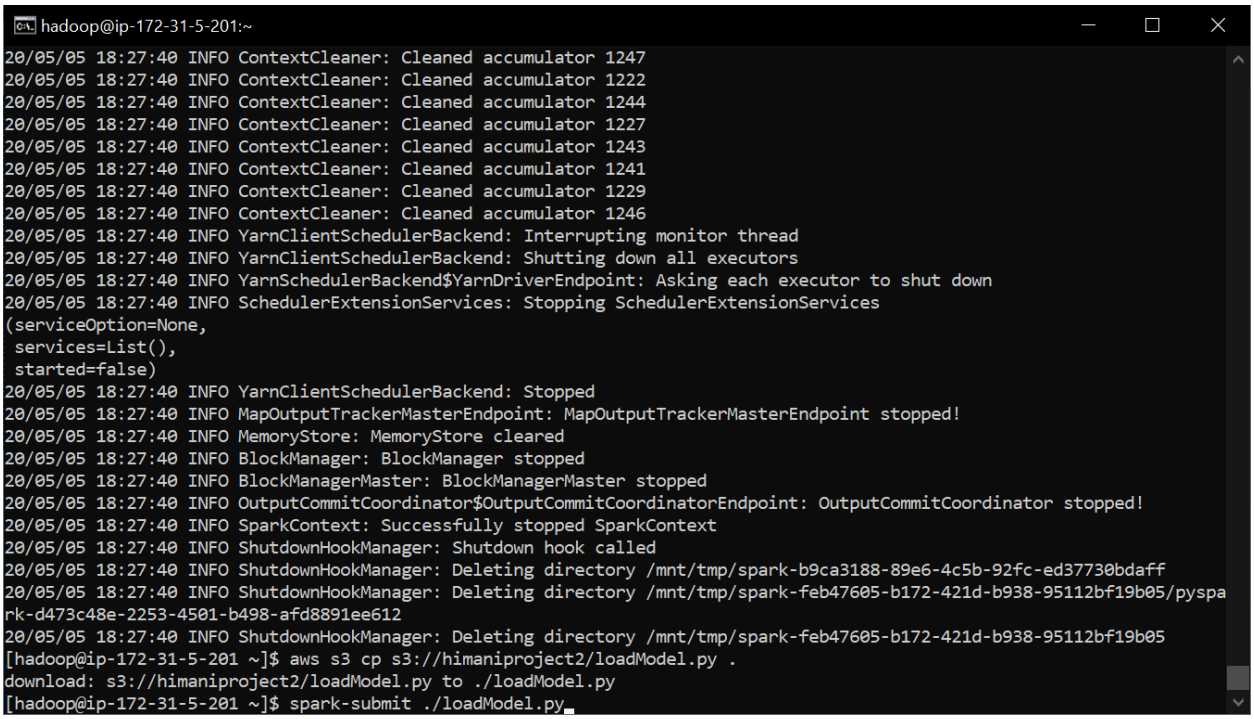


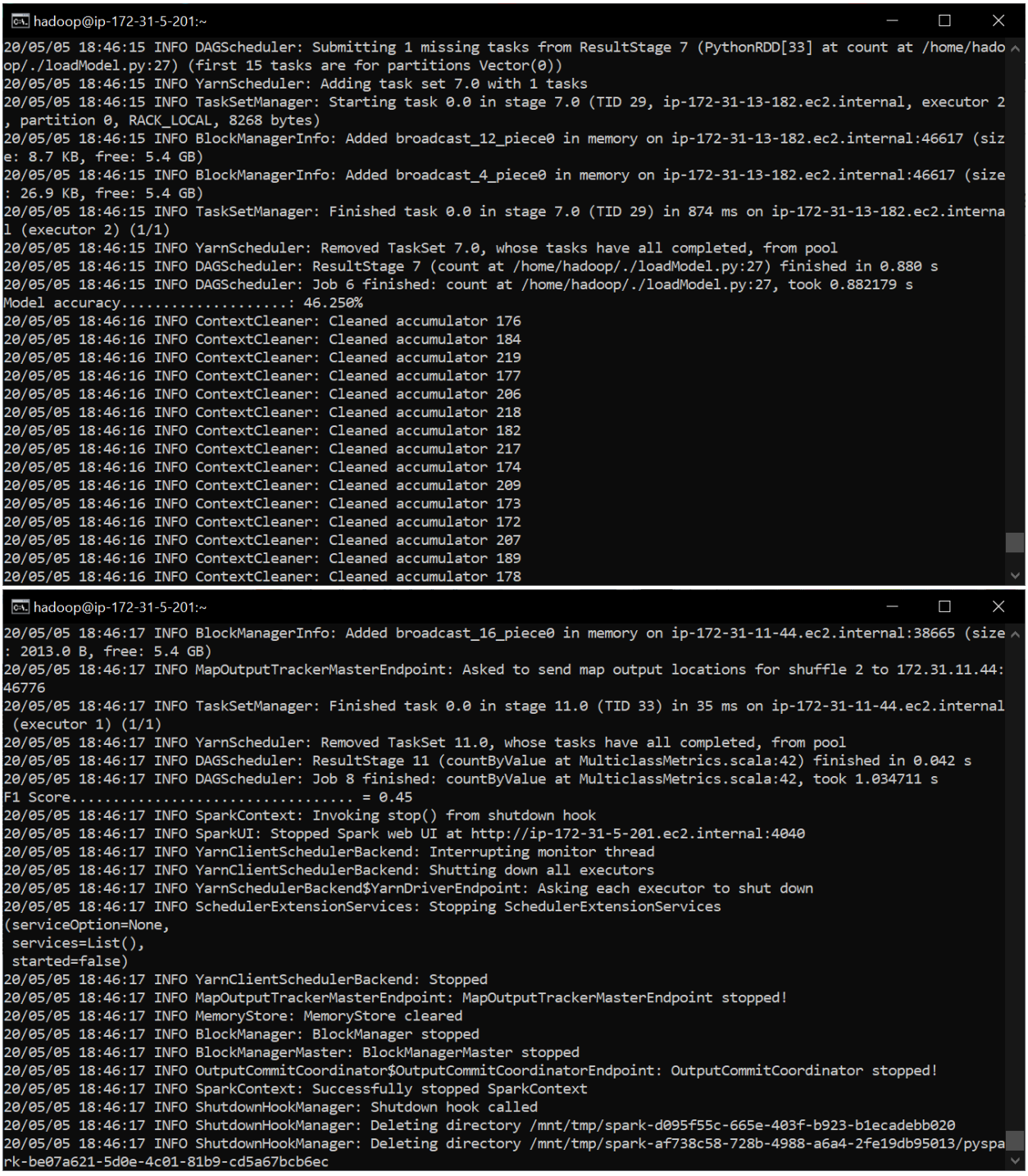

hadoop@ip-172-31-5-201:~
[hadoop@ip-172-31-5-201 ~]$ spark-submit ./saveModel.py
20/05/05 18:27:10 INFO SparkContext: Running Spark version 2.4.4
20/05/05 18:27:10 INFO SparkContext: Submitted application: saveModel.py
20/05/05 18:27:10 INFO SecurityManager: Changing view acls to: hadoop
20/05/05 18:27:10 INFO SecurityManager: Changing modify acls to: hadoop
20/05/05 18:27:10 INFO SecurityManager: Changing view acls groups to:
20/05/05 18:27:10 INFO SecurityManager: Changing modify acls groups to:
20/05/05 18:27:10 INFO SecurityManager: SecurityManager: authentication disabled; ui acls disabled; users with view permissions: Set(hadoop); groups with view permissions: Set(); users with modify permissions: Set(hadoop); groups with modify permissions: Set()
20/05/05 18:27:10 INFO Utils: Successfully started service 'sparkDriver' on port 35963.
20/05/05 18:27:10 INFO SparkEnv: Registering MapOutputTracker
20/05/05 18:27:10 INFO SparkEnv: Registering BlockManagerMaster
20/05/05 18:27:10 INFO BlockManagerMasterEndpoint: Using org.apache.spark.storage.DefaultTopologyMapper for getting topology information
20/05/05 18:27:10 INFO BlockManagerMasterEndpoint: BlockManagerMasterEndpoint up
20/05/05 18:27:10 INFO DiskBlockManager: Created local directory at /mnt/tmp/blockmgr-b266552f-baf9-488b-982d-fead1506c820
20/05/05 18:27:10 INFO MemoryStore: MemoryStore started with capacity 1028.8 MB
20/05/05 18:27:10 INFO SparkEnv: Registering OutputCommitCoordinator
20/05/05 18:27:10 INFO Utils: Successfully started service 'SparkUI' on port 4040.
20/05/05 18:27:10 INFO SparkUI: Bound SparkUI to 0.0.0.0, and started at http://ip-172-31-5-201.ec2.internal:4040
20/05/05 18:27:11 INFO Utils: Using initial executors = 50, max of spark.dynamicAllocation.initialExecutors, spark.dynamicAllocation.minExecutors and spark.executor.instances
20/05/05 18:27:11 INFO RMProxy: Connecting to ResourceManager at ip-172-31-5-201.ec2.internal/172.31.5.201:8032
20/05/05 18:27:11 INFO Client: Requesting a new application from cluster with 3 NodeManagers
20/05/05 18:27:11 INFO Client: Verifying our application has not requested more than the maximum memory capability of the cluster (12288 MB per container)
20/05/05 18:27:11 INFO Client: Will allocate AM container, with 896 MB memory including 384 MB overhead
20/05/05 18:27:11 INFO Client: Setting up container launch context for our AM


hadoop@ip-172-31-5-201:~
20/05/05 18:27:40 INFO ContextCleaner: Cleaned accumulator 1219
20/05/05 18:27:40 INFO ContextCleaner: Cleaned accumulator 1237
20/05/05 18:27:40 INFO ContextCleaner: Cleaned accumulator 1234
20/05/05 18:27:40 INFO ContextCleaner: Cleaned accumulator 1242
20/05/05 18:27:40 INFO ContextCleaner: Cleaned accumulator 1247
20/05/05 18:27:40 INFO ContextCleaner: Cleaned accumulator 1222
20/05/05 18:27:40 INFO ContextCleaner: Cleaned accumulator 1244
20/05/05 18:27:40 INFO ContextCleaner: Cleaned accumulator 1227
20/05/05 18:27:40 INFO ContextCleaner: Cleaned accumulator 1243
20/05/05 18:27:40 INFO ContextCleaner: Cleaned accumulator 1241
20/05/05 18:27:40 INFO ContextCleaner: Cleaned accumulator 1229
20/05/05 18:27:40 INFO ContextCleaner: Cleaned accumulator 1246
20/05/05 18:27:40 INFO YarnClientSchedulerBackend: Interrupting monitor thread
20/05/05 18:27:40 INFO YarnClientSchedulerBackend: Shutting down all executors
20/05/05 18:27:40 INFO YarnSchedulerBackend$YarnDriverEndpoint: Asking each executor to shut down
20/05/05 18:27:40 INFO SchedulerExtensionServices: Stopping SchedulerExtensionServices
(serviceOption=None,
services=List(),
started=false)
20/05/05 18:27:40 INFO YarnClientSchedulerBackend: Stopped
20/05/05 18:27:40 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
20/05/05 18:27:40 INFO MemoryStore: MemoryStore cleared
20/05/05 18:27:40 INFO BlockManager: BlockManager stopped
20/05/05 18:27:40 INFO BlockManagerMaster: BlockManagerMaster stopped
20/05/05 18:27:40 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stopped!
20/05/05 18:27:40 INFO SparkContext: Successfully stopped SparkContext
20/05/05 18:27:40 INFO ShutdownHookManager: Shutdown hook called
20/05/05 18:27:40 INFO ShutdownHookManager: Deleting directory /mnt/tmp/spark-b9ca3188-89e6-4c5b-92fc-ed37730bdaff
20/05/05 18:27:40 INFO ShutdownHookManager: Deleting directory /mnt/tmp/spark-feb47605-b172-421d-b938-95112bf19b05/pyspark-d473c48e-2253-4501-b498-afd8891ee612

- Create another cluster with only one instance and load prediction model.







Here decision tree model is giving 46.25% accuracy and 0.45 f1 score.