2025

Home Project 1

Implement multiple-input-files in Hadoop

DANIEL DADZIE APPIAH | 156801227 | DDAPPIAH

3RD MARCH, 2025.

Table of Contents

1 Introduction 2

2 0-Screen for Starting Project 2

3 Showing Data input files and Scripts in the folder. 3

4 Validating Folder Contents 5

5 Testing Mapper and Reducer Program 10

6 Executing the Python scripts via Hadoop Streaming JAR file for implementing multiple input files in Hadoop 14

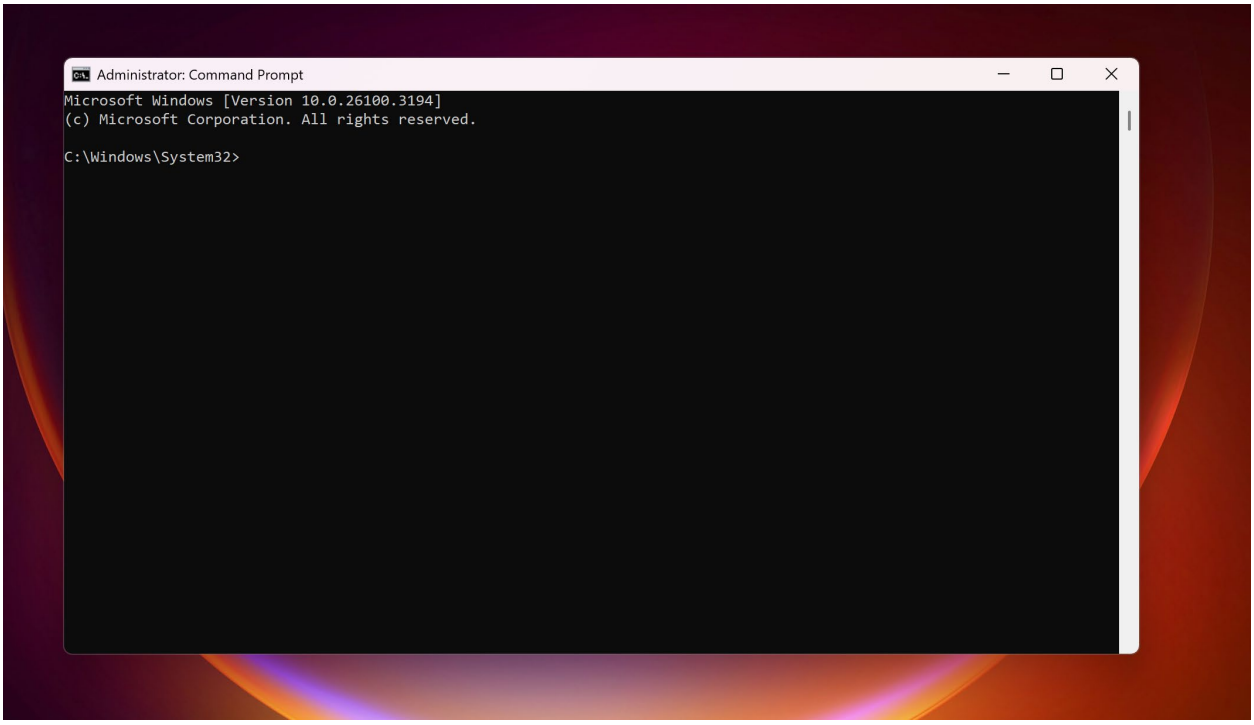
7 Analysis for streaming jar job for the multiple input file system 22

1 Introduction

We often have different file formats when processing multiple input files in Hadoop MapReduce. For example, text files and CSV files. Each file type requires its custom mapper, but all the data must be processed by a single reducer to aggregate and combine results.

2 0-Screen for Starting Project

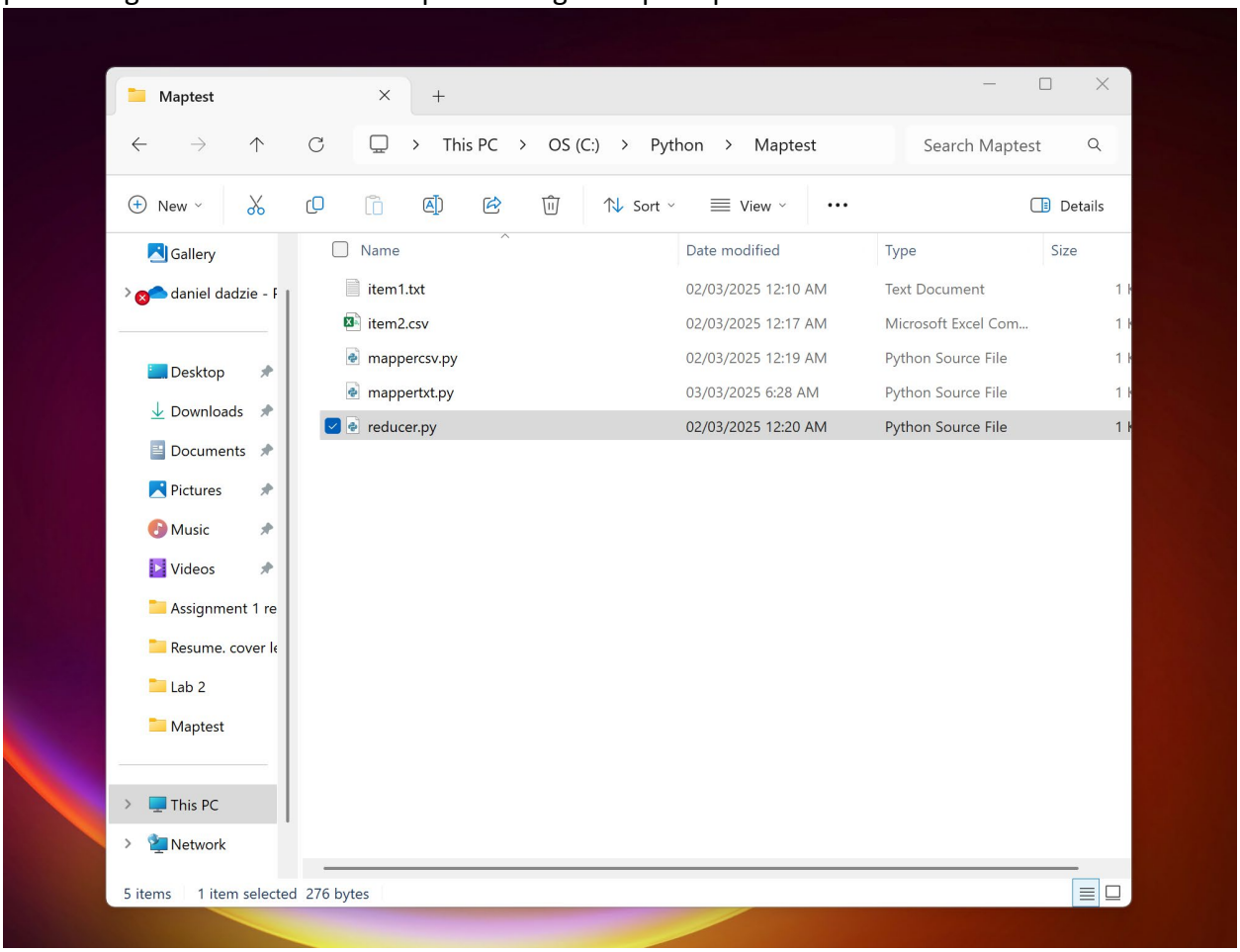
Starting up the command line as an administrator. Use of Windows Terminal.



3 Showing Data input files and Scripts in the folder.

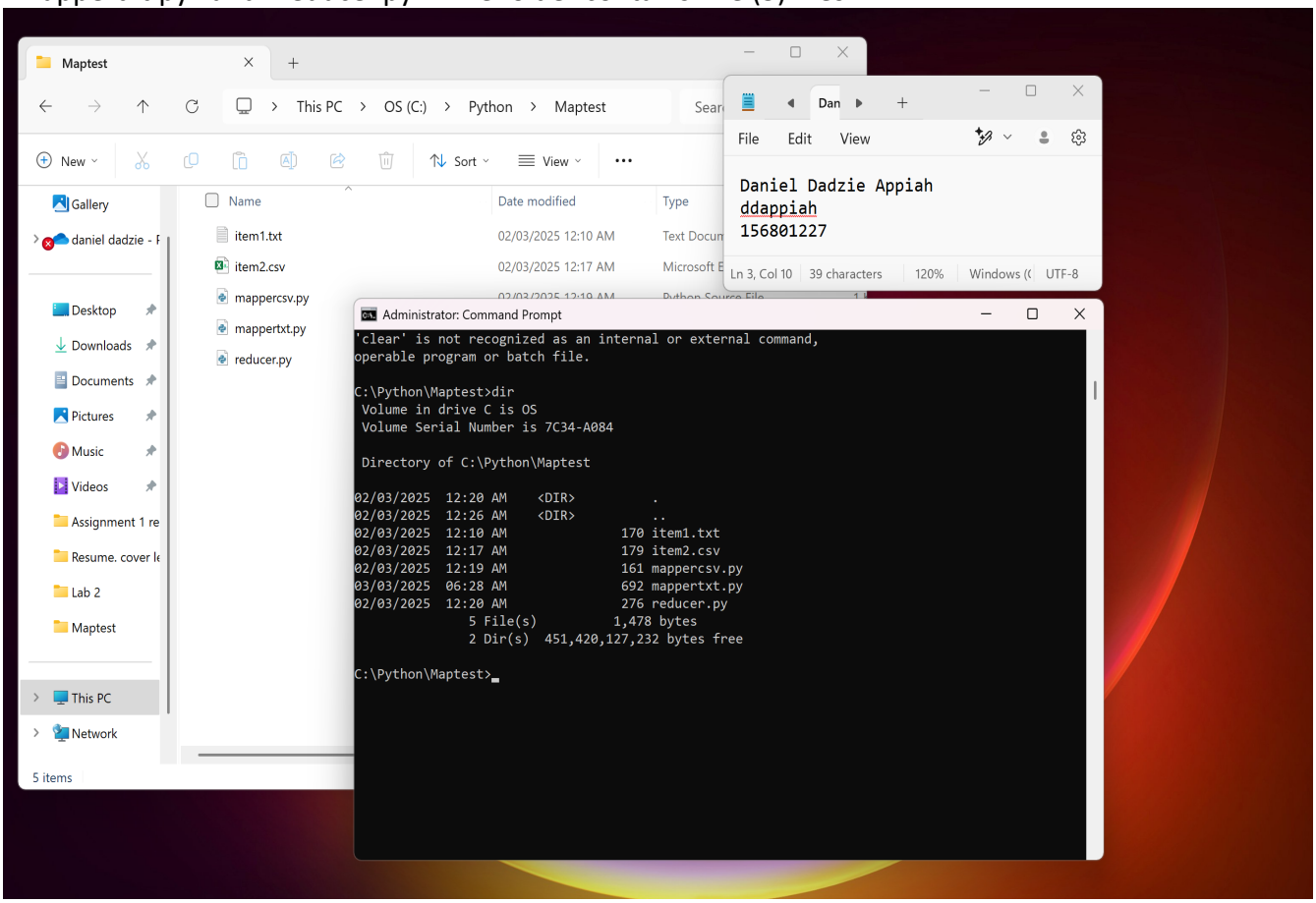
Create a Sample Input File.

Notepad files are created in a folder called “Python,” located in drive C. There are two input files with different formats. They are “item1.txt” and “item2.csv”, which will be used for future processing demonstrations of implementing multiple input files.



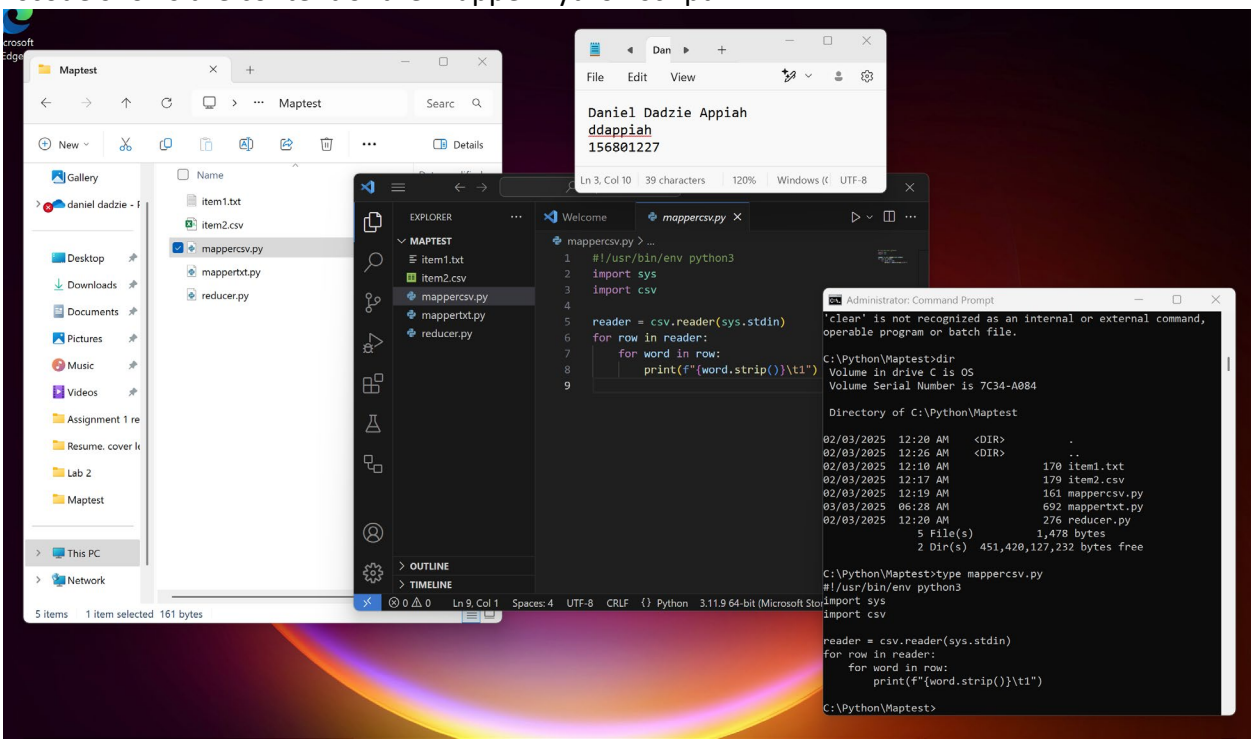
Folder Content

Content in the folder “Maptest” contains “item1.txt,” “item2.csv,” “mappercsv.py,” “mappertxt.py” and “reducer.py”. The folder contains five (5) files.

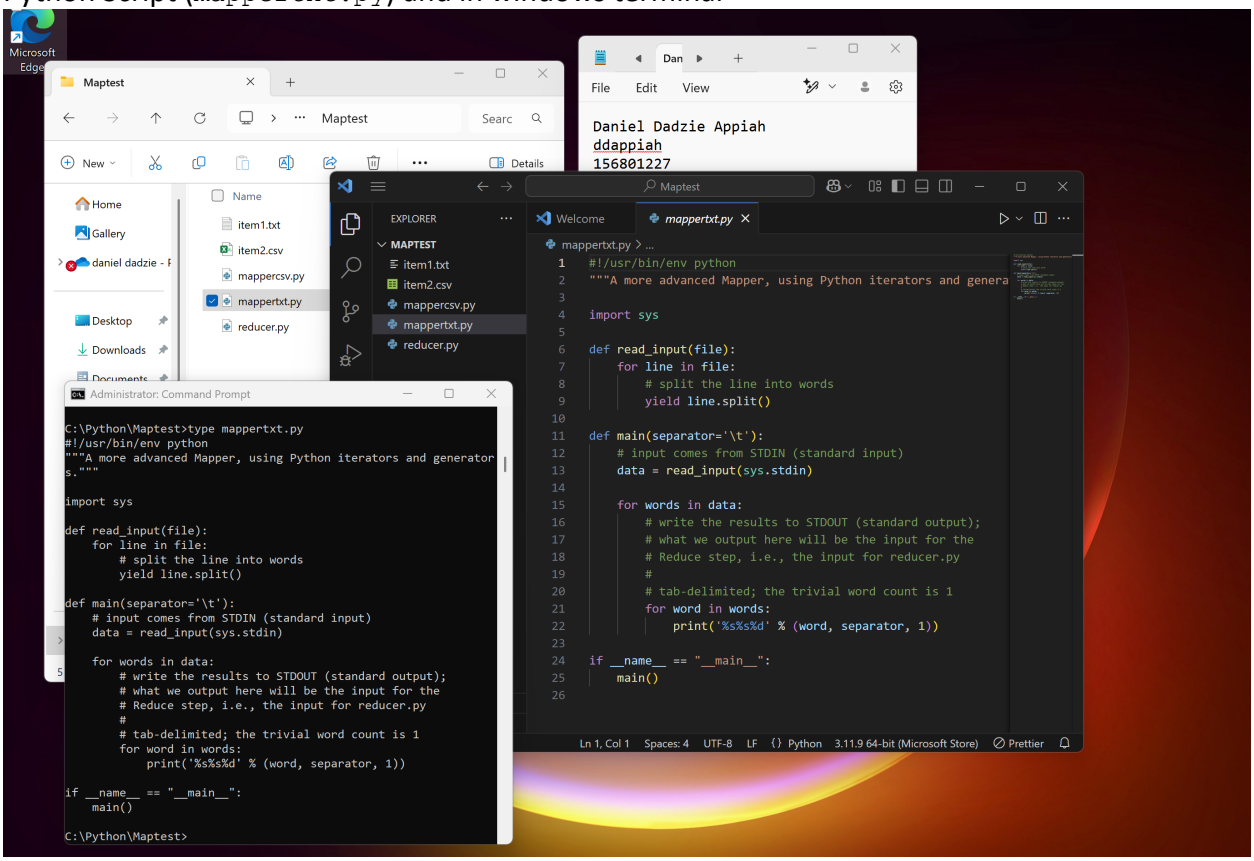


4 Validating Folder Contents

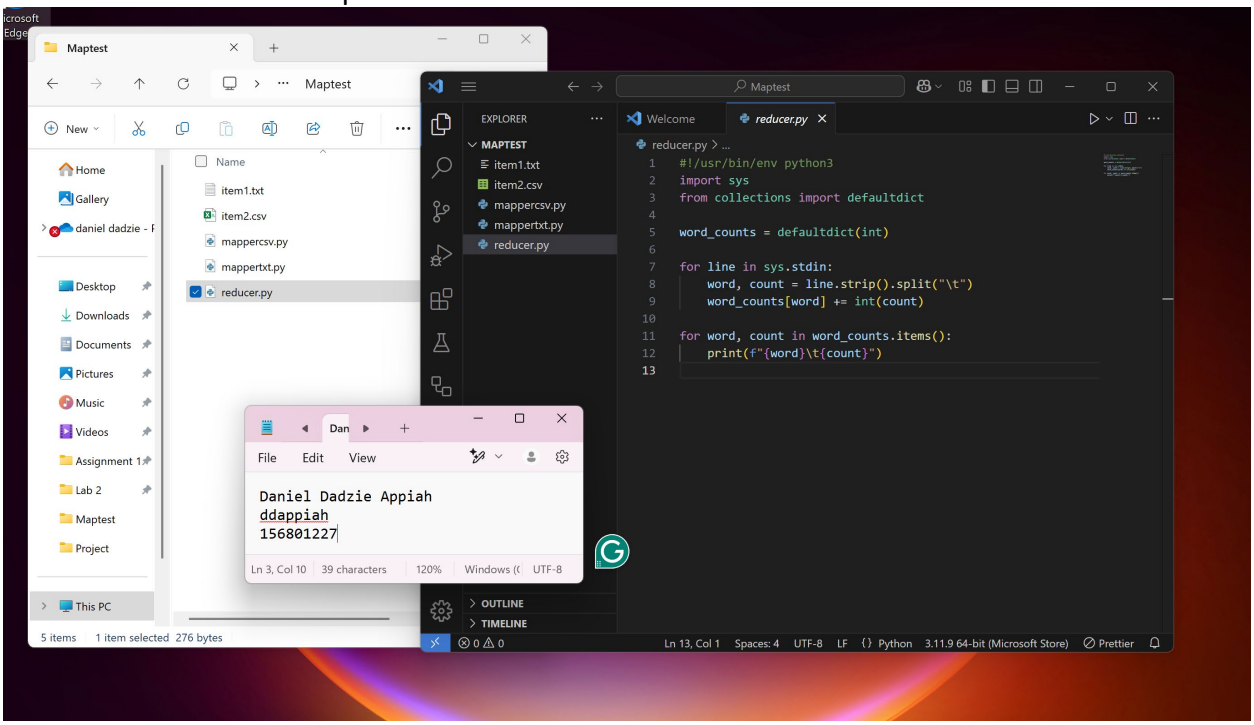
Details of created files in the "**Maptest**" folder**.** That is Mapper Python Script (`mappercsv.py`) and in windows terminal. The image shows the folder containing the mapper python file. The vscode shows the content of the mapper Python script.



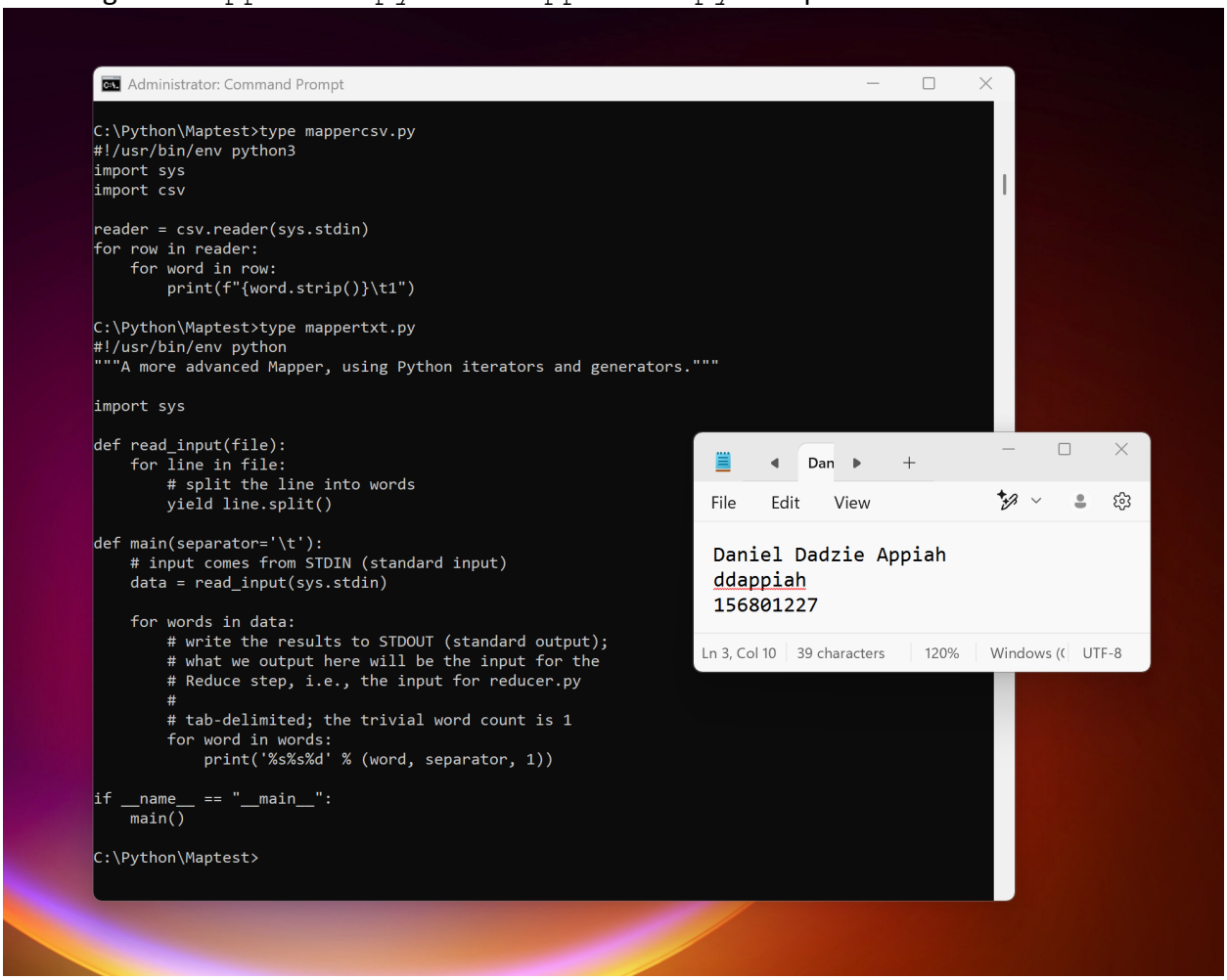
Python Script (`mappertxt.py`) and in windows terminal



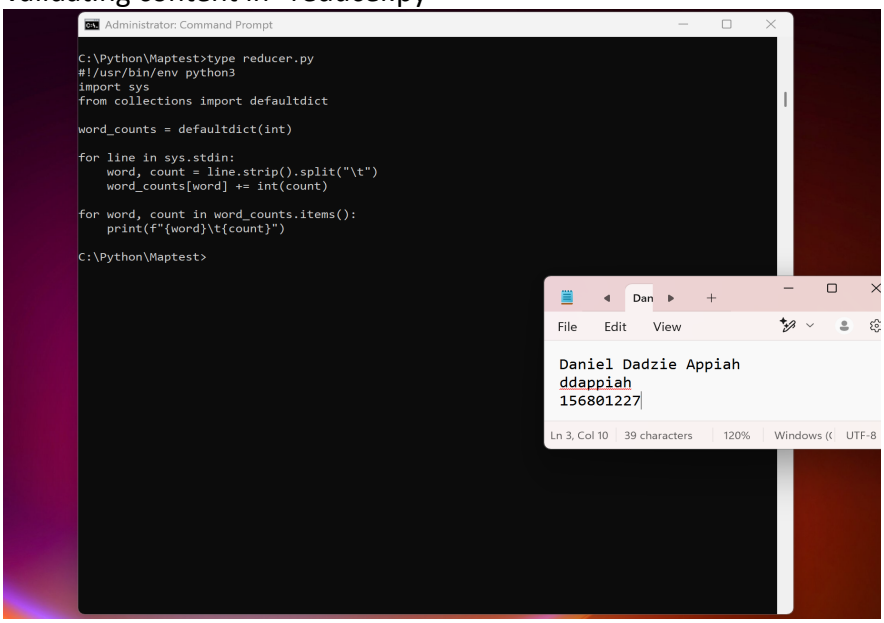
Writing the Reducer Python Script (`reducer.py`) and placing it in the "Scripts" folder. The script was made using Visual Studio code (vs code). The folder contains the reducer Python file and the vs code shows the script content



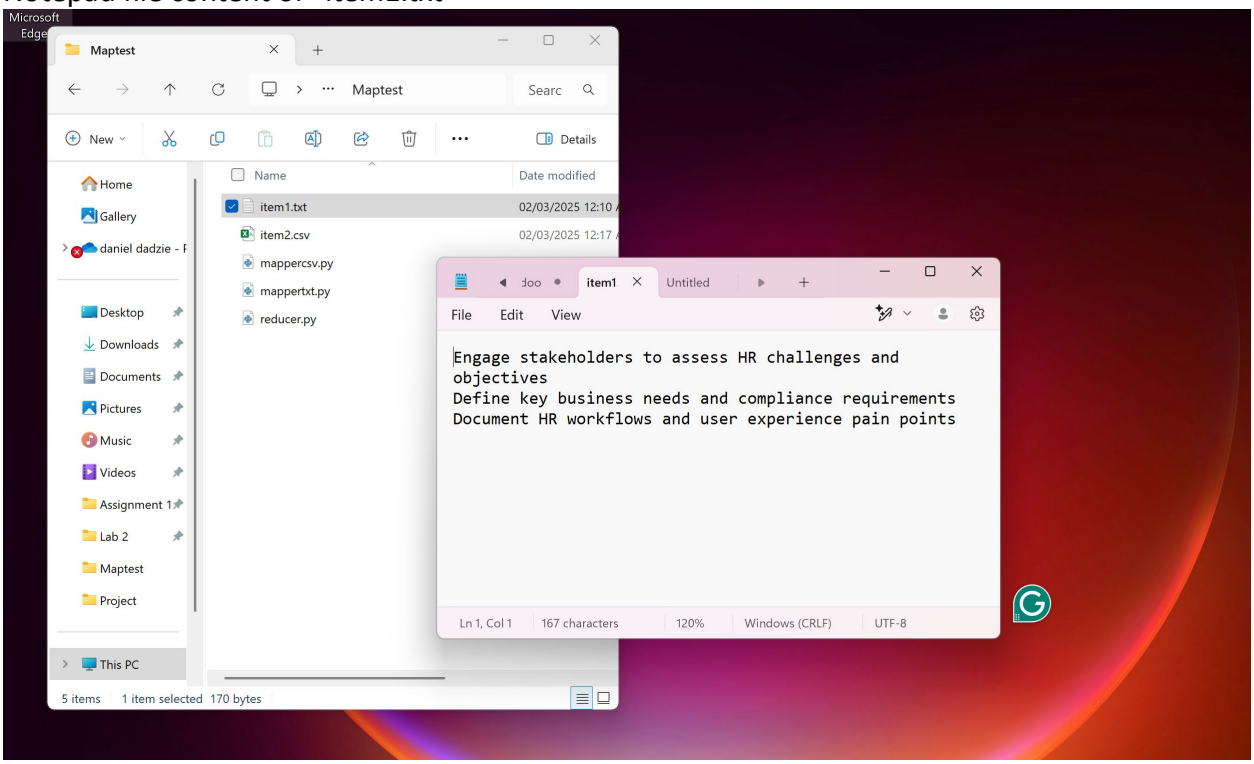
Checking the "`mappercsv.py`" and "`mappertxt.py`" Script from the windows terminal.



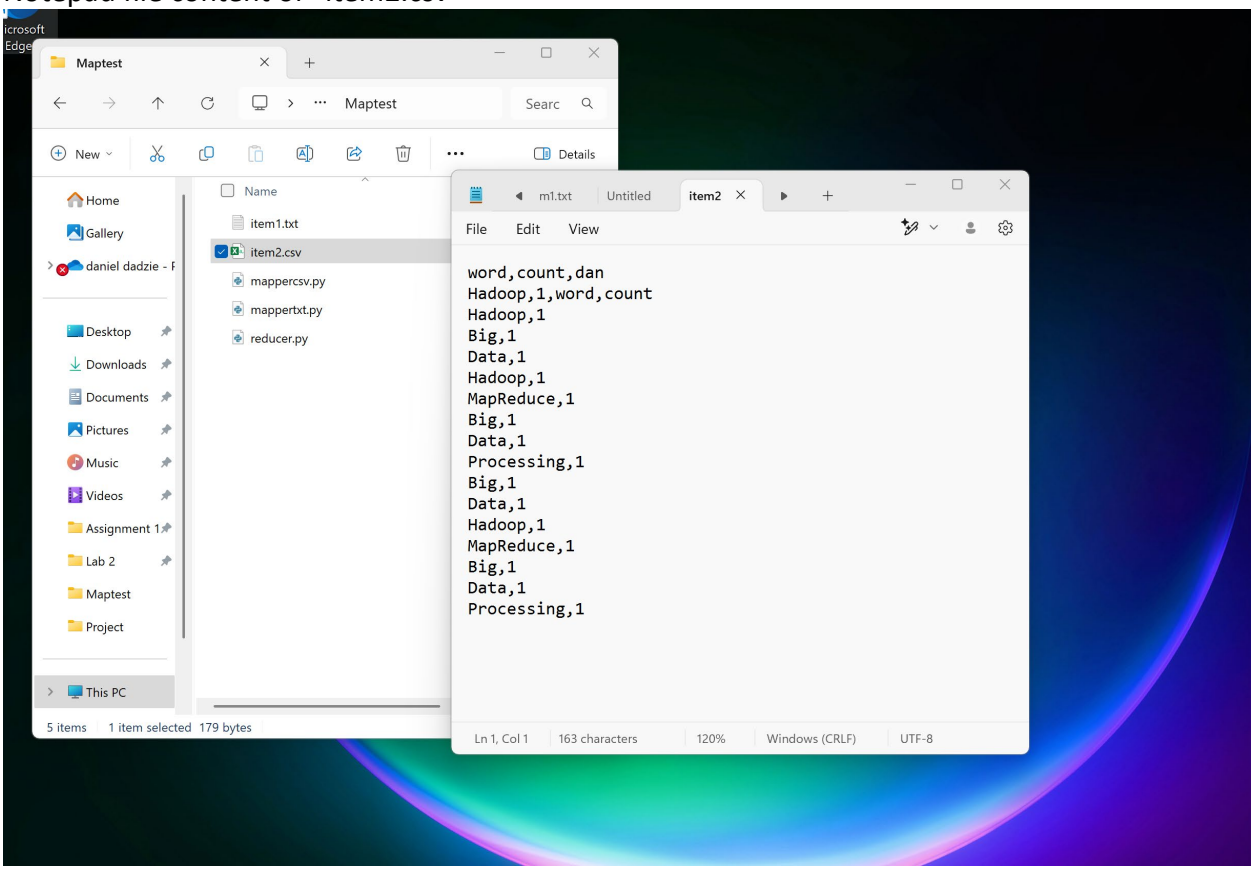
Validating content in "reducer.py"



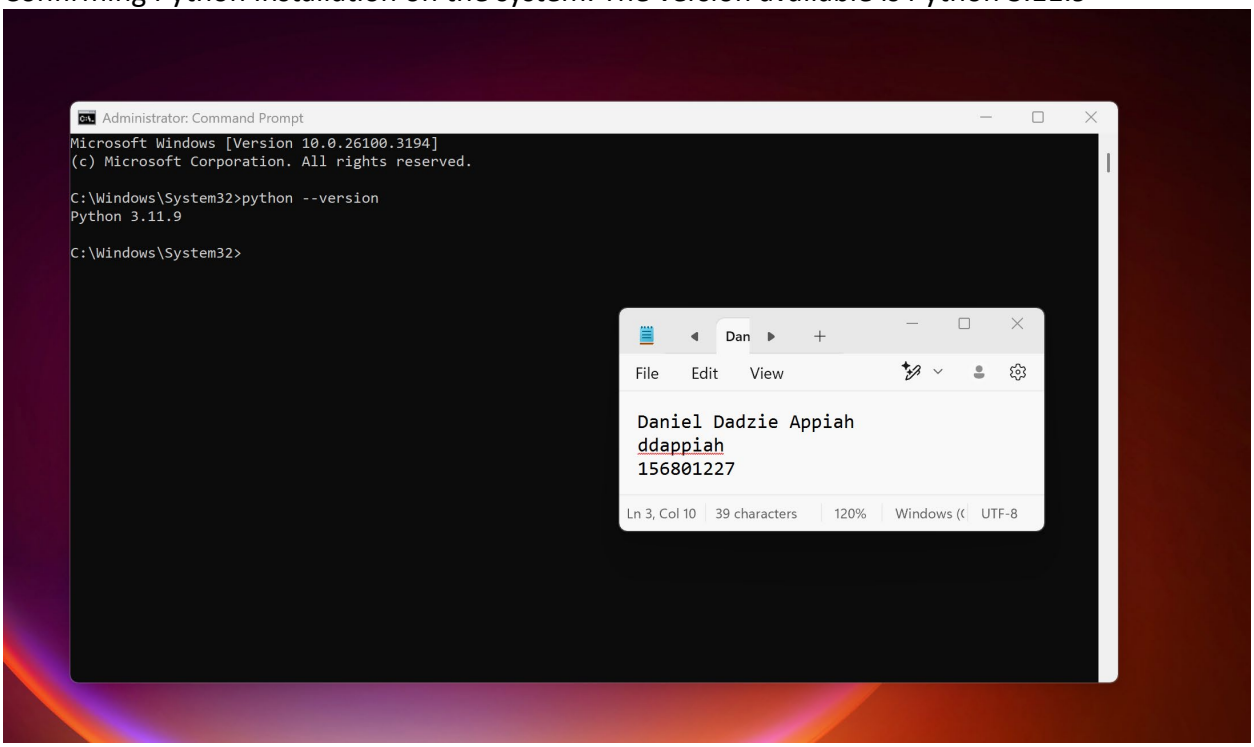
Notepad file content of “item1.txt”



Notepad file content of “item2.csv”



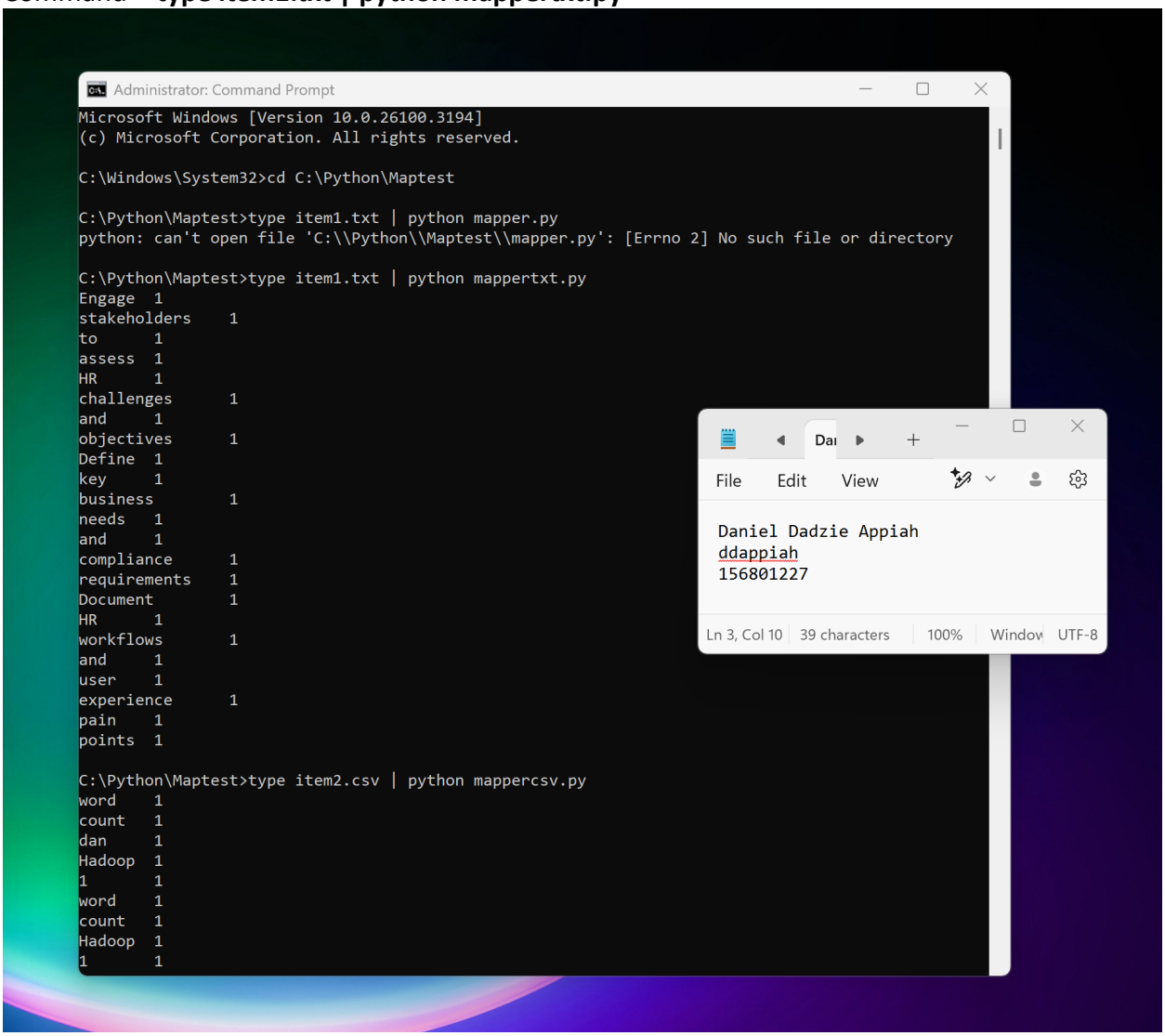
Confirming Python Installation on the system. The version available is Python 3.11.9



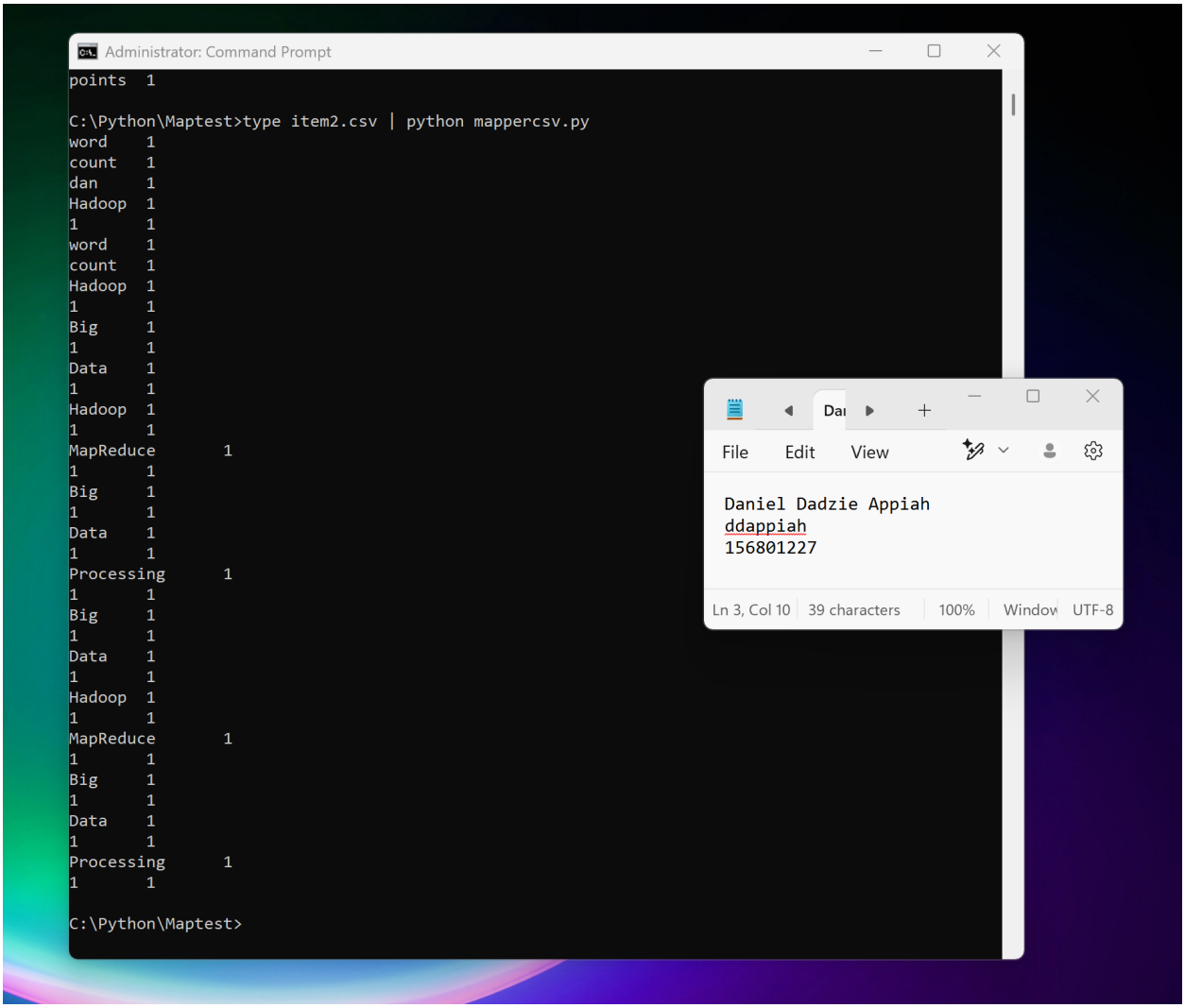
5 Testing Mapper and Reducer Program

Performing Mapper on the Content of the “item1.txt”.

Command- **type item1.txt | python mappertxt.py**

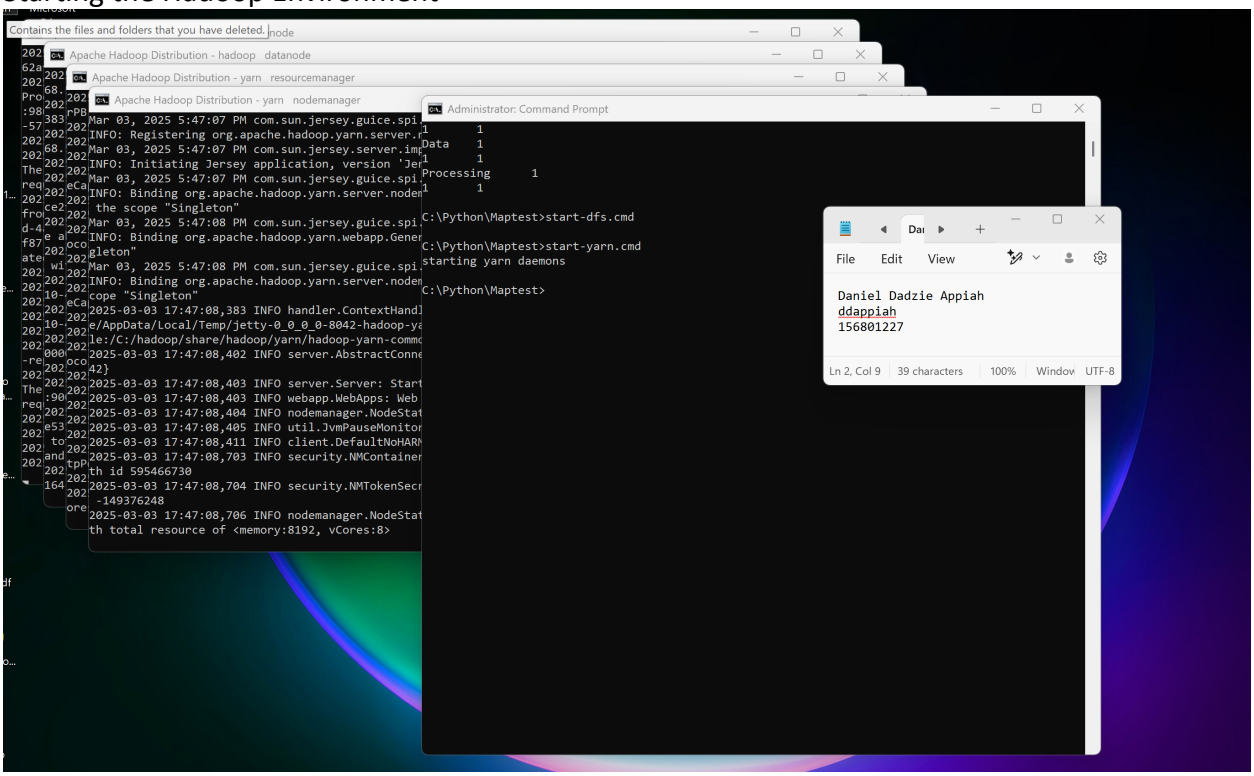


Mapper on the Content of the “item2.csv”
Command- **type item2.csv | python mappercsv.py**

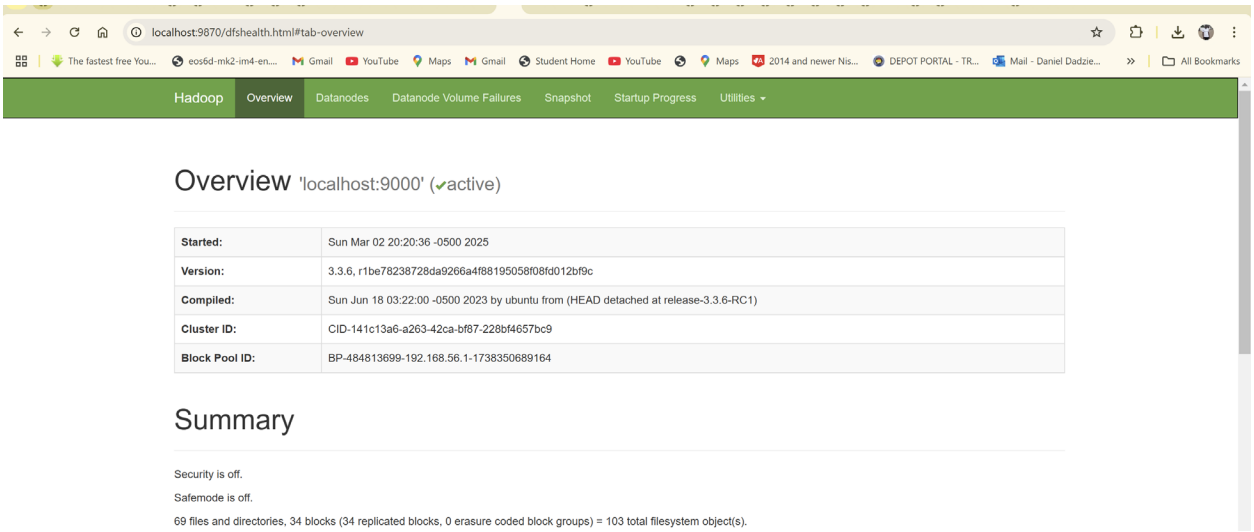


Reducer on the Content for item1 . `txt`” and “item2.csv”

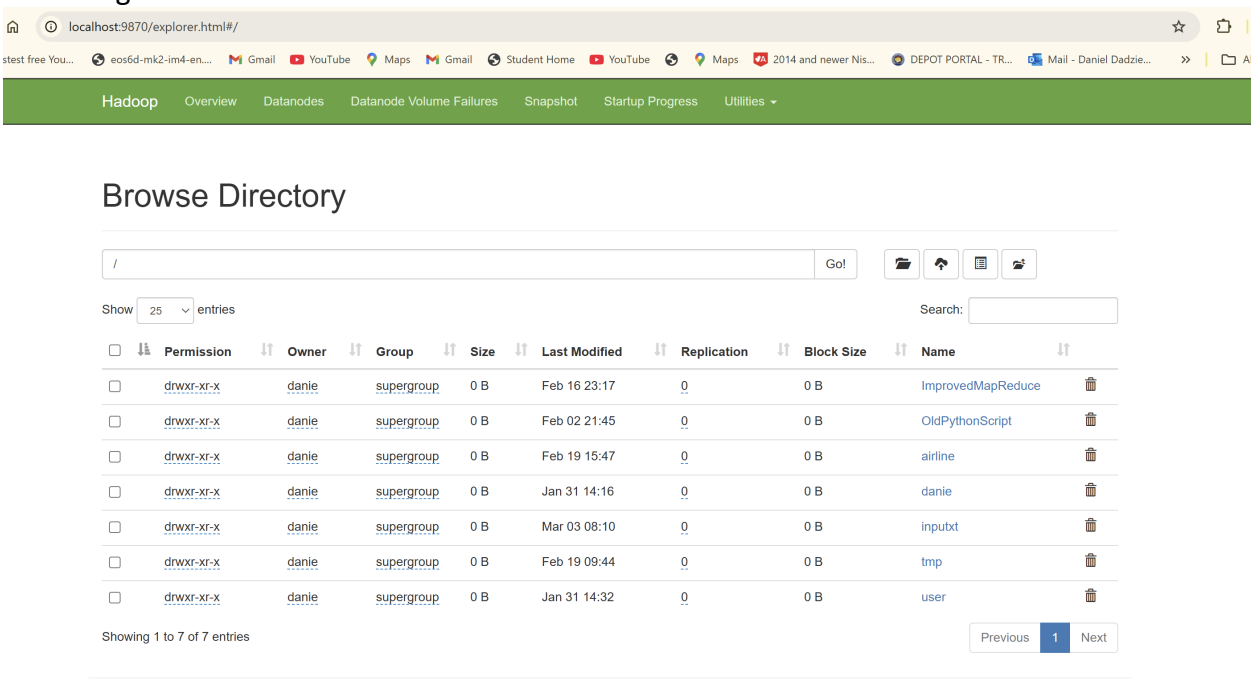
Starting the Hadoop Environment

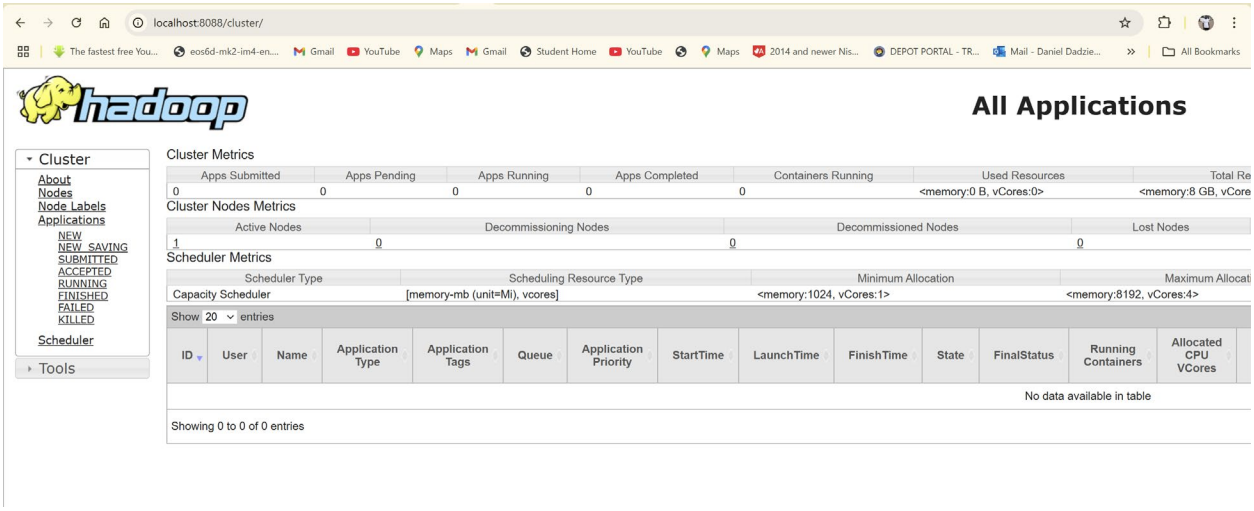


Confirming the webpages for Hadoop and cluster are active.
This is to confirm after starting Hadoop in the terminal the webpage is as well active.



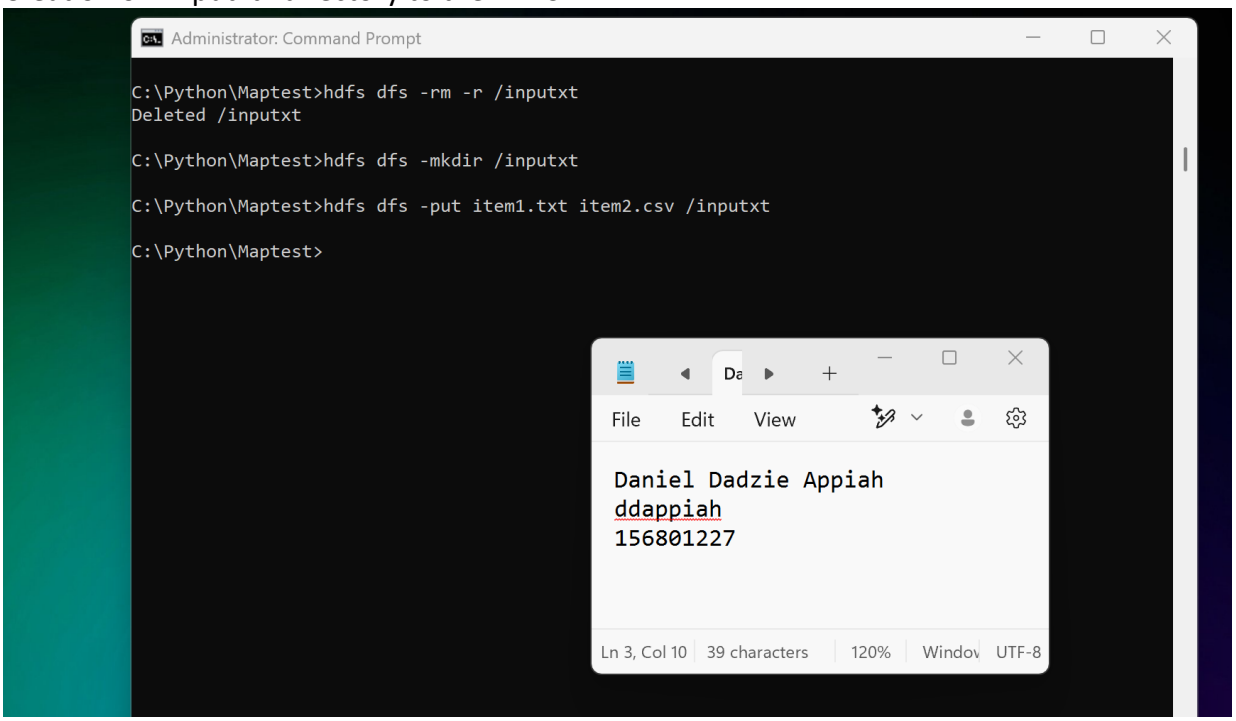
Validating the directories on the HDFS.



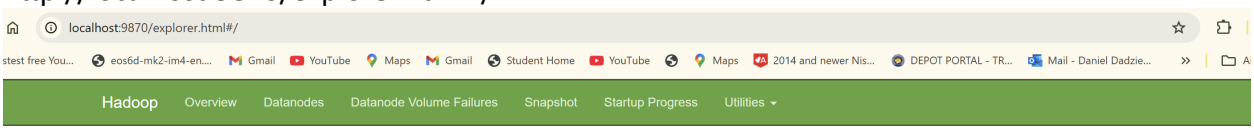


6 Executing the Python scripts via Hadoop Streaming JAR file for implementing multiple input files in Hadoop

Creation of “inputxt” directory to the HDFS



Navigating to the “Hadoop Distributed File System and Validating files inside on http://localhost:9870/explorer.html#/



Browse Directory

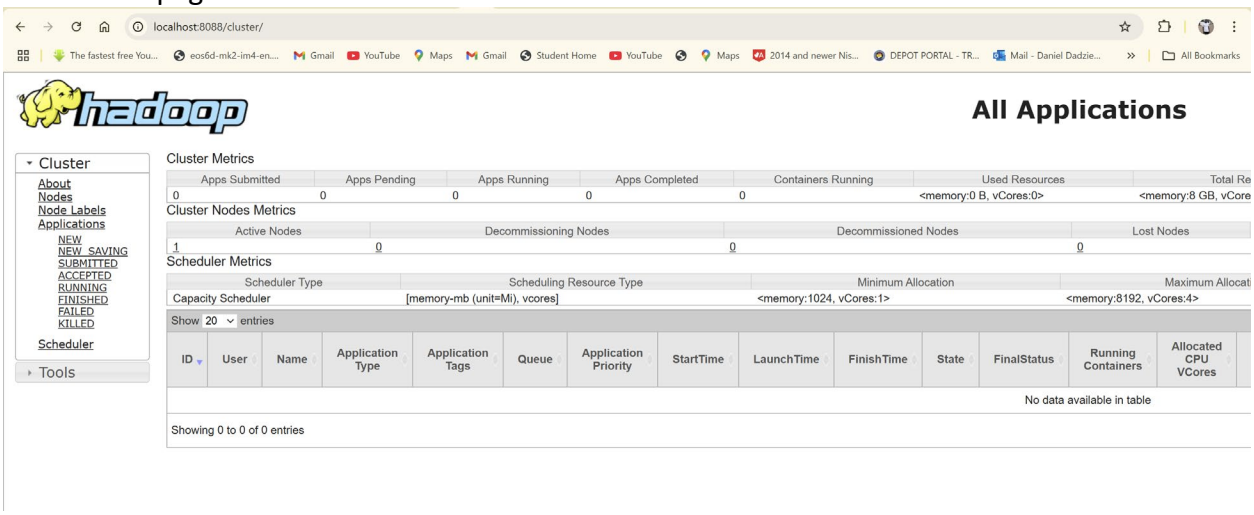
/ Go!

Show 25 entries Search:

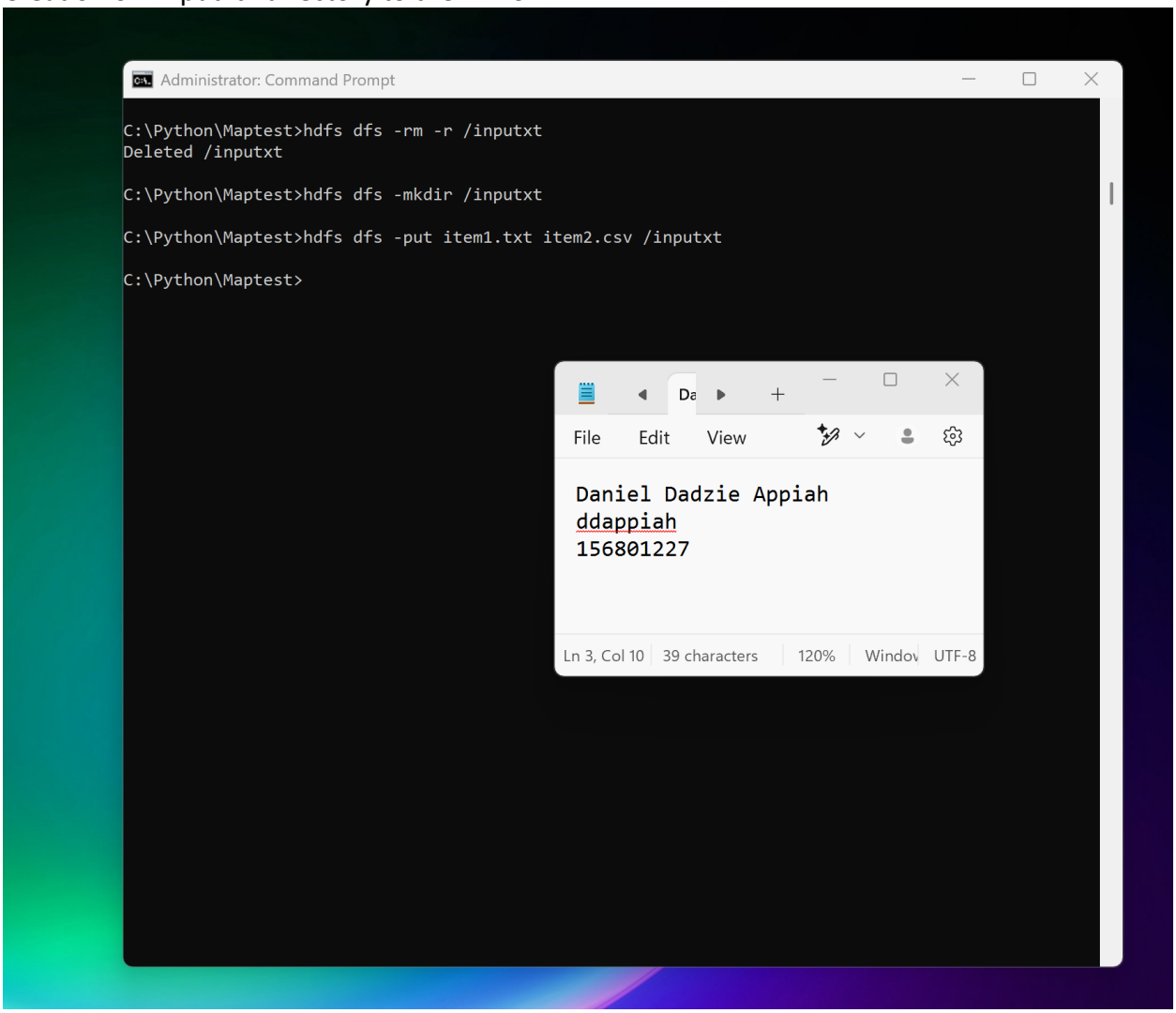
Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxr-xr-x	danie	supergroup	0 B	Feb 16 23:17	0	0 B	ImprovedMapReduce
drwxr-xr-x	danie	supergroup	0 B	Feb 02 21:45	0	0 B	OldPythonScript
drwxr-xr-x	danie	supergroup	0 B	Feb 19 15:47	0	0 B	airline
drwxr-xr-x	danie	supergroup	0 B	Jan 31 14:16	0	0 B	danie
drwxr-xr-x	danie	supergroup	0 B	Mar 03 08:10	0	0 B	inputxt
drwxr-xr-x	danie	supergroup	0 B	Feb 19 09:44	0	0 B	tmp
drwxr-xr-x	danie	supergroup	0 B	Jan 31 14:32	0	0 B	user

Showing 1 to 7 of 7 entries Previous 1 Next

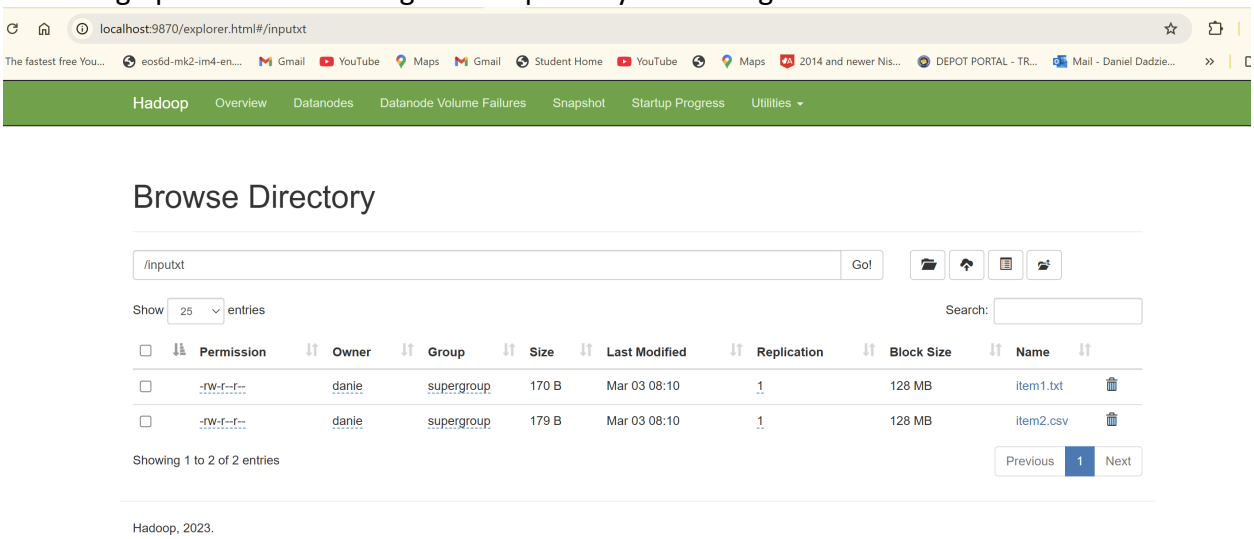
Cluster webpage



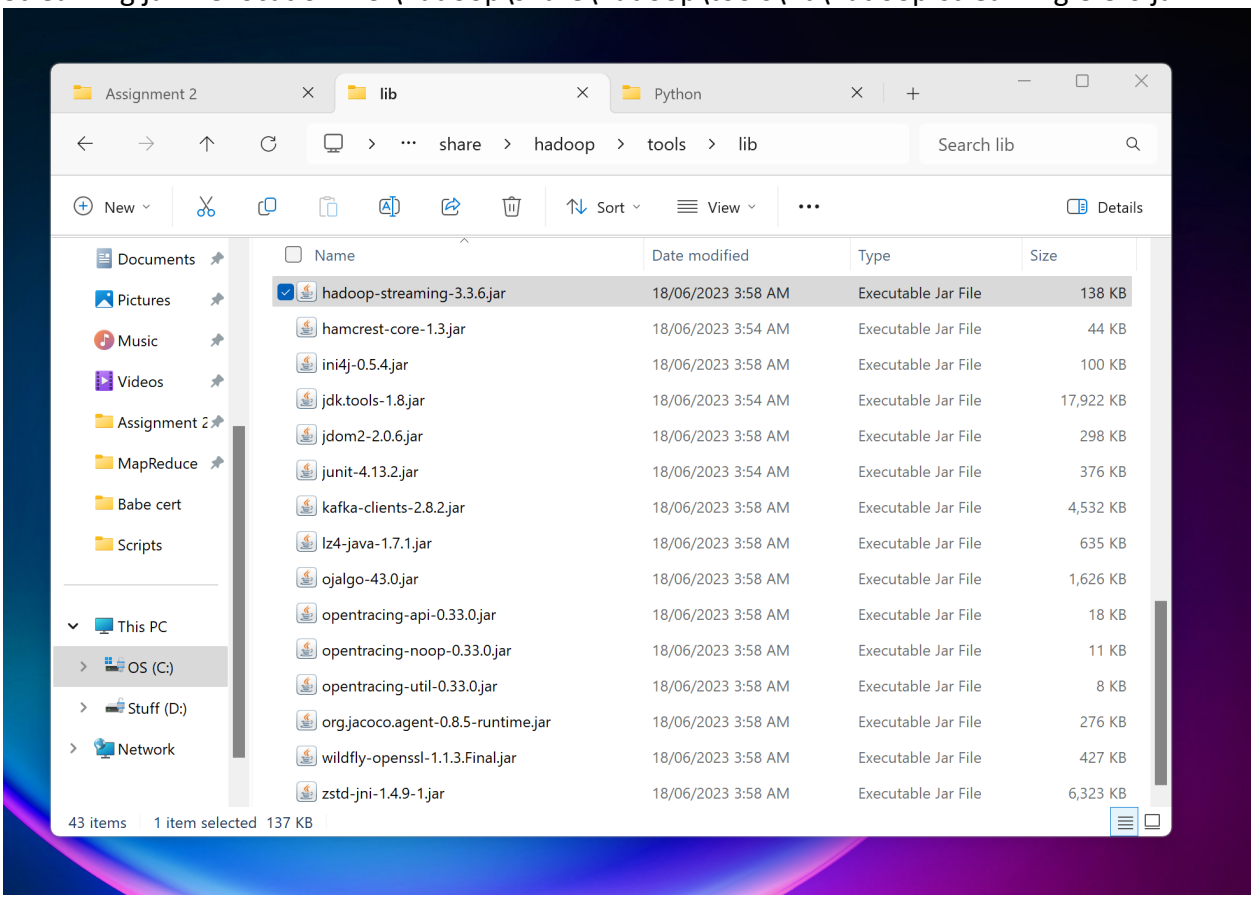
Creation of “inputxt” directory to the HDFS



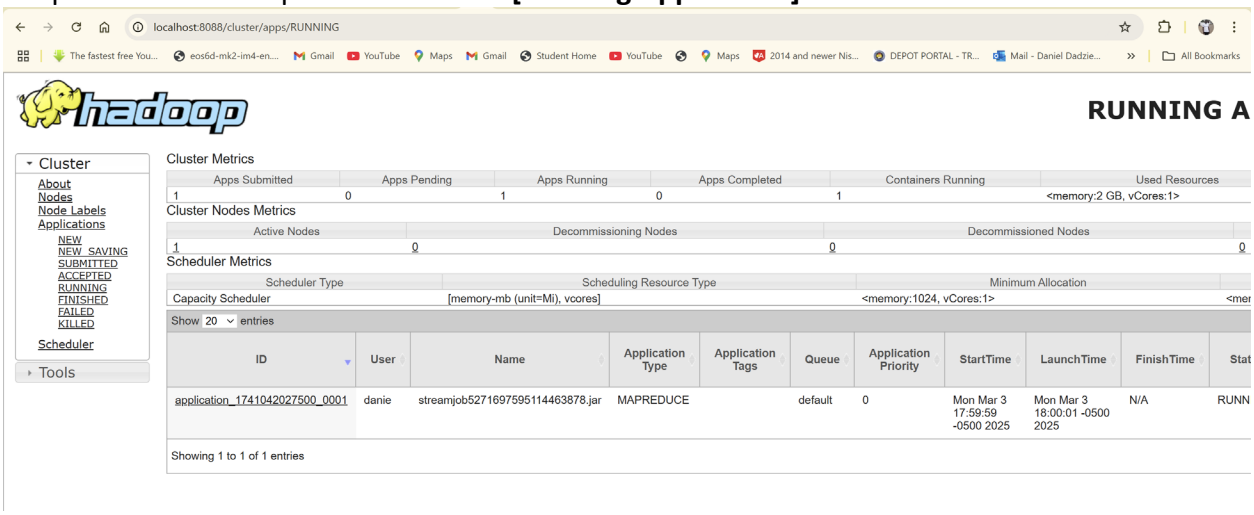
Validating uploaded files through Hadoop File System Using the Terminal



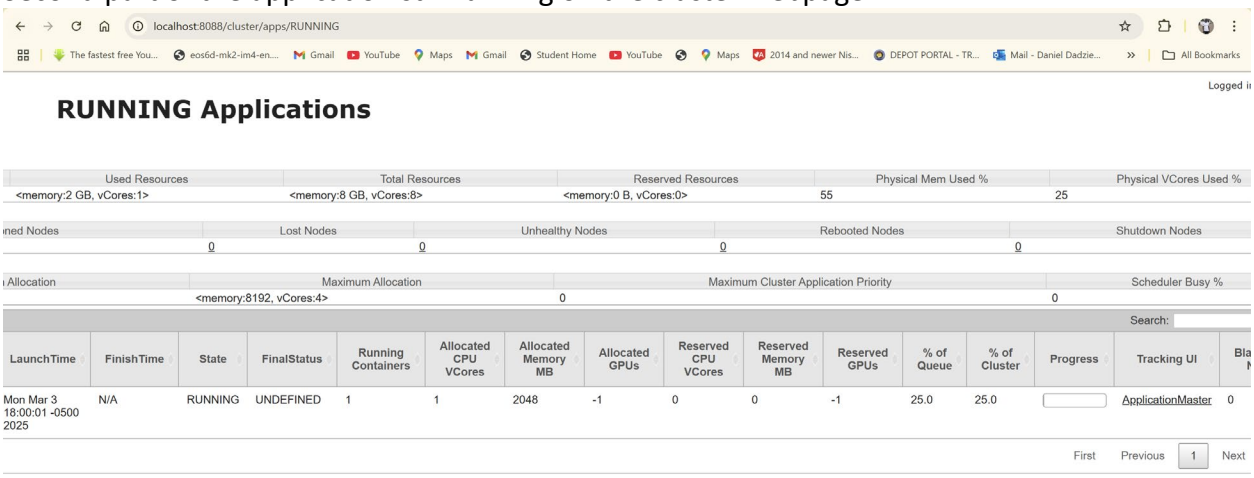
Streaming jar file location: "C:\hadoop\share\hadoop\tools\lib\hadoop-streaming-3.3.6.jar"



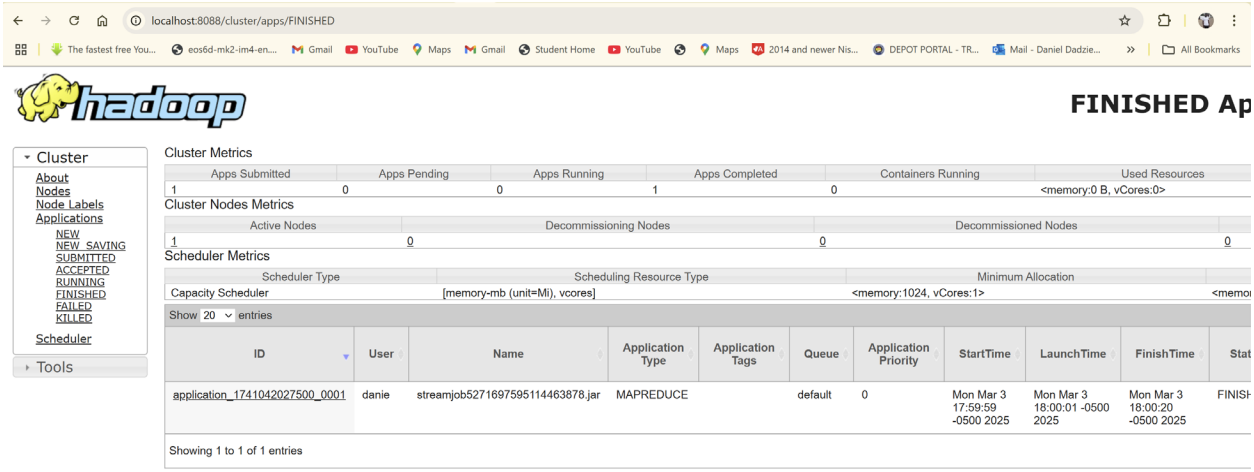
Output on the Hadoop Cluster website **[Running Application]**



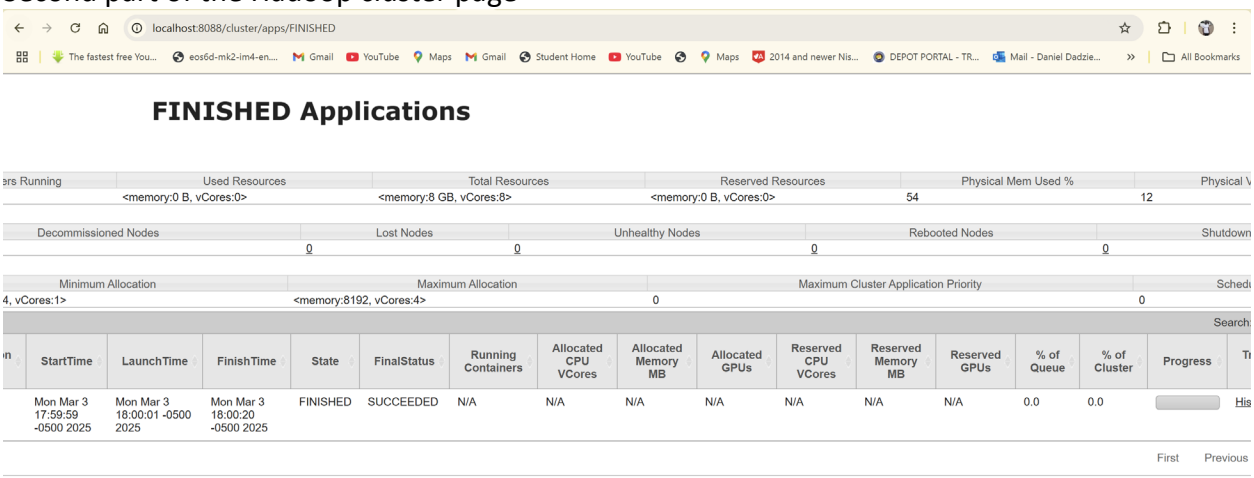
Second part of the application still running on the cluster webpage.



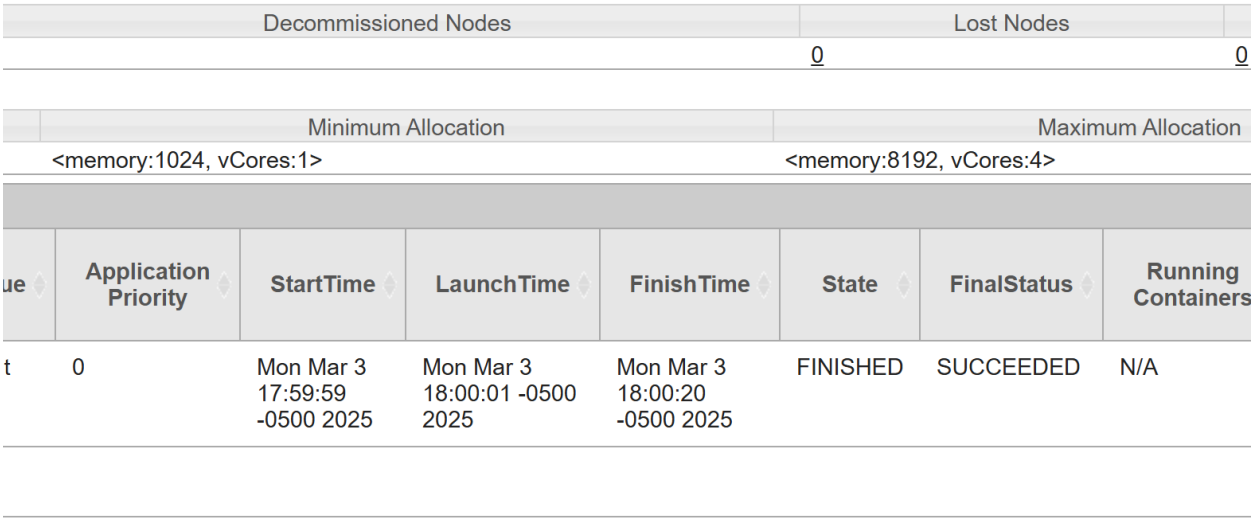
Output on Cluster when the jar streaming is **Completed Running**



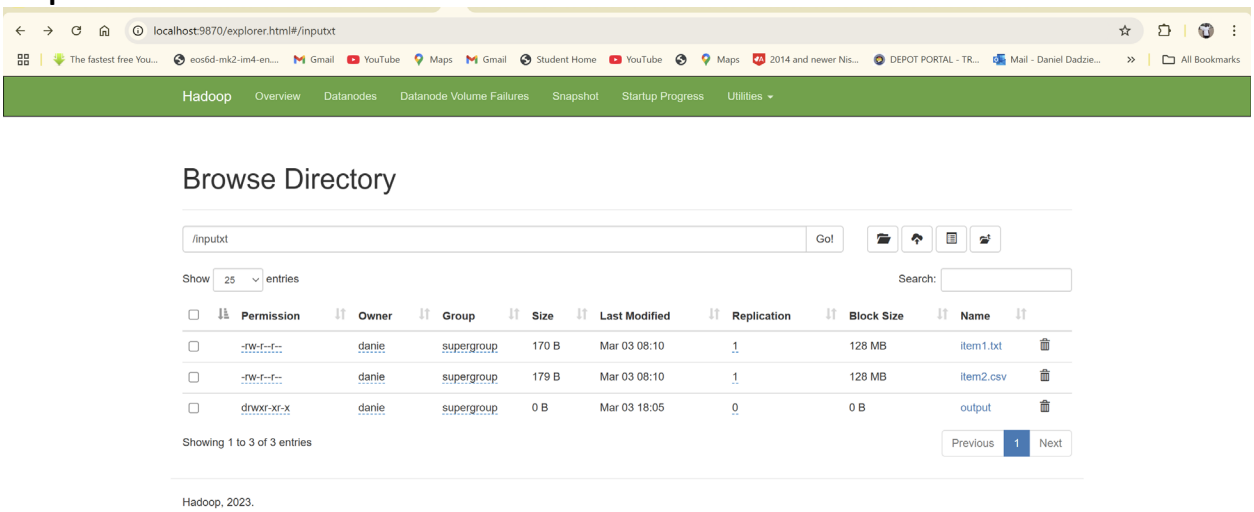
Second part of the Hadoop cluster page



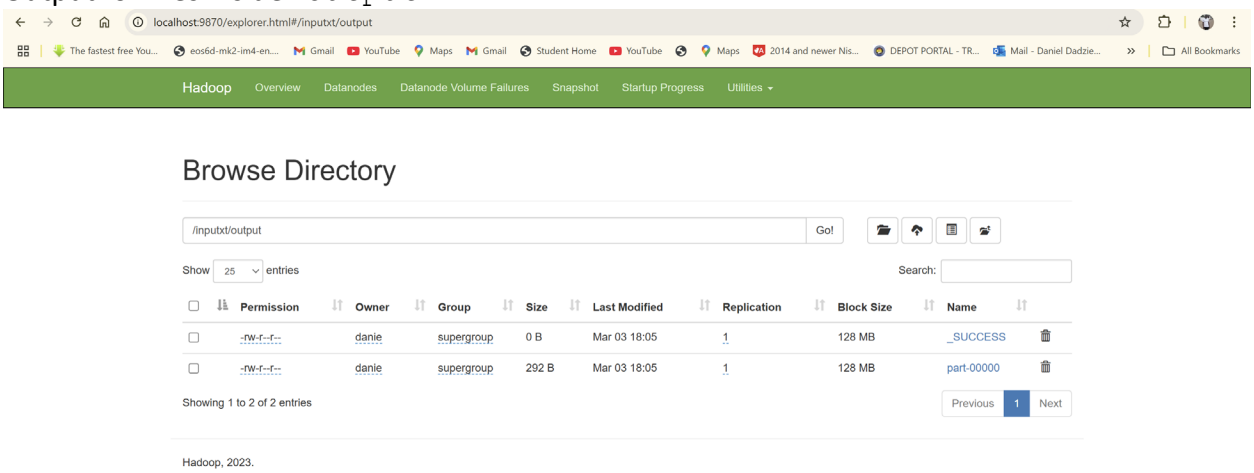
Zooming in on the mapreduce job



Output on the Hadoop website when the jar streaming is **Finished Running and produces an output**



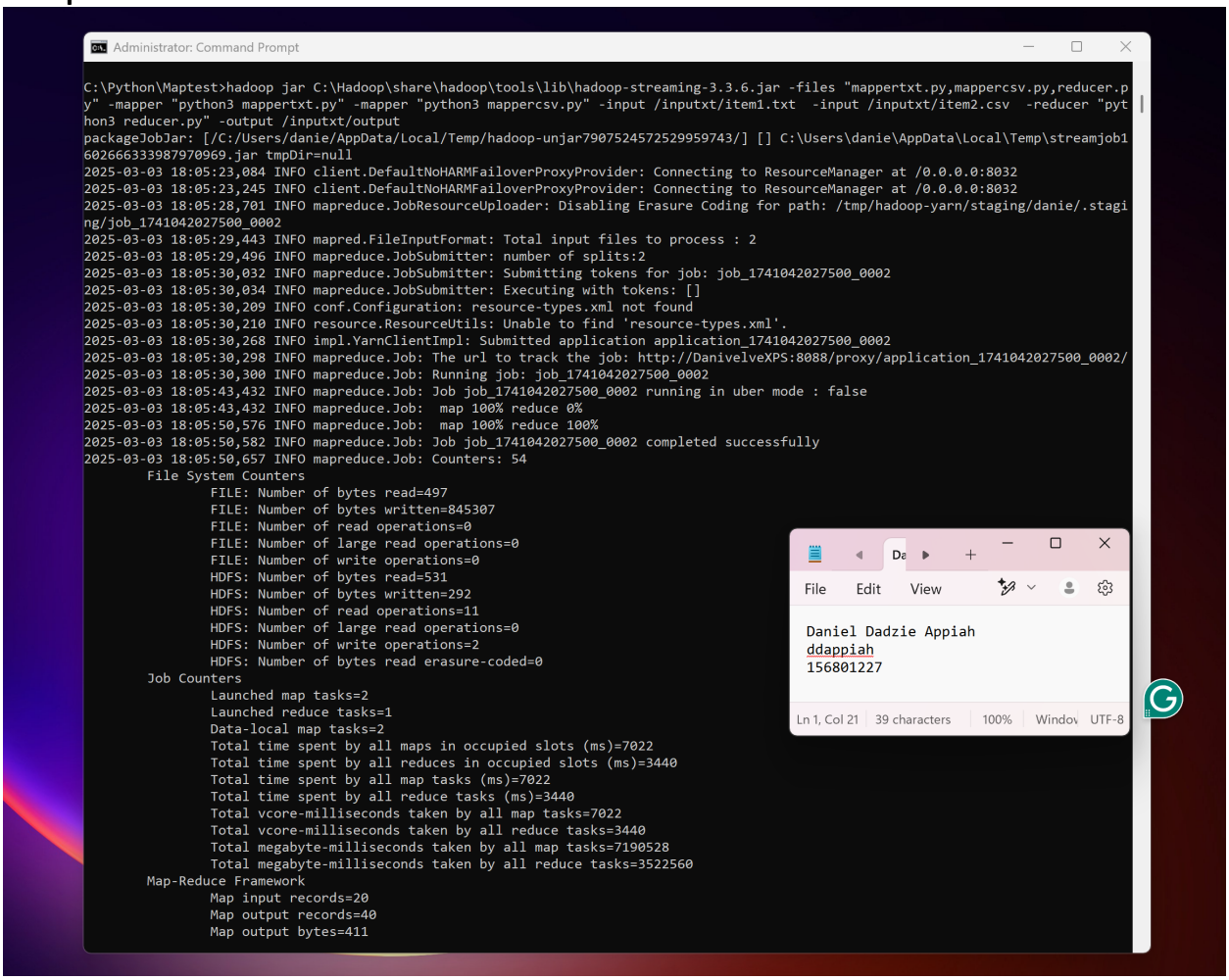
Output for files inside “`output`”



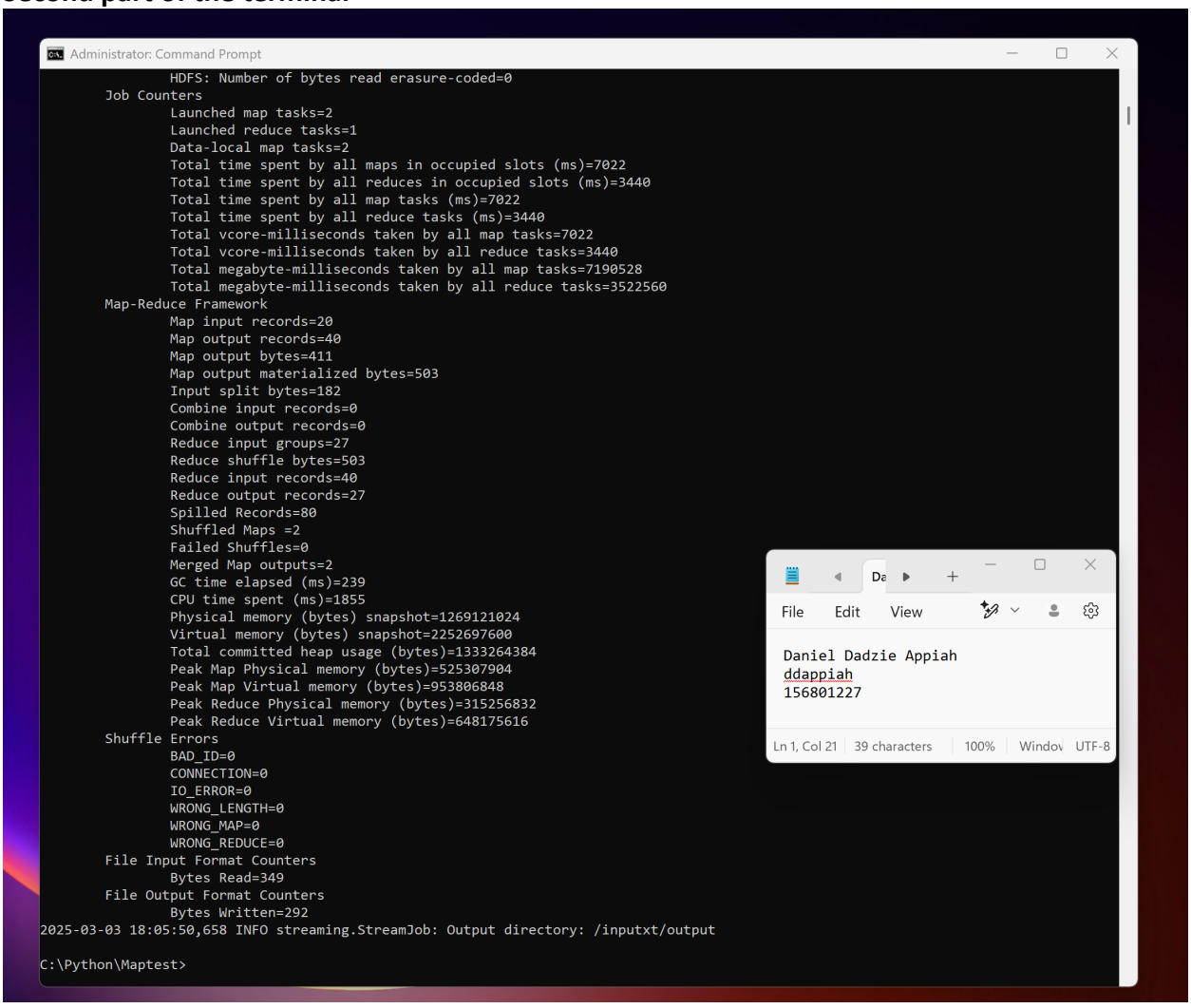
Showing command from terminal **[python Script in Project]**

hadoop jar C:\Hadoop\share\hadoop\tools\lib\hadoop-streaming-3.3.6.jar -files "mappertxt.py,mappercsv.py,reducer.py" -mapper "python3 mappertxt.py" -mapper "python3 mappercsv.py" -input /inputxt/item1.txt -input /inputxt/item2.csv -reducer "python3 reducer.py" -output /inputxt/output

First part of the terminal



Second part of the terminal



7 Analysis for streaming jar job for the multiple input file system

Analysis of the Hadoop Job Execution
Start Time: Mon, Mar 3, 17:59:59 -0500 2025
Launch Time: Mon, Mar 3, 18:00:01 -0500 2025
Finish Time: Mon, Mar 3, 18:00:20 -0500 2025
State: FINISHED
Final Status: SUCCEEDED

Performance Insights
Total Execution Time: 19 seconds (from 18:00:01 to 18:00:20).
Minimal Delay Before Launch: 2 seconds (from 17:59:59 to 18:00:01).
Efficient Resource Allocation:
A short delay before execution suggests quick scheduling.
The job utilized available resources effectively.
No Failures or Retries: Job completed on the first attempt.

Final Analysis
Job executed successfully with optimal efficiency.
Minimal launch delay indicates good resource allocation.
Fast execution time suggests well-optimized job processing.