

2025

Home Project 1

Implement multiple-input-files in Hadoop

DANIEL DADZIE APPIAH | 156801227 | DDAPPIAH

3RD MARCH, 2025.

Table of Contents

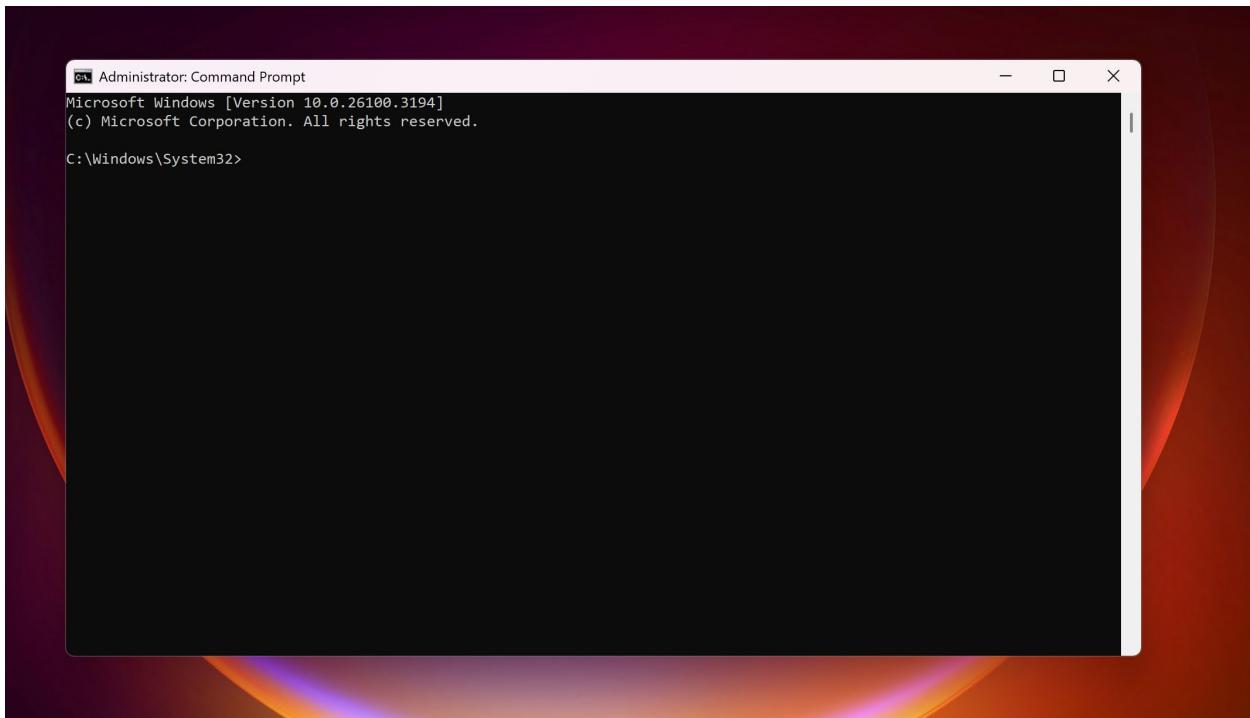
1	Introduction	2
2	O-Screen for Starting Project	2
3	Showing Data input files and Scripts in the folder.	3
4	Validating Folder Contents	5
5	Testing Mapper and Reducer Program	10
6	Executing the Python scripts via Hadoop Streaming JAR file for implementing multiple input files in Hadoop	14
7	Analysis 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 O-Screen for Starting Project

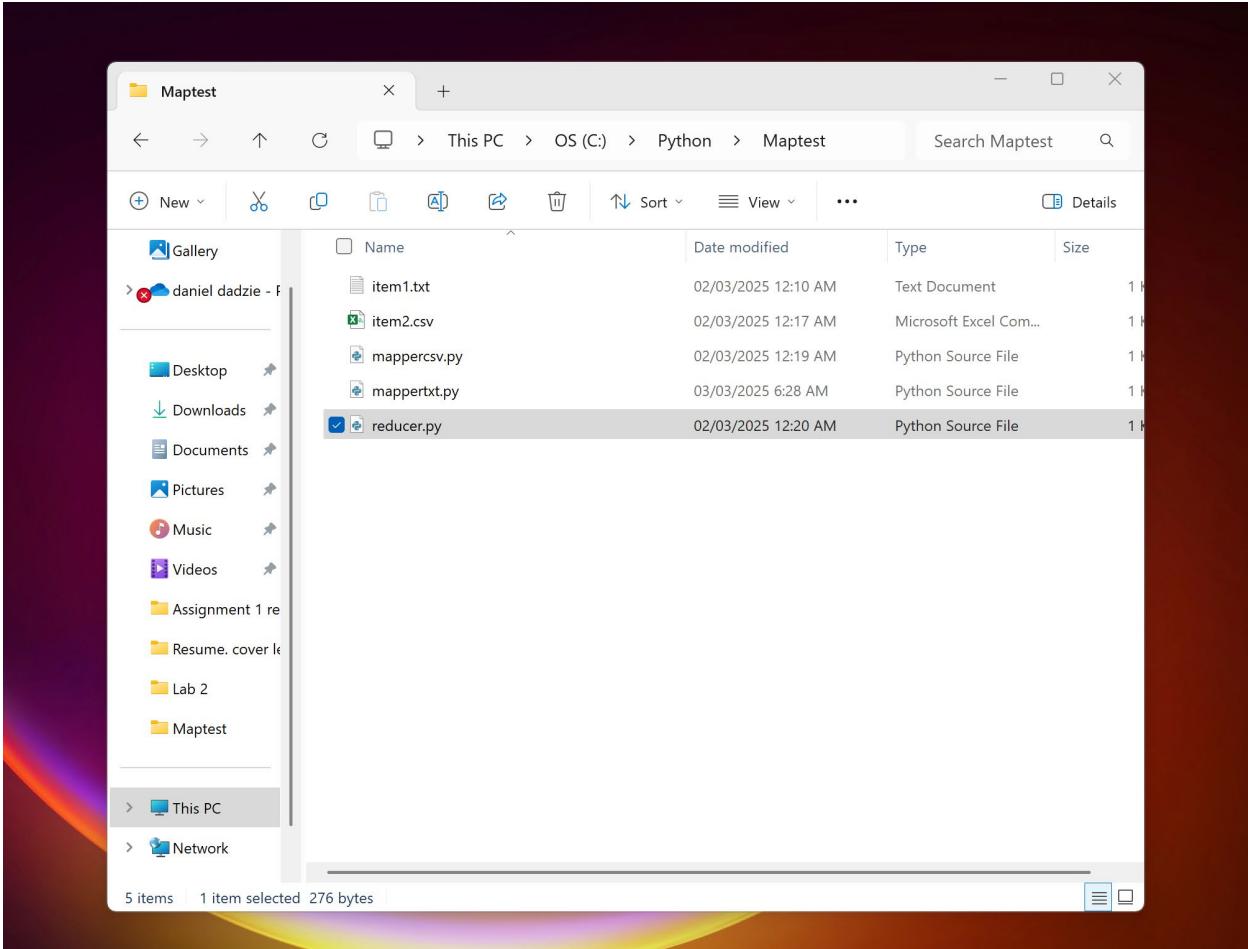
Starting up the command line as an administrator. Use of Windows Terminal.



3 Showing Data input files and Scripts in the folder.

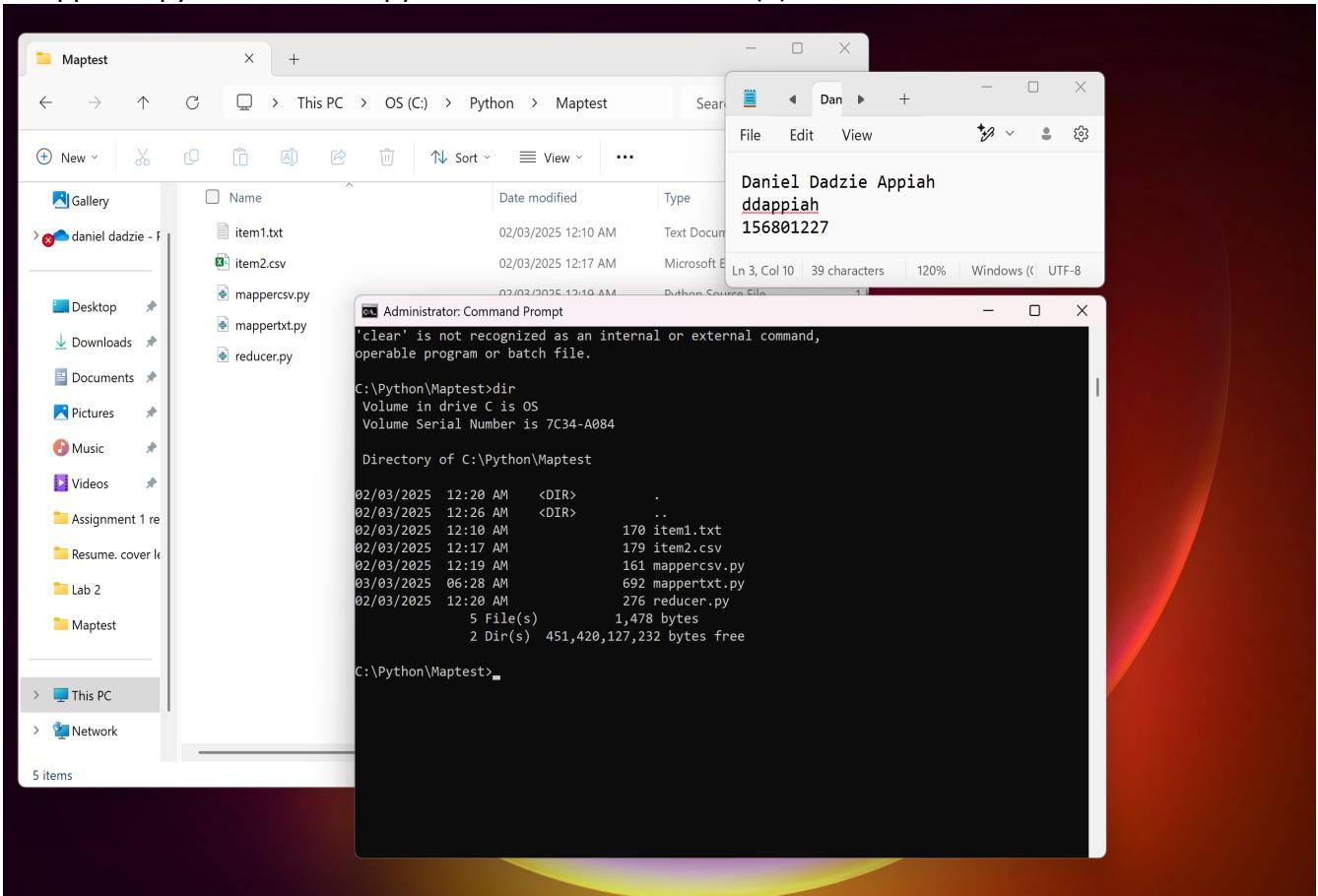
Create a Sample Input File.

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.



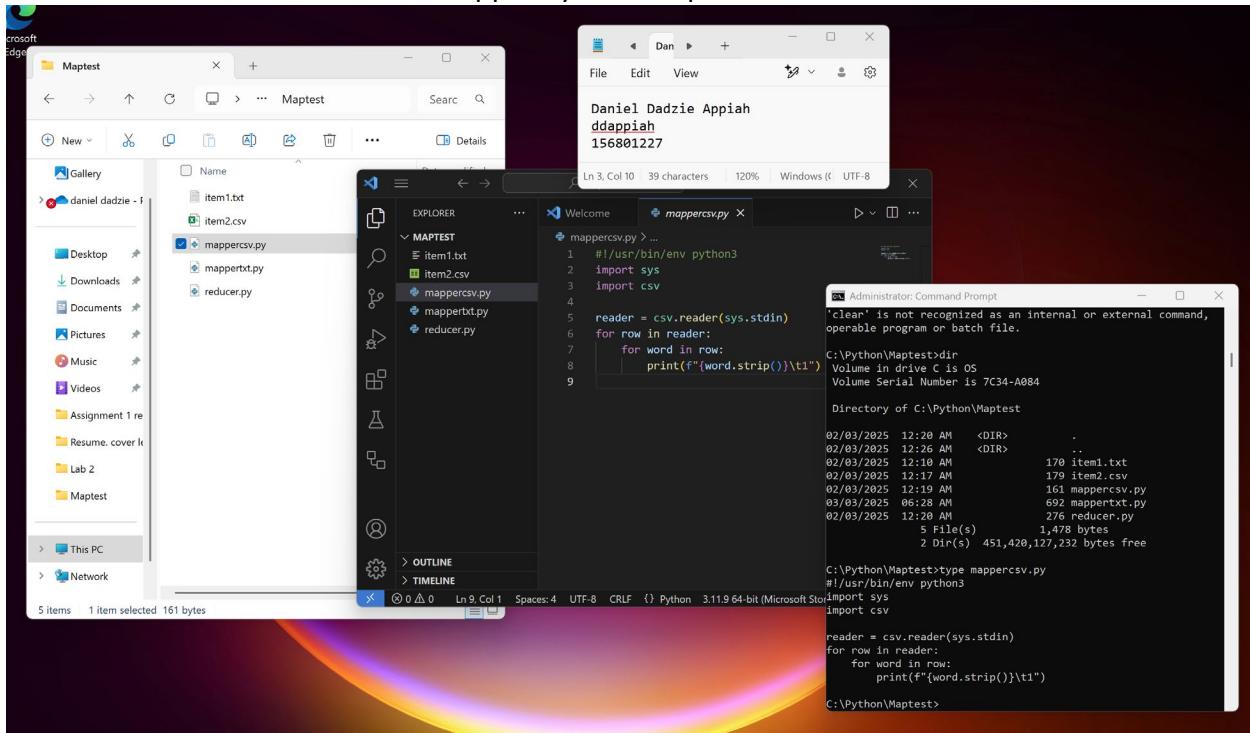
Folder Content

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

The screenshot shows a Windows desktop environment. In the center is a Microsoft Edge browser window displaying a login page for 'Dan' with fields for 'Name' (Daniel Dadzie Appiah), 'ddappiah', and '156801227'. To the left of the browser is a File Explorer window titled 'Maptest' showing files: item1.txt, item2.csv, mappercsv.py, mappertxt.py, and reducer.py. Below the browser is a Command Prompt window with the following text:

```
C:\Python\Maptest>type mappertxt.py
#!/usr/bin/env python
"""A more advanced Mapper, using Python iterators and generator functions.

import sys

def read_input(file):
    for line in file:
        # split the line into words
        yield line.split()

def main(separator='\t'):
    # input comes from STDIN (standard input)
    data = read_input(sys.stdin)

    for words in data:
        # write the results to STDOUT (standard output);
        # what we output here will be the input for the
        # Reduce step, i.e., the input for reducer.py
        #
        # tab-delimited; the trivial word count is 1
        for word in words:
            print('%s%s%d' % (word, separator, 1))

if __name__ == "__main__":
    main()
C:\Python\Maptest>
```

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

The screenshot shows a Windows desktop environment. In the center is a Microsoft Edge browser window displaying a login page for 'Dan' with fields for 'Name' (Daniel Dadzie Appiah), 'ddappiah', and '156801227'. To the left of the browser is a File Explorer window titled 'Maptest' showing files: item1.txt, item2.csv, mappercsv.py, mappertxt.py, and reducer.py. Below the browser is a Visual Studio Code window with the file 'reducer.py' open. The code is as follows:

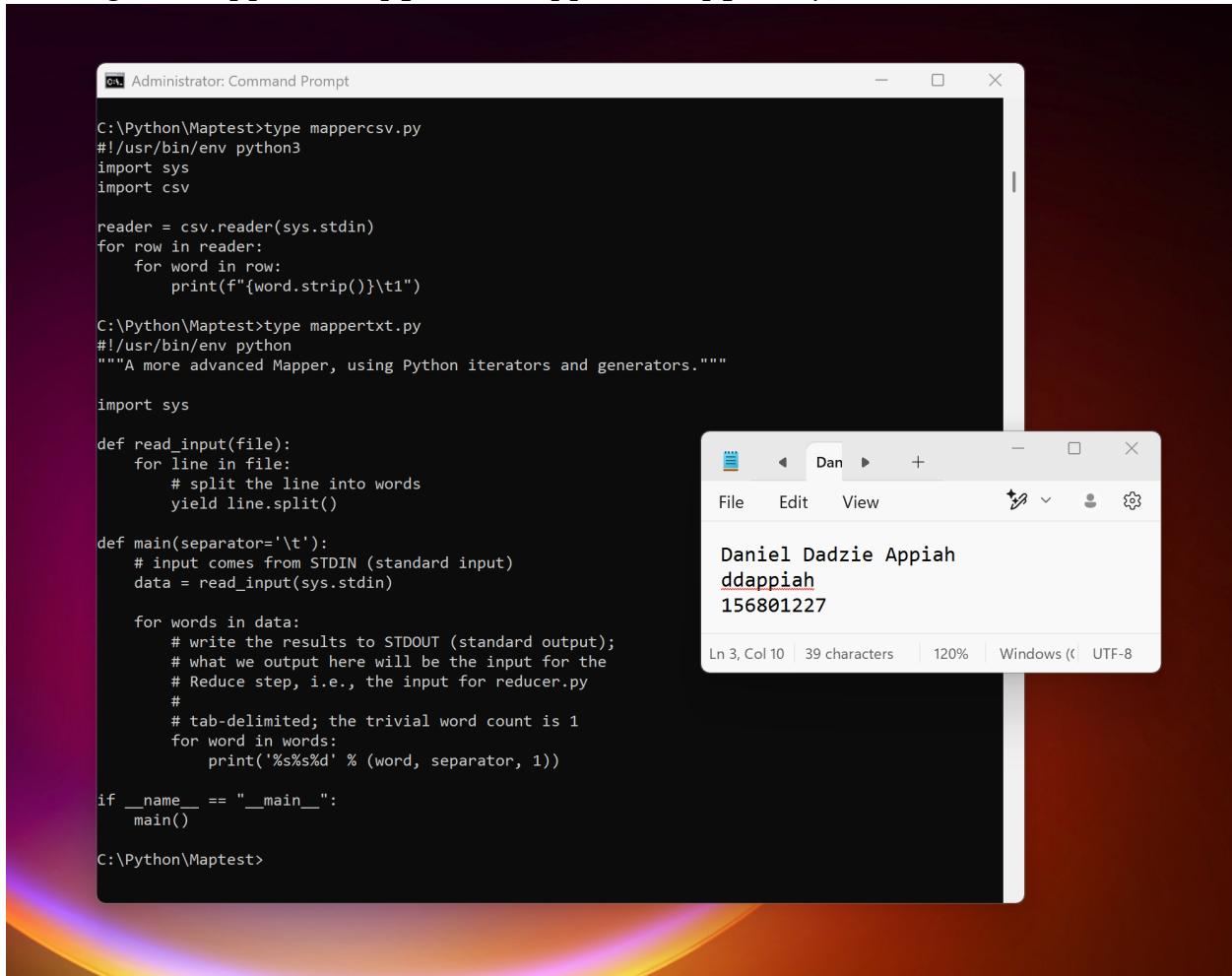
```
#!/usr/bin/env python3
import sys
from collections import defaultdict

word_counts = defaultdict(int)

for line in sys.stdin:
    word, count = line.strip().split("\t")
    word_counts[word] += int(count)

for word, count in word_counts.items():
    print("{word}\t{count}")
```

Checking the “mappercsv.py” and “mappertxt.py” Script from the windows terminal.



```
C:\Python\Maptest>type mappercsv.py
#!/usr/bin/env python3
import sys
import csv

reader = csv.reader(sys.stdin)
for row in reader:
    for word in row:
        print(f"{word.strip()}\t1")

C:\Python\Maptest>type mappertxt.py
#!/usr/bin/env python
"""A more advanced Mapper, using Python iterators and generators."""

import sys

def read_input(file):
    for line in file:
        # split the line into words
        yield line.split()

def main(separator='\t'):
    # input comes from STDIN (standard input)
    data = read_input(sys.stdin)

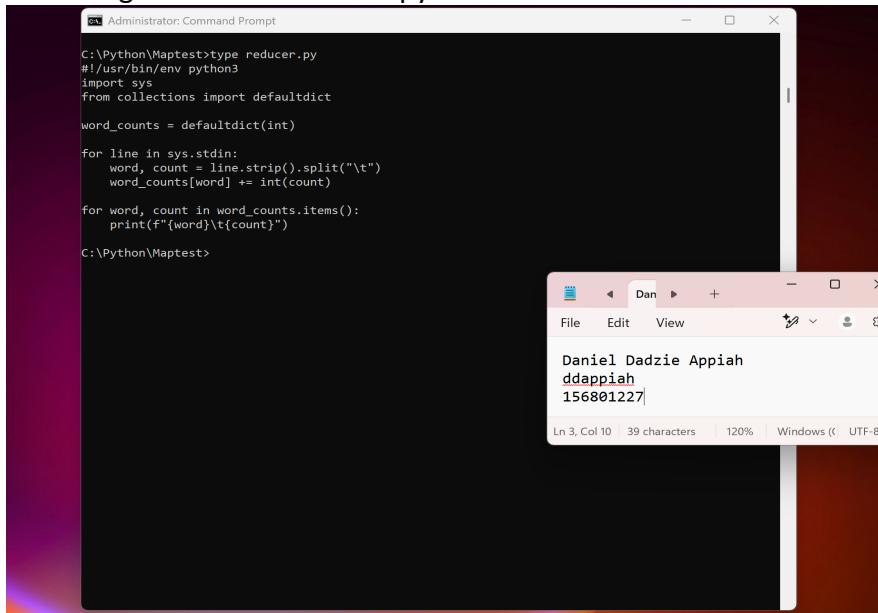
    for words in data:
        # write the results to STDOUT (standard output);
        # what we output here will be the input for the
        # Reduce step, i.e., the input for reducer.py
        #
        # tab-delimited; the trivial word count is 1
        for word in words:
            print('%s%s%d' % (word, separator, 1))

if __name__ == "__main__":
    main()

C:\Python\Maptest>
```

Daniel Dadzie Appiah
ddappiah
156801227

Validating content in “reducer.py”



```
C:\Python\Maptest>type reducer.py
#!/usr/bin/env python3
import sys
from collections import defaultdict

word_counts = defaultdict(int)

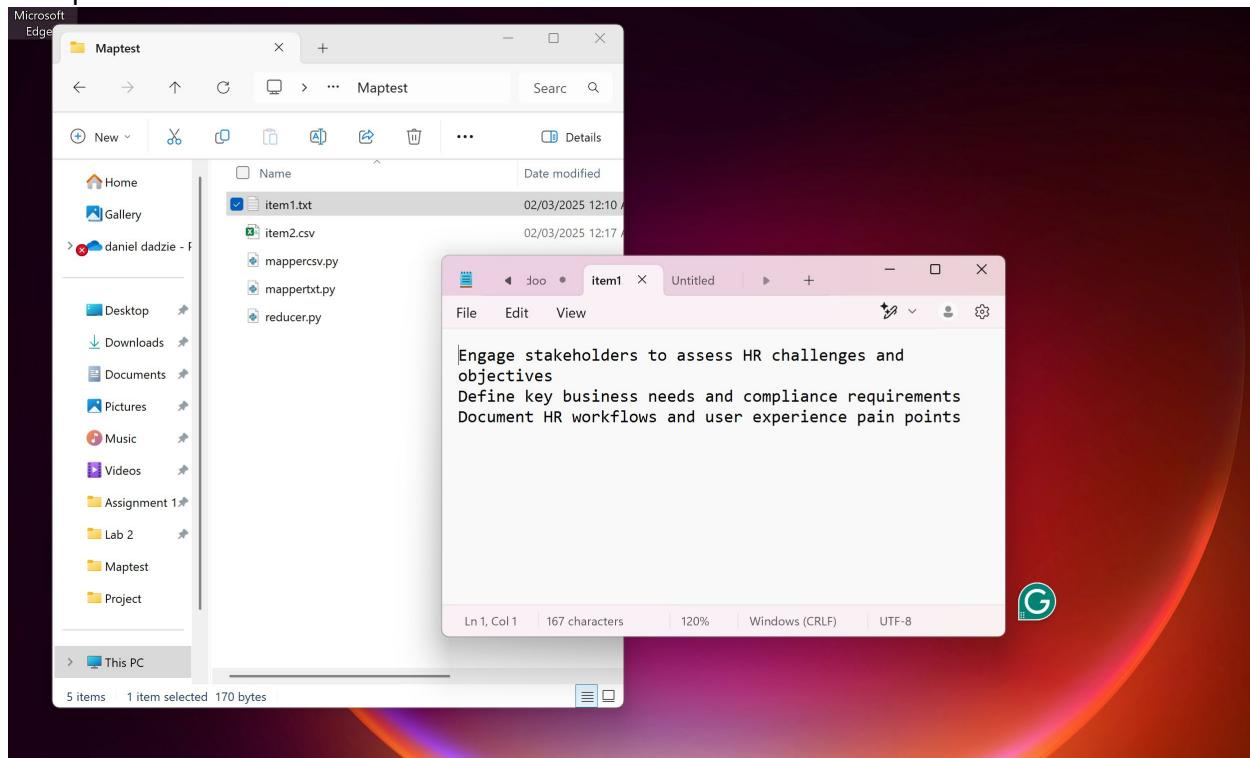
for line in sys.stdin:
    word, count = line.strip().split("\t")
    word_counts[word] += int(count)

for word, count in word_counts.items():
    print(f'{word}\t{count}')

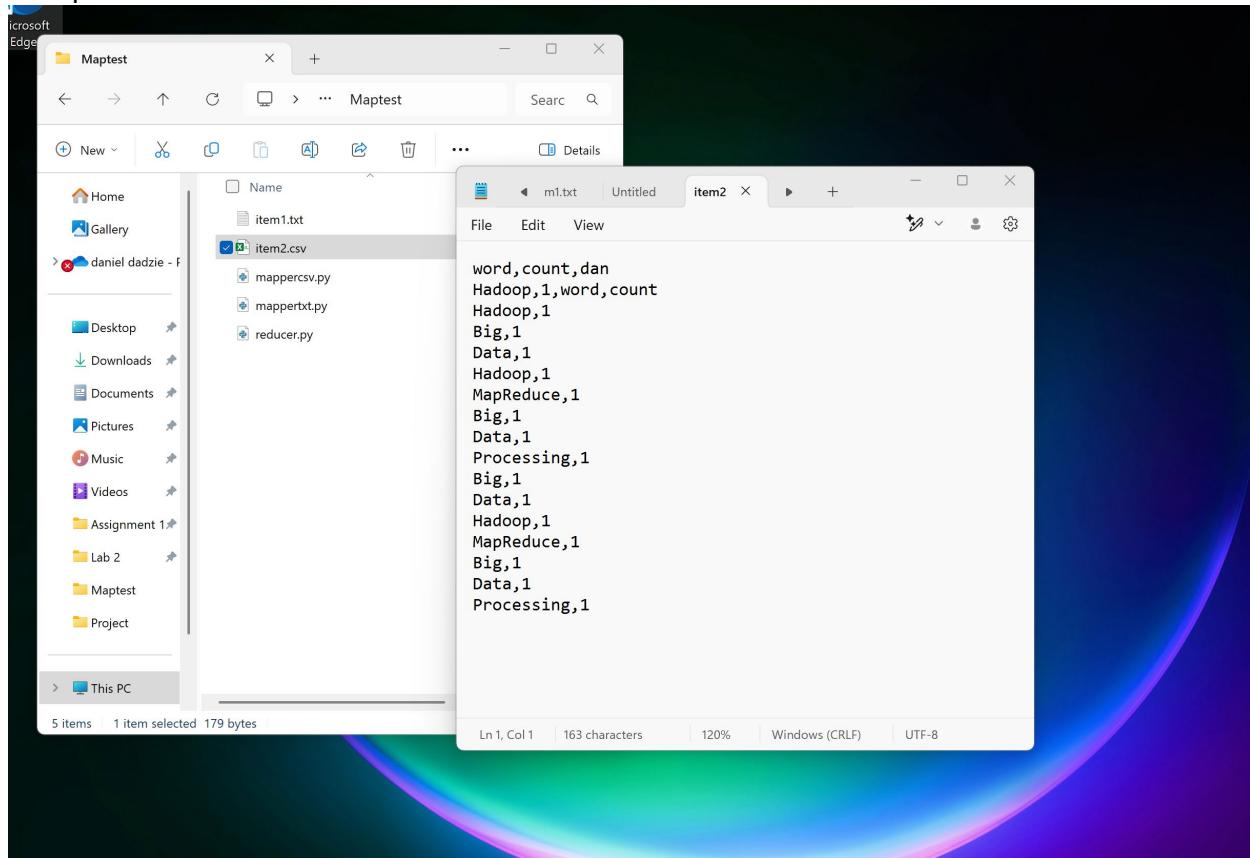
C:\Python\Maptest>
```

Daniel Dadzie Appiah
ddappiah
156801227

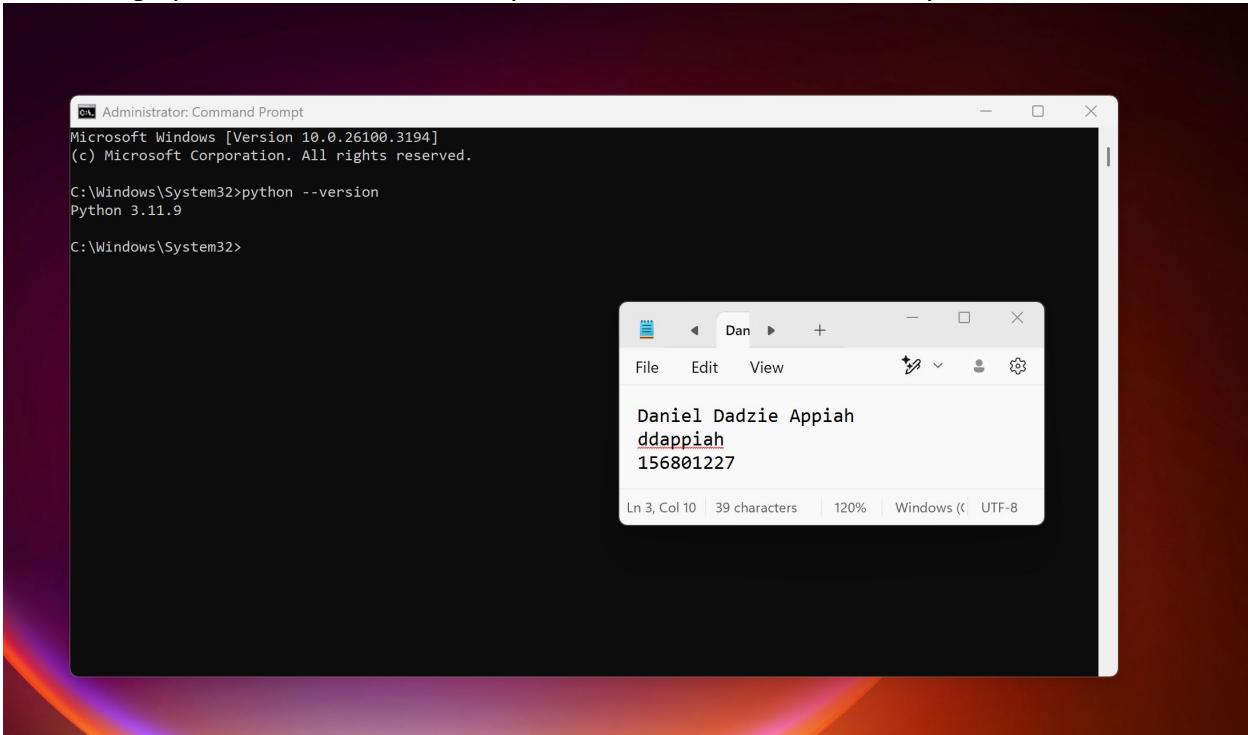
Notepad file content of "item1.txt"



Notepad file content of "item2.csv"



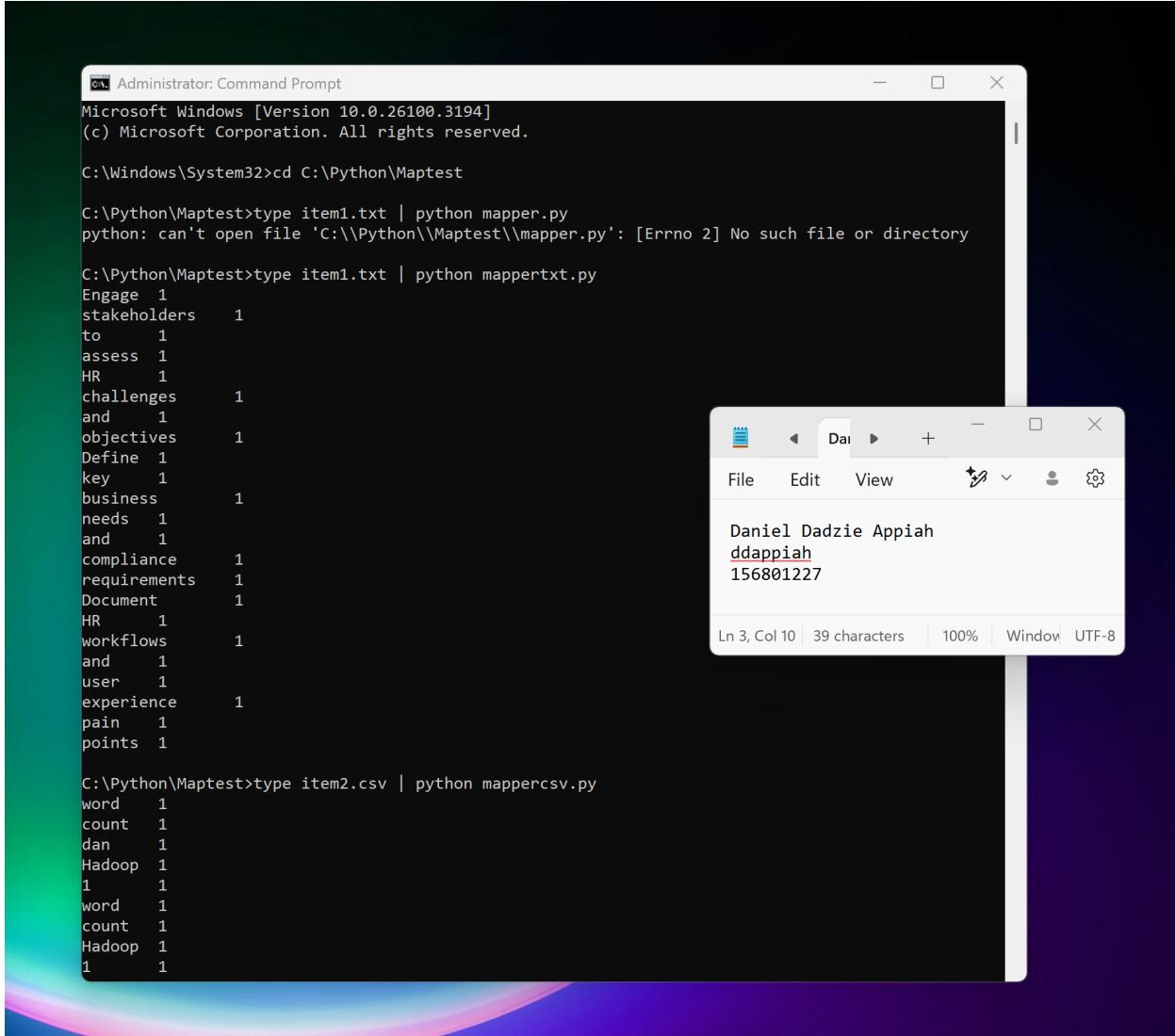
Confirming Python Installation on the system. The version available is Python 3.11.9



5 Testing Mapper and Reducer Program

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

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



```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.26100.3194]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>cd C:\Python\Maptest

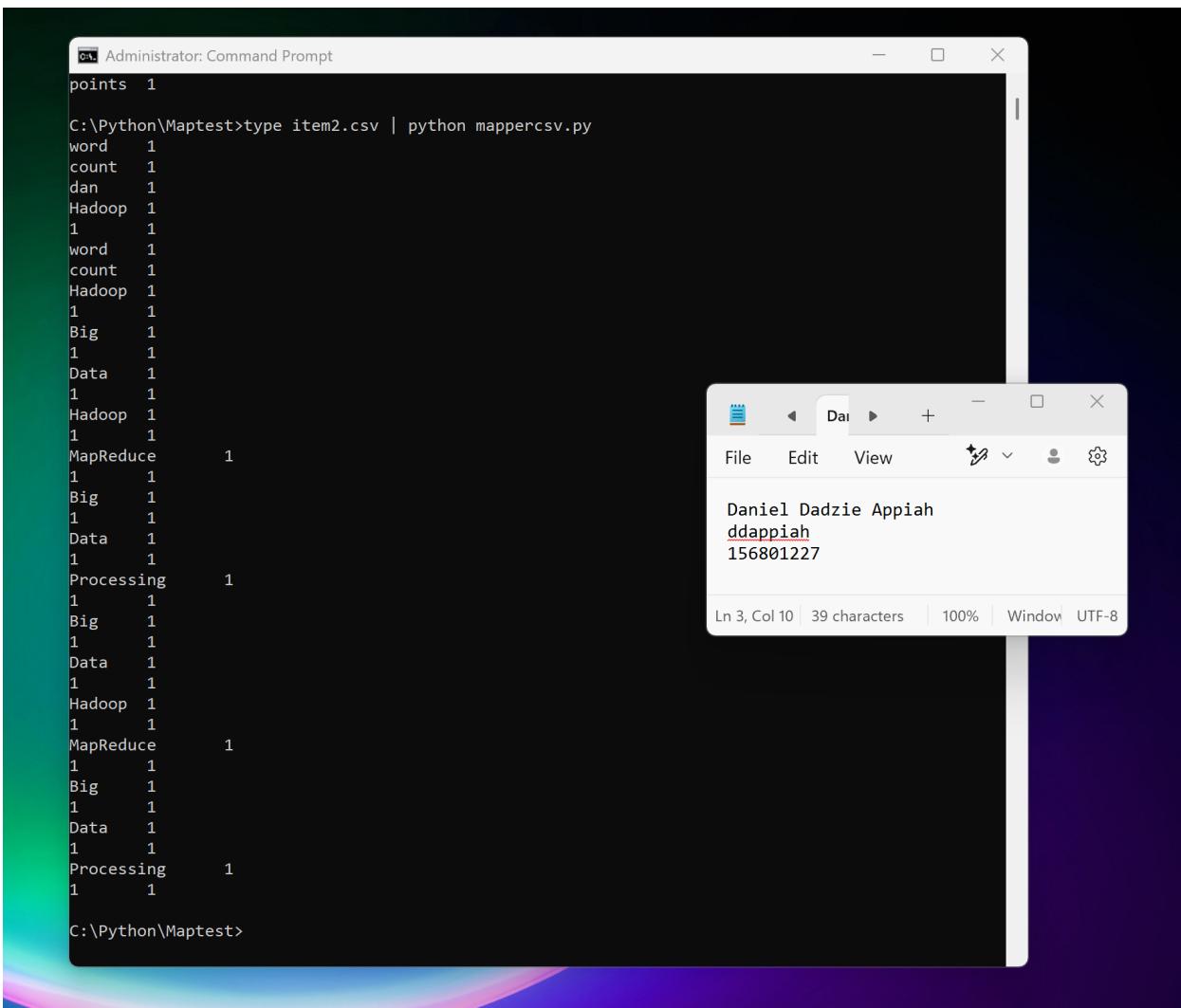
C:\Python\Maptest>type item1.txt | python mapper.py
python: can't open file 'C:\\Python\\Maptest\\mapper.py': [Errno 2] No such file or directory

C:\Python\Maptest>type item1.txt | python mappertxt.py
Engage 1
stakeholders 1
to 1
assess 1
HR 1
challenges 1
and 1
objectives 1
Define 1
key 1
business 1
needs 1
and 1
compliance 1
requirements 1
Document 1
HR 1
workflows 1
and 1
user 1
experience 1
pain 1
points 1

C:\Python\Maptest>type item2.csv | python mappercsv.py
word 1
count 1
dan 1
Hadoop 1
1 1
word 1
count 1
Hadoop 1
1 1
```

Mapper on the Content of the “item2.csv”

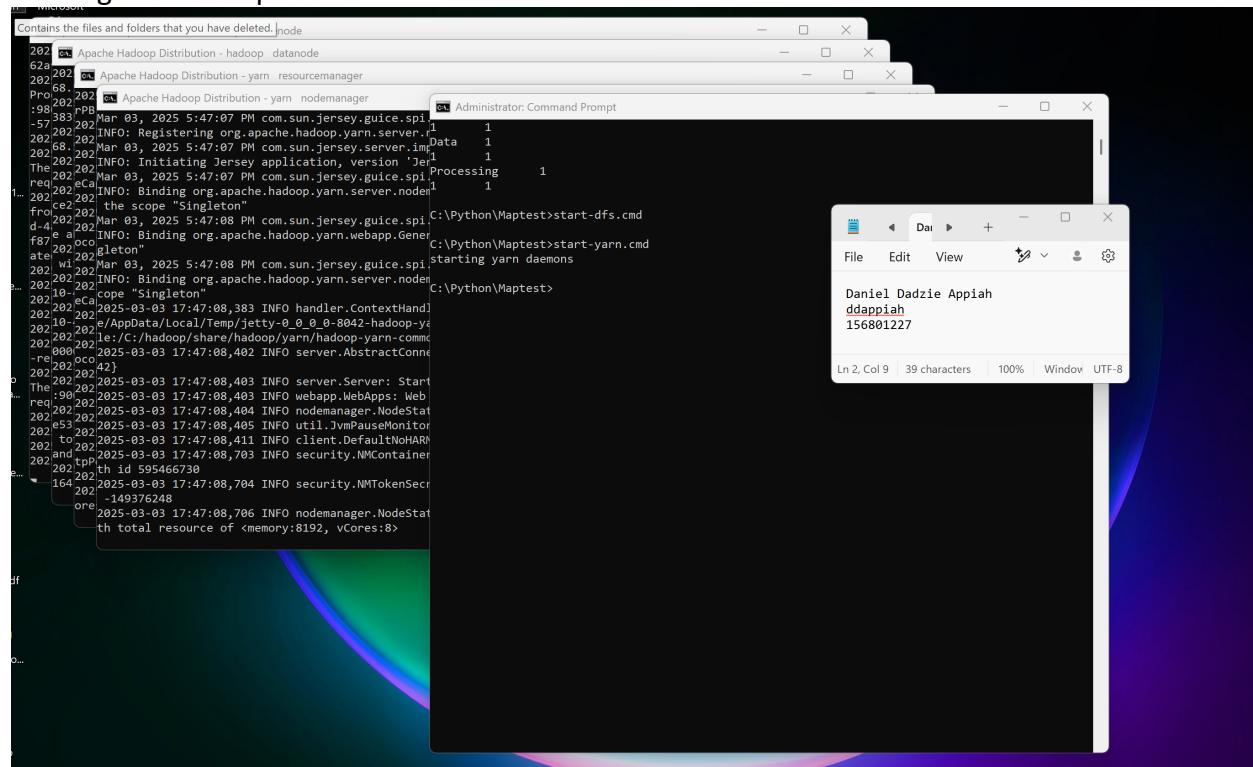
Command- type item2.csv | python mappercsv.py



The image shows a Windows desktop environment. In the foreground, there is a Command Prompt window titled "Administrator: Command Prompt". The command typed is "C:\Python\Maptest>type item2.csv | python mappercsv.py". The output of the command is displayed in the window, showing various words and their counts, such as "points 1", "word 1", "count 1", etc. In the background, there is a Notepad window titled "Dai". The content of the Notepad window is "Daniel Dadzie Appiah" followed by a red underline over "ddappiah", and the number "156801227". The status bar at the bottom of the Notepad window shows "Ln 3, Col 10 | 39 characters | 100% | Window | UTF-8".

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.

The screenshot shows a web browser displaying the Hadoop Overview page at localhost:9000. The URL bar shows the address. The page has a green header bar with tabs for Hadoop, Overview, Datanodes, Datanode Volume Failures, Snapshot, Startup Progress, and Utilities.

Overview 'localhost:9000' (active)

Started:	Sun Mar 02 20:20:36 -0500 2025
Version:	3.3.6, r1be78238728da9266a4f88195058f08fd012bf9c
Compiled:	Sun Jun 18 03:22:00 -0500 2023 by ubuntu from (HEAD detached at release-3.3.6-RC1)
Cluster ID:	CID-141c13a6-a263-42ca-bf87-228bf4657bc9
Block Pool ID:	BP-484813699-192.168.56.1-1738350689164

Summary

Security is off.
Safemode is off.
69 files and directories, 34 blocks (34 replicated blocks, 0 erasure coded block groups) = 103 total filesystem object(s).

Validating the directories on the HDFS.

The screenshot shows a web browser window with the URL `localhost:9870/explorer.html#`. The page title is "Browse Directory". The main content is a table listing files in the root directory:

	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	inputtxt
□	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

Browse Directory

The screenshot shows the same HDFS file browser interface. The table lists the following files in the root directory:

	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	inputtxt
□	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

The screenshot shows the Hadoop cluster monitoring interface. The top navigation bar includes links for "About", "Nodes", "Node Labels", "Applications", "Scheduler", and "Tools". The main content area is titled "All Applications".

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Used Resources	Total Re
0	0	0	0	0	<memory:0 B, vCores:0>	<memory:8 GB, vCore

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes
1	0	0	0

Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocat
Capacity Scheduler	[memory-mb (unit=Mi), vcores]	<memory:1024, vCores:1>	<memory:8192, vCores:4>

Applications

ID	User	Name	Application Type	Application Tags	Queue	Application Priority	StartTime	LaunchTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU Vcores
No data available in table													

Showing 0 to 0 of 0 entries

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

Creation of “inputtxt” directory to the HDFS

The screenshot shows a Windows Command Prompt window titled "Administrator: Command Prompt" and a text editor window. The command prompt shows the following sequence of commands:

```
C:\Python\Maptest>hdfs dfs -rm -r /inputtxt
Deleted /inputtxt

C:\Python\Maptest>hdfs dfs -mkdir /inputtxt

C:\Python\Maptest>hdfs dfs -put item1.txt item2.csv /inputtxt

C:\Python\Maptest>
```

The text editor window displays the contents of the file "item1.txt" which contains:

Daniel Dadzie Appiah
ddappiah
156801227

Below the text editor, status information is shown: Ln 3, Col 10 | 39 characters | 120% | Window | UTF-8.

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

The screenshot shows a web browser window for the Hadoop DFS browser at the URL <http://localhost:9870/explorer.html#/>. The page has a green header bar with links for "Hadoop", "Overview", "Datanodes", "Datanode Volume Failures", "Snapshot", "Startup Progress", and "Utilities". The main content area is titled "Browse Directory" and shows a list of files and directories under the root path "/".

Browse Directory

The screenshot shows a table listing the contents of the root directory. The columns are: Show (dropdown set to 25), Permission, Owner, Group, Size, Last Modified, Replication, Block Size, Name, and a trash bin icon. The table rows are:

Show	25	entries	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	inputtxt	
			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

The screenshot shows the Hadoop Cluster Metrics page at localhost:8088/cluster/. The left sidebar has sections for Cluster (About, Nodes, Node Labels, Applications), Scheduler (Scheduler, Tools), and a status bar with 'Scheduler Type: Capacity Scheduler'. The main area displays 'All Applications' with a table header:

ID	User	Name	Application Type	Application Tags	Queue	Application Priority	StartTime	LaunchTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU Vcores
----	------	------	------------------	------------------	-------	----------------------	-----------	------------	------------	-------	-------------	--------------------	----------------------

Below the table, a message says "Showing 0 to 0 of 0 entries".

Creation of “inputtxt” directory to the HDFS

The screenshot shows a Windows Command Prompt window titled "Administrator: Command Prompt" with the following commands and output:

```
C:\Python\Maptest>hdfs dfs -rm -r /inputtxt
Deleted /inputtxt

C:\Python\Maptest>hdfs dfs -mkdir /inputtxt

C:\Python\Maptest>hdfs dfs -put item1.txt item2.csv /inputtxt

C:\Python\Maptest>
```

Below the command prompt is a Notepad window displaying the following text:

Daniel Dadzie Appiah
ddappiah
156801227

The status bar at the bottom of the Notepad window shows "Ln 3, Col 10 | 39 characters | 120% | Window | UTF-8".

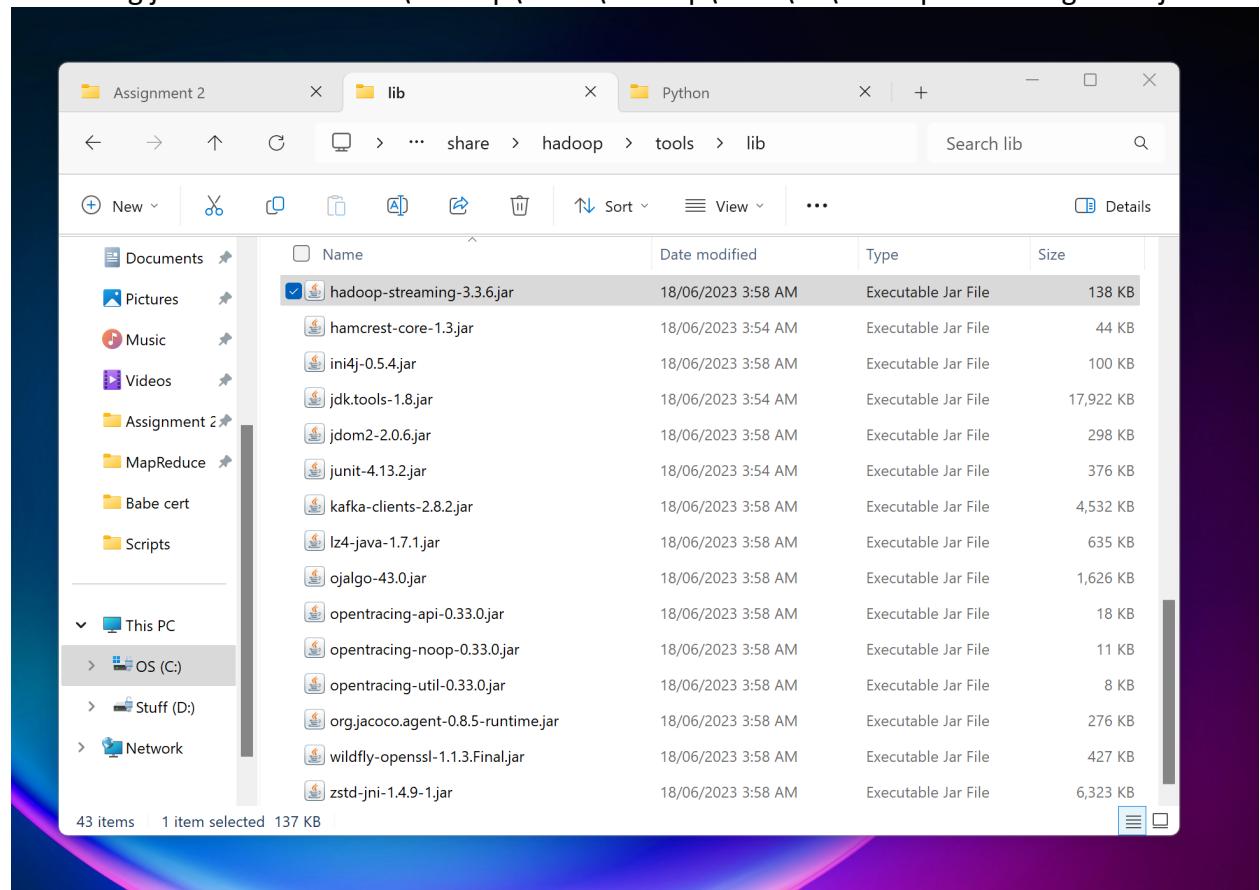
Validating uploaded files through Hadoop File System Using the Terminal

The screenshot shows a web-based Hadoop File System browser interface. At the top, there's a header bar with links for 'Hadoop', 'Overview', 'Datanodes', 'Datanode Volume Failures', 'Snapshot', 'Startup Progress', and 'Utilities'. Below the header is a search bar with the path '/inputtxt' and a 'Go!' button. A toolbar with icons for file operations like copy, move, delete, and refresh is visible. The main area is titled 'Browse Directory' and displays a table of file entries:

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	danie	supergroup	170 B	Mar 03 08:10	1	128 MB	item1.txt
-rw-r--r--	danie	supergroup	179 B	Mar 03 08:10	1	128 MB	item2.csv

Below the table, it says 'Showing 1 to 2 of 2 entries'. There are 'Previous' and 'Next' buttons at the bottom right. The footer of the page says 'Hadoop, 2023.'

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



Output on the Hadoop Cluster website [Running Application]

The screenshot shows the Hadoop Cluster website at localhost:8088/cluster/apps/RUNNING. The page title is "RUNNING A". On the left, there's a sidebar with a "Cluster Metrics" section containing links for "About", "Nodes", "Node Labels", "Applications", and "Scheduler". Below this is a "Tools" section. The main content area displays "Cluster Metrics" and "Cluster Nodes Metrics" tables. Under "Applications", a table lists one application: "application_1741042027500_0001" by user "danie". The application details include Name: "streamjob5271697595114463878.jar", Application Type: "MAPREDUCE", Queue: "default", Start Time: "Mon Mar 3 17:59:59 -0500 2025", Launch Time: "Mon Mar 3 18:00:01 -0500 2025", Finish Time: "N/A", and Status: "RUNN". At the bottom, it says "Showing 1 to 1 of 1 entries".

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

The screenshot shows the Hadoop Cluster website at localhost:8088/cluster/apps/RUNNING. The page title is "RUNNING Applications". The main content area displays "Used Resources", "Total Resources", "Reserved Resources", "Physical Mem Used %", and "Physical Vcores Used %". Below these are sections for "Nodes", "Allocation", and "Search". A table at the bottom lists application details: "LaunchTime": "Mon Mar 3 18:00:01 -0500 2025", "FinishTime": "N/A", "State": "RUNNING", "FinalStatus": "UNDEFINED", "Running Containers": "1", "Allocated CPU Vcores": "1", "Allocated Memory MB": "2048", "Allocated GPUs": "-1", "Reserved CPU Vcores": "0", "Reserved Memory MB": "0", "Reserved GPUs": "-1", "% of Queue": "25.0", "% of Cluster": "25.0", "Progress": "ApplicationMaster", "Tracking UI": "0". Navigation buttons "First", "Previous", "Next", and "Last" are at the bottom.

Output on Cluster when the jar streaming is Completed Running

The screenshot shows the Hadoop Cluster website at localhost:8088/cluster/apps/FINISHED. The page title is "FINISHED Ap". The main content area displays "Cluster Metrics" and "Cluster Nodes Metrics" tables. Under "Applications", a table lists one application: "application_1741042027500_0001" by user "danie". The application details are identical to the previous screenshots. At the bottom, it says "Showing 1 to 1 of 1 entries".

Second part of the Hadoop cluster page

The screenshot shows the 'FINISHED Applications' section of the Hadoop cluster page. It displays various metrics and a table of completed jobs.

Metrics:

- Used Resources: <memory:8 GB, vCores:8>
- Total Resources: <memory:8 GB, vCores:8>
- Reserved Resources: 54
- Physical Mem Used %: 12
- Decommissioned Nodes: 0
- Lost Nodes: 0
- Unhealthy Nodes: 0
- Rebooted Nodes: 0
- Shutdown: 0
- Minimum Allocation: <memory:1024, vCores:1>
- Maximum Allocation: <memory:8192, vCores:4>
- Maximum Cluster Application Priority: 0

Table:

Job ID	StartTime	LaunchTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU Vcores	Allocated Memory MB	Allocated GPUs	Reserved CPU Vcores	Reserved Memory MB	Reserved GPUs	% of Queue	% of Cluster	Progress	Time
Mon Mar 3 17:59:59 -0500 2025	Mon Mar 3 18:00:01 -0500 2025	Mon Mar 3 18:00:20 -0500 2025	Mon Mar 3 18:00:20 -0500 2025	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0	0.0	100%	His

Buttons: First, Previous

Zooming in on the mapreduce job

The screenshot shows the details of a specific mapreduce job.

Metrics:

- Decommissioned Nodes: 0
- Lost Nodes: 0
- Minimum Allocation: <memory:1024, vCores:1>
- Maximum Allocation: <memory:8192, vCores:4>

Table:

Job ID	Application Priority	StartTime	LaunchTime	FinishTime	State	FinalStatus	Running Containers
t	0	Mon Mar 3 17:59:59 -0500 2025	Mon Mar 3 18:00:01 -0500 2025	Mon Mar 3 18:00:20 -0500 2025	FINISHED	SUCCEEDED	N/A

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

The screenshot shows the 'Browse Directory' page of the Hadoop website.

Header:

- localhost:9870/explorer.html#/inputtxt
- Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities

Table:

File	Owner	Group	Size	Last Modified	Replication	Block Size	Name
item1.txt	danie	supergroup	170 B	Mar 03 08:10	1	128 MB	item1.txt
item2.csv	danie	supergroup	179 B	Mar 03 08:10	1	128 MB	item2.csv
output	danie	supergroup	0 B	Mar 03 18:05	0	0 B	output

Page Bottom:

- Showing 1 to 3 of 3 entries
- Previous 1 Next
- Hadoop, 2023.

Output for files inside “output”

The screenshot shows a web browser window with the URL `localhost:9870/explorer.html#/inputtxt/output`. The browser's address bar and toolbar are visible at the top. Below the toolbar, there is a green header bar with the word "Hadoop" and several navigation links: Overview, Datanodes, Datanode Volume Failures, Snapshot, Startup Progress, Utilities, and a dropdown menu.

Browse Directory

/inputtxt/output										<input type="button" value="Go!"/>						
Show 25 entries										Search: <input type="text"/>						
<input type="checkbox"/>		Permission		Owner		Group		Size		Last Modified		Replication		Block Size		Name
<input type="checkbox"/>		-rw-r--r--		danie		supergroup		0 B		Mar 03 18:05		1		128 MB		_SUCCESS
<input type="checkbox"/>		-rw-r--r--		danie		supergroup		292 B		Mar 03 18:05		1		128 MB		part-00000

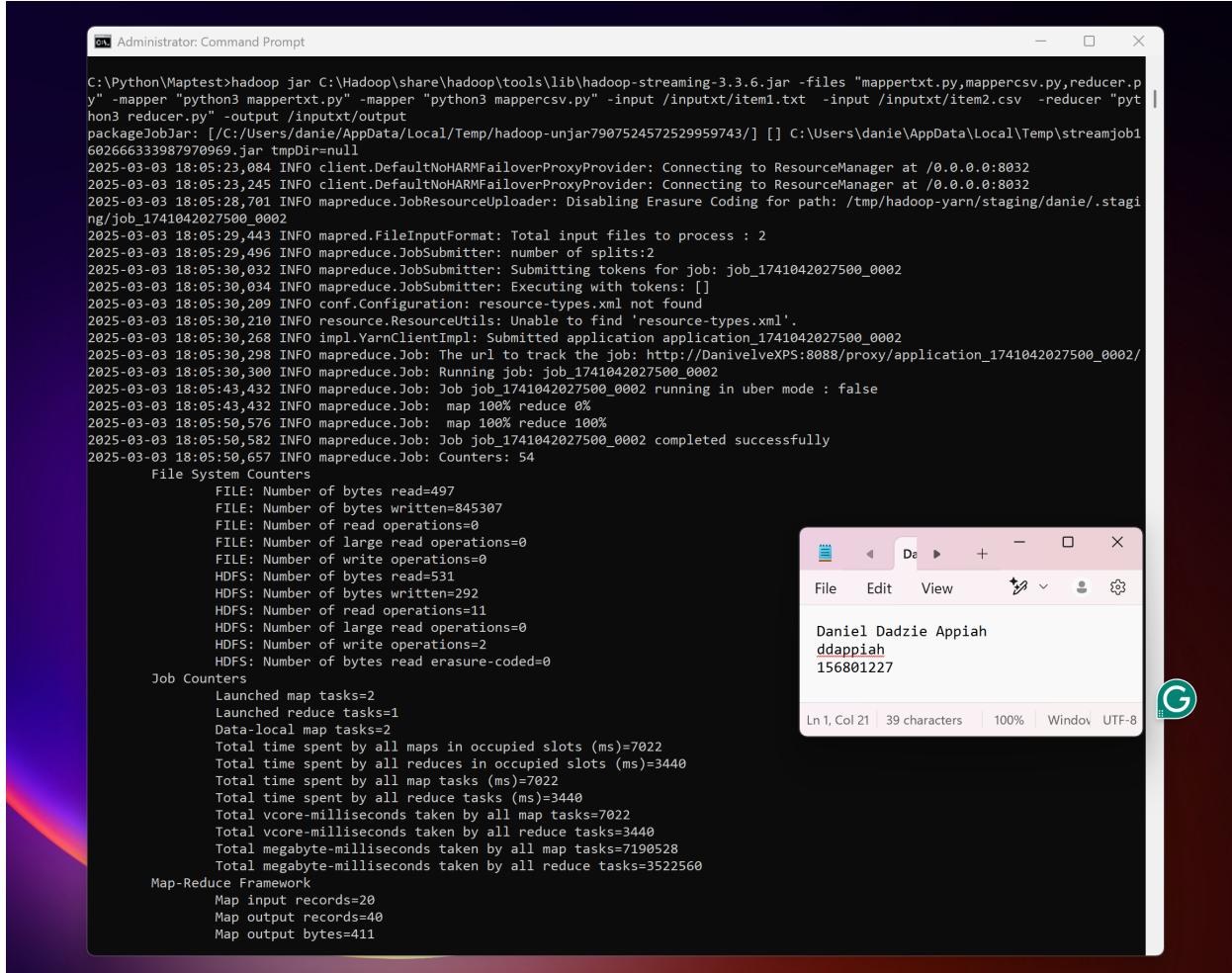
Showing 1 to 2 of 2 entries

Hadoop, 2023.

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 /inputtxt/item1.txt -input /inputtxt/item2.csv -reducer "python3  
reducer.py" -output /inputtxt/output
```

First part of the terminal



The screenshot shows two windows side-by-side. The left window is a Command Prompt titled 'Administrator: Command Prompt' with the following log output:

```
C:\Python\Maptest>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 /inputtxt/item1.txt -input /inputtxt/item2.csv -reducer "python3  
reducer.py" -output /inputtxt/output  
packageJobJar: [C:/Users/danie/AppData/Local/Temp/hadoop-unjar7907524572529959743/] [] C:/Users/danie/AppData/Local/Temp/streamjob1  
602666333987970969.jar tmpDir=null  
2025-03-03 18:05:23,084 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /0.0.0.0:8032  
2025-03-03 18:05:24,591 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /0.0.0.0:8032  
2025-03-03 18:05:28,701 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/danie/.staging/job_1741042027500_0002  
2025-03-03 18:05:29,443 INFO mapred.FileInputFormat: Total input files to process : 2  
2025-03-03 18:05:29,496 INFO mapreduce.JobSubmitter: number of splits:2  
2025-03-03 18:05:30,032 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1741042027500_0002  
2025-03-03 18:05:30,034 INFO mapreduce.JobSubmitter: Executing with tokens: []  
2025-03-03 18:05:30,209 INFO conf.Configuration: resource-types.xml not found  
2025-03-03 18:05:30,210 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.  
2025-03-03 18:05:30,268 INFO impl.YarnClientImpl: Submitted application application_1741042027500_0002  
2025-03-03 18:05:30,298 INFO mapreduce.Job: The url to track the job: http://DanivelveXPS:8088/proxy/application_1741042027500_0002/  
2025-03-03 18:05:30,300 INFO mapreduce.Job: Running job: job_1741042027500_0002  
2025-03-03 18:05:43,432 INFO mapreduce.Job: Job job_1741042027500_0002 running in uber mode : false  
2025-03-03 18:05:43,432 INFO mapreduce.Job: map 100% reduce 0%  
2025-03-03 18:05:50,576 INFO mapreduce.Job: map 100% reduce 100%  
2025-03-03 18:05:50,582 INFO mapreduce.Job: Job job_1741042027500_0002 completed successfully  
2025-03-03 18:05:50,657 INFO mapreduce.Job: Counters: 54  
File System Counters  
FILE: Number of bytes read=497  
FILE: Number of bytes written=845307  
FILE: Number of read operations=0  
FILE: Number of large read operations=0  
FILE: Number of write operations=0  
HDFS: Number of bytes read=531  
HDFS: Number of bytes written=292  
HDFS: Number of read operations=11  
HDFS: Number of large read operations=0  
HDFS: Number of write operations=2  
HDFS: Number of bytes read erasure-coded=0  
Job Counters  
Launched map tasks=2  
Launched reduce tasks=1  
Data-local map tasks=2  
Total time spent by all maps in occupied slots (ms)=7022  
Total time spent by all reduces in occupied slots (ms)=3440  
Total time spent by all map tasks (ms)=7022  
Total time spent by all reduce tasks (ms)=3440  
Total vcore-milliseconds taken by all map tasks=7022  
Total vcore-milliseconds taken by all reduce tasks=3440  
Total megabyte-milliseconds taken by all map tasks=7190528  
Total megabyte-milliseconds taken by all reduce tasks=3522560  
Map-Reduce Framework  
Map input records=20  
Map output records=40  
Map output bytes=411
```

The right window is a Notepad window with the following content:

```
Daniel Dadzie Appiah  
ddappiah  
156801227
```

Second part of the terminal

The image shows two terminal windows side-by-side. The left window is a Windows Command Prompt titled 'Administrator: Command Prompt'. It displays various HDFS and MapReduce counters. The right window is a Mac OS X terminal window showing user information.

Windows Command Prompt Content:

```
Administrator: Command Prompt
HDFS: Number of bytes read erasure-coded=0
Job Counters
    Launched map tasks=2
    Launched reduce tasks=1
    Data-local map tasks=2
    Total time spent by all maps in occupied slots (ms)=7022
    Total time spent by all reduces in occupied slots (ms)=3440
    Total time spent by all map tasks (ms)=7022
    Total time spent by all reduce tasks (ms)=3440
    Total vcore-milliseconds taken by all map tasks=7022
    Total vcore-milliseconds taken by all reduce tasks=3440
    Total megabyte-milliseconds taken by all map tasks=7190528
    Total megabyte-milliseconds taken by all reduce tasks=3522560
Map-Reduce Framework
    Map input records=20
    Map output records=40
    Map output bytes=411
    Map output materialized bytes=503
    Input split bytes=182
    Combine input records=0
    Combine output records=0
    Reduce input groups=27
    Reduce shuffle bytes=503
    Reduce input records=40
    Reduce output records=27
    Spilled Records=80
    Shuffled Maps =2
    Failed Shuffles=0
    Merged Map outputs=2
    GC time elapsed (ms)=239
    CPU time spent (ms)=1855
    Physical memory (bytes) snapshot=1269121024
    Virtual memory (bytes) snapshot=2252697600
    Total committed heap usage (bytes)=1333264384
    Peak Map Physical memory (bytes)=525307964
    Peak Map Virtual memory (bytes)=953806848
    Peak Reduce Physical memory (bytes)=315256832
    Peak Reduce Virtual memory (bytes)=648175616
Shuffle Errors
    BAD_ID=0
    CONNECTION=0
    IO_ERROR=0
    WRONG_LENGTH=0
    WRONG_MAP=0
    WRONG_REDUCE=0
File Input Format Counters
    Bytes Read=349
File Output Format Counters
    Bytes Written=292
2025-03-03 18:05:50,658 INFO streaming.StreamJob: Output directory: /inputtxt/output
C:\Python\Maptest>
```

Mac OS X Terminal Content:

```
Daniel Dadzie Appiah
ddappiah
156801227
Ln 1, Col 21 | 39 characters | 100% | Window | UTF-8
```

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.