**Data Engineering Document**

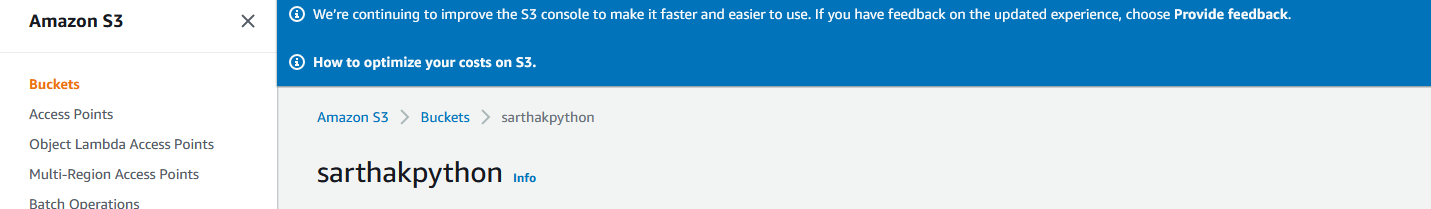
This document uses the below services:

1. AWS EMR
2. AWS Glue
3. AWS S3
4. Spark
5. Python
6. VScode
7. Github

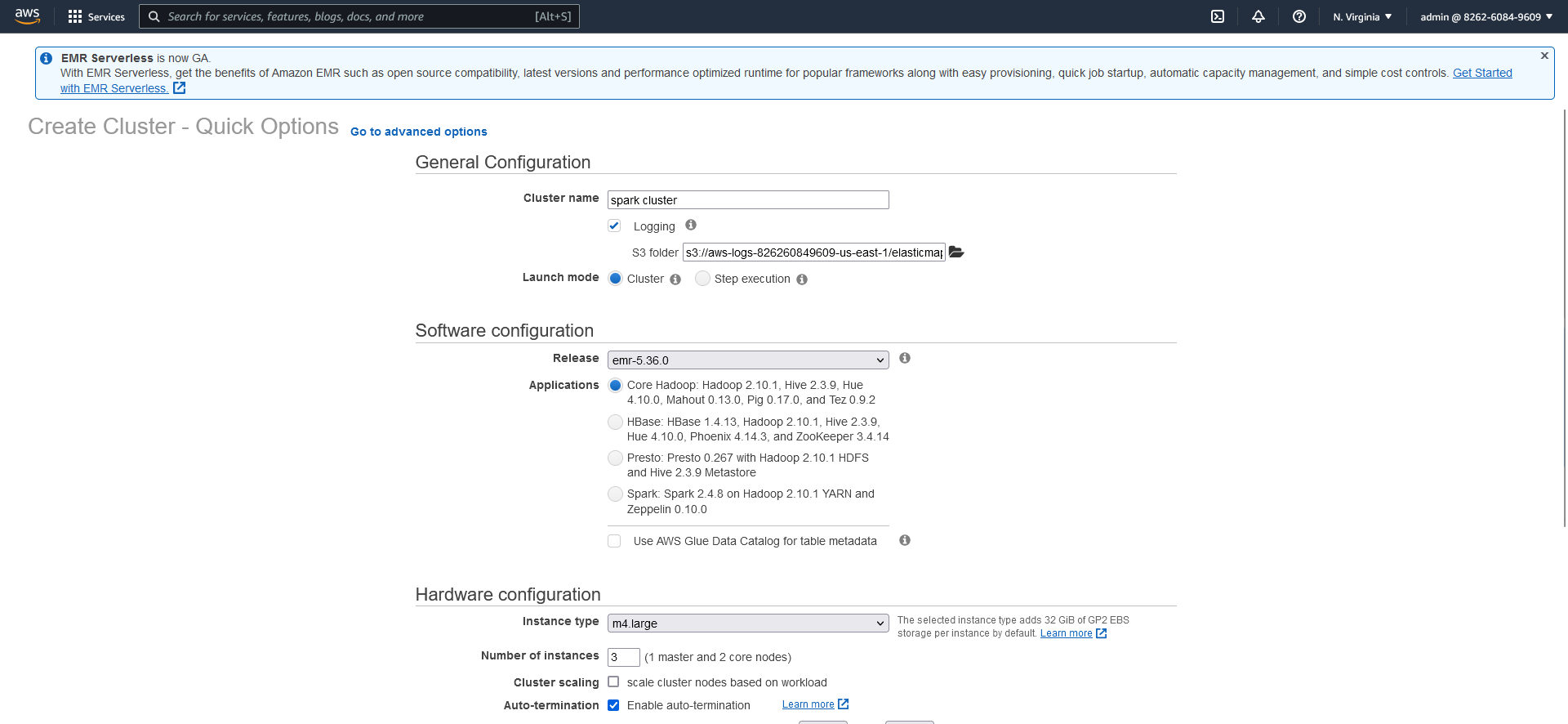
The reason I chose AWS EMR over AWS Glue is AWS EMR Spark has a good efficiency to handle complex tasks. Your requirements are complex that is why spark is the best tool to solve complex tasks due to its optimization.

Following are the steps to create and deploy the application in AWS.

Step 1 : Create S3 bucket.

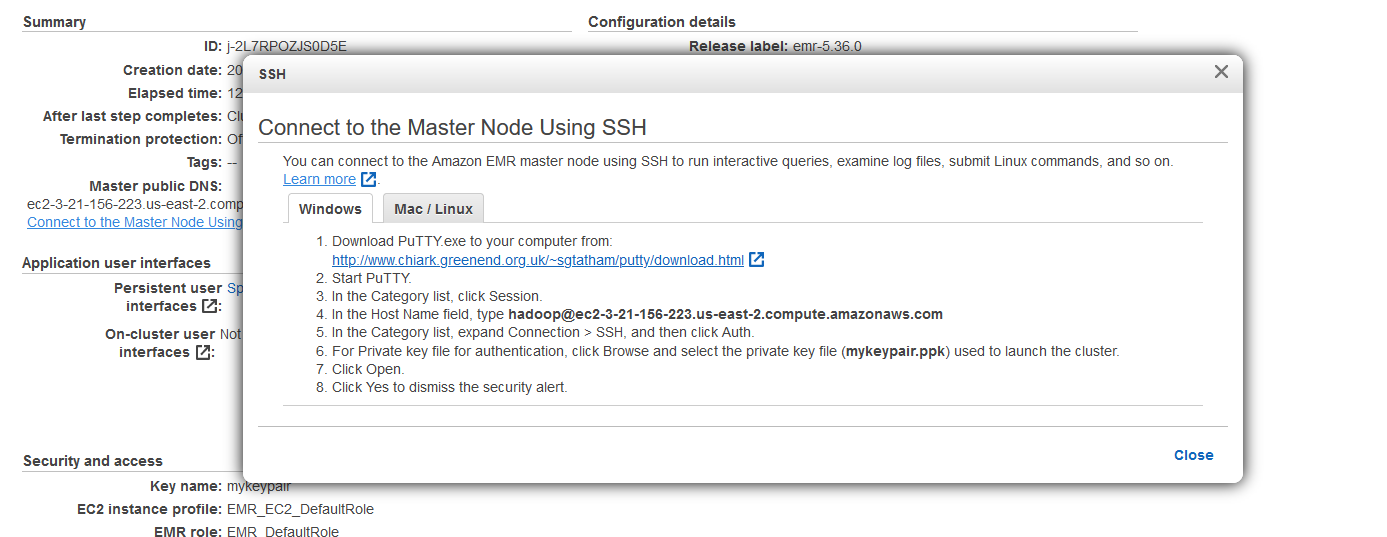


Step 2 : Create AWS EMR cluster “spark cluster” with the below configurations.



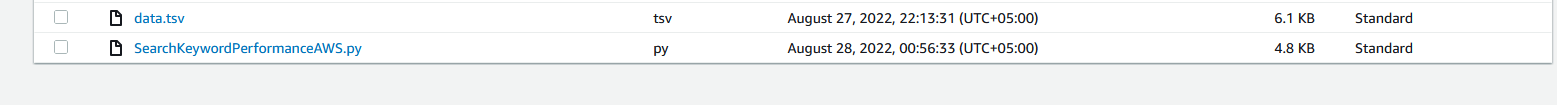
Step 3: Now wait for running status of cluster.

Step 4: Connect cluster with **PUTTY** application as per below steps in the screenshot.



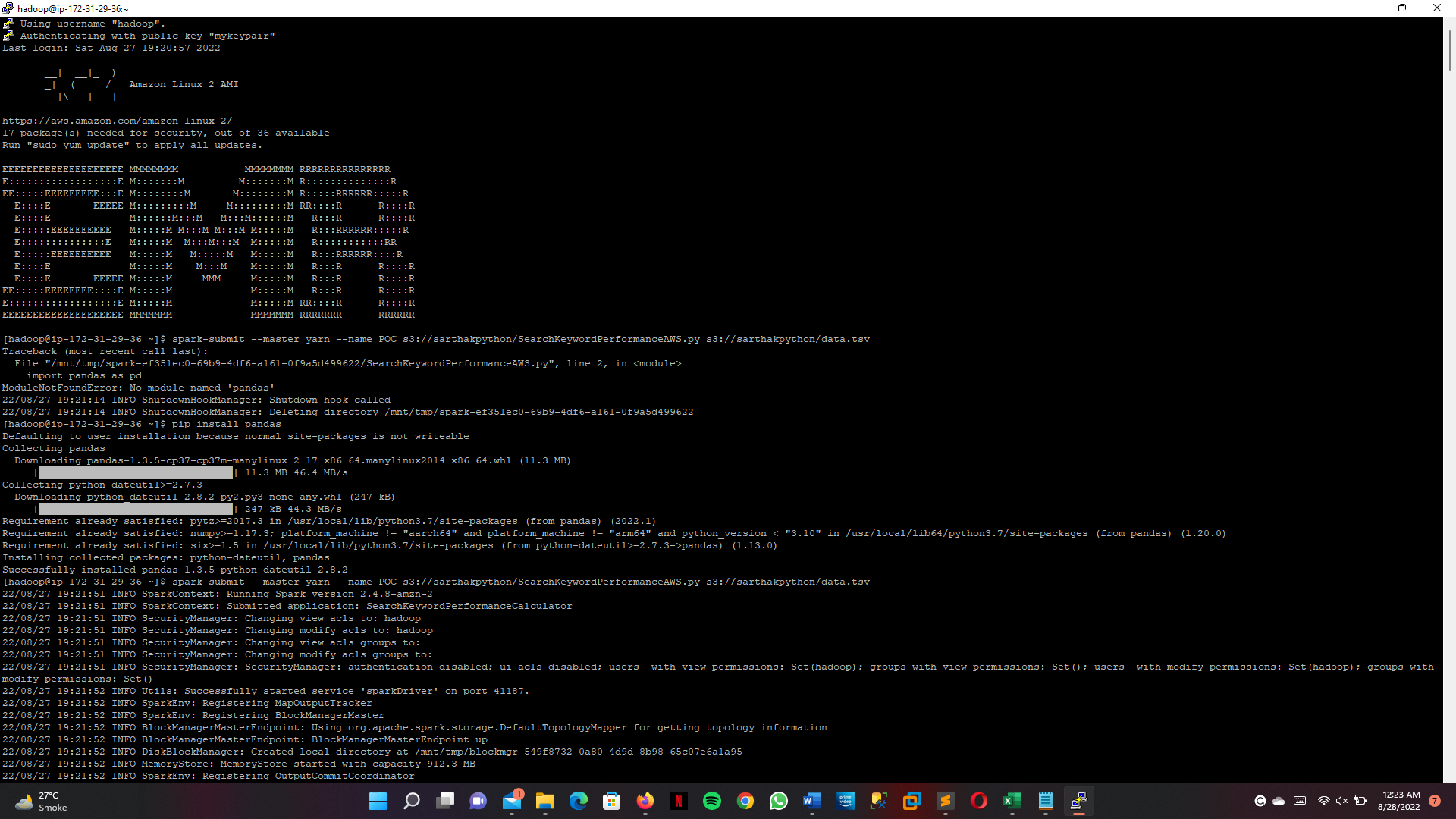
Step 5 : Create script in the notepad or sublime or visual studio code where you want.

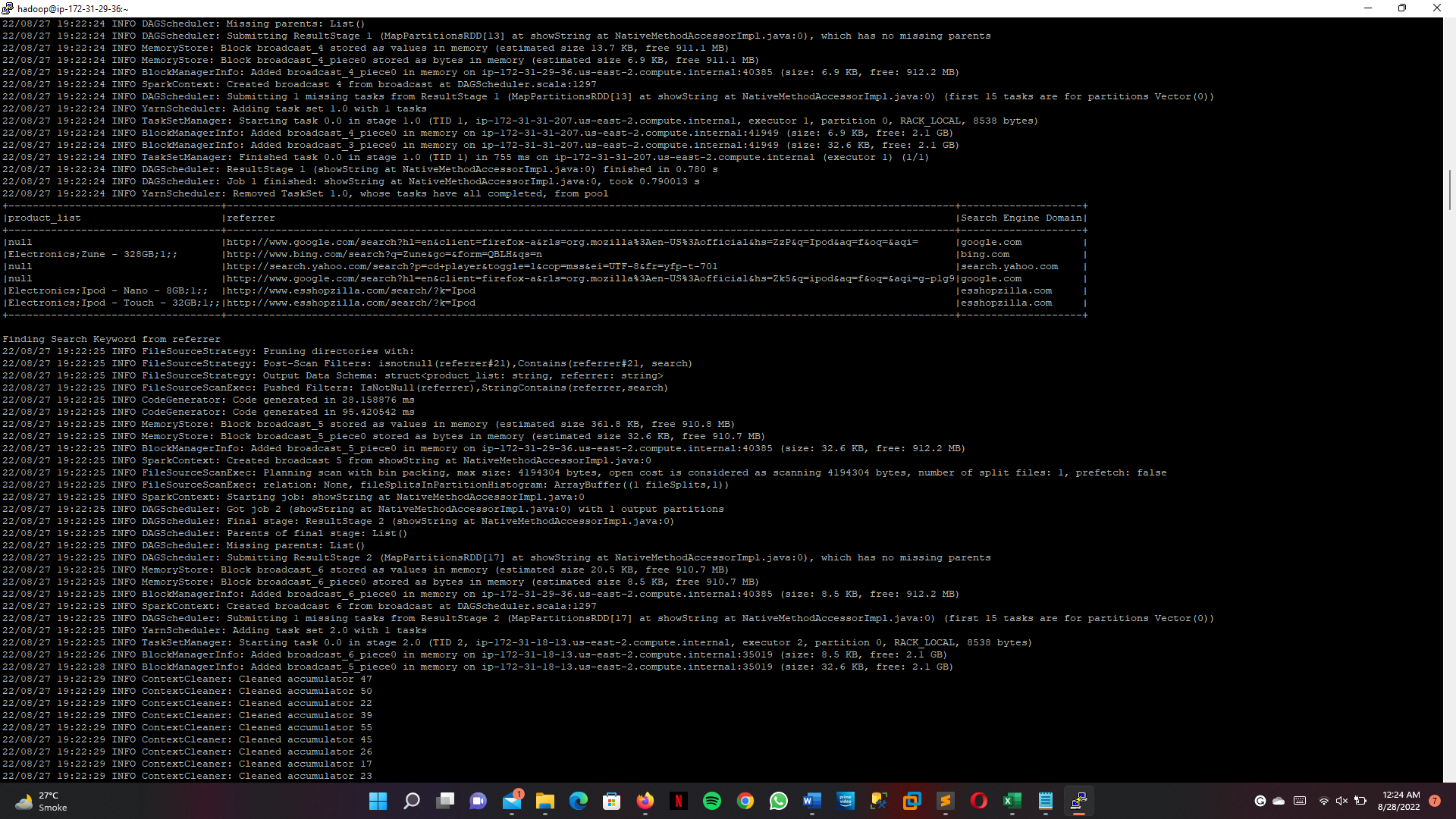
Step 6 : upload the script and data into AWS S3.

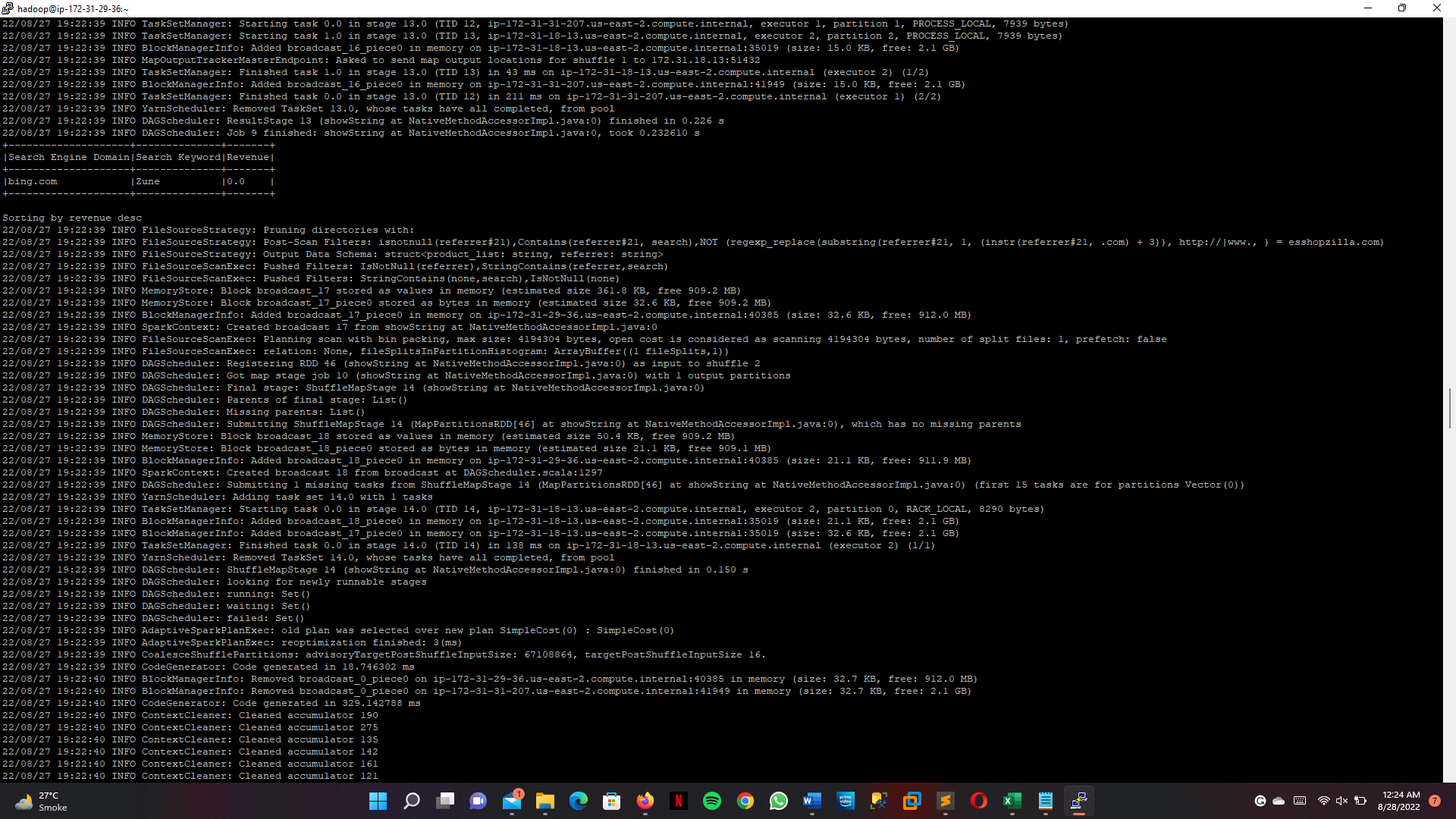


Step 6 : Run the Script in AWS cluster at the **PUTTY** application by using the following command.

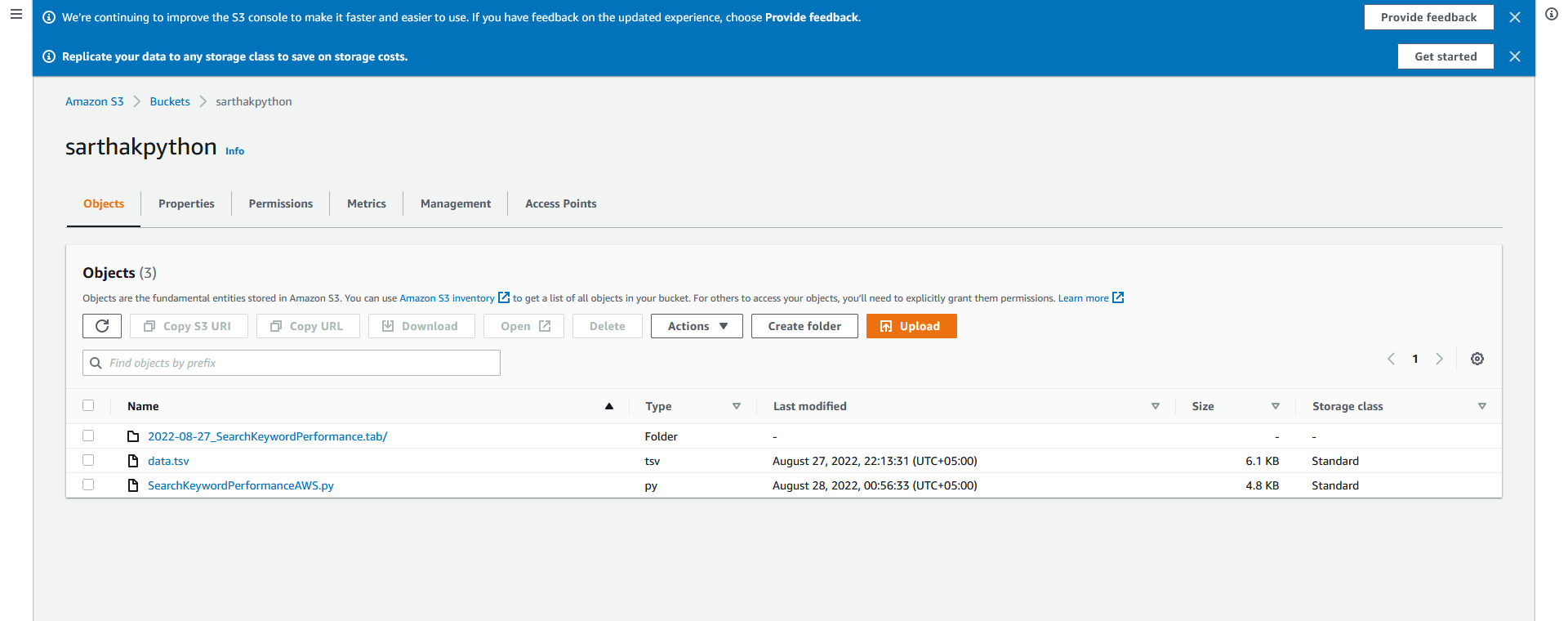
spark-submit --master yarn --name POC s3://sarthakpython/SearchKeywordPerformanceAWS.py s3://sarthakpython/data.tsv



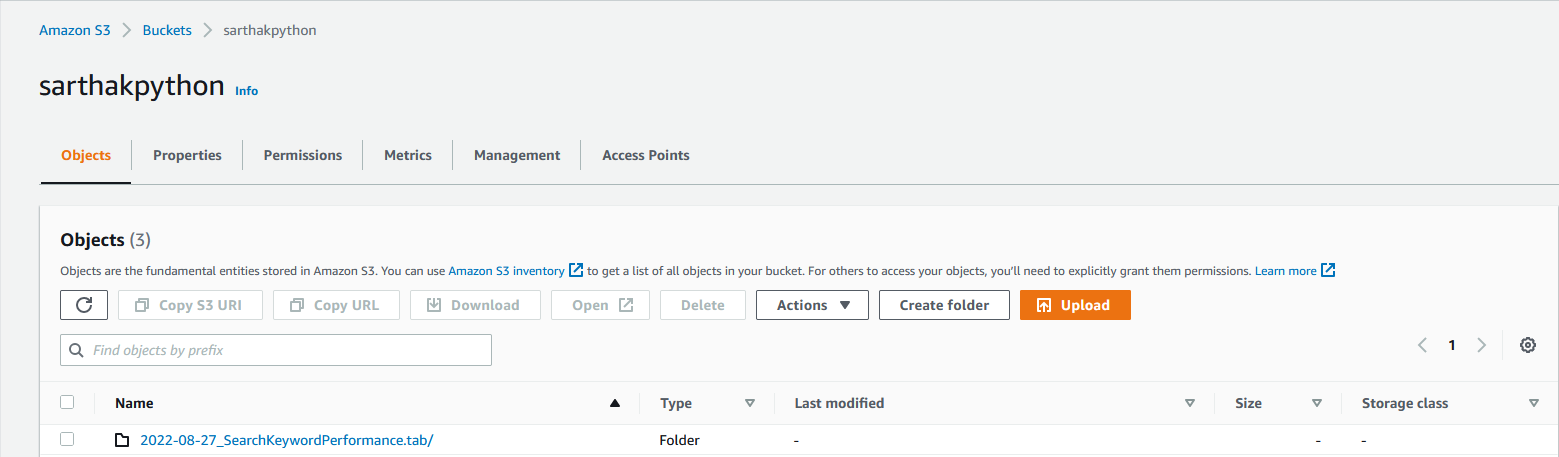




Step 7 : Terminate the AWS EMR cluster to save the cost.



Step 8 : Check the output file in the S3.



Considerations:

* The application performs the function expected.
* It can tolerate the exception handling or using the software in unexpected ways.
* Its performance is good enough for the required use case, under the expected load and data volume.
* Limited physical resources like memory, incorrect memory management, or an inefficient database engine can result into problem.
* Complicated database schema, bad indexing, and server limitations can be problem
* Lack of monitoring tools and too many external dependencies which prevent any unauthorized access