S. B. JAIN INSTITUTE OF TECHNOLOGY, MANAGEMENT & RESEARCH, NAGPUR.

Practical No. 3

Aim: To Develop a MapReduce program to calculate the frequency of a word in a given file.

Name of Student: **Shrutika Pradeep Bagdi**

Roll No.: **CS22130**

Semester/Year: **7th/4th**

Academic Session: 2025 – 2026

Date of Performance: _____

Date of Submission: _____

AIM: To Develop a MapReduce program to calculate the frequency of a word in a given file.

OBJECTIVE/EXPECTED LEARNING OUTCOME:

The objectives and expected learning outcome of this practical are:

- Students will be able to facilitate concurrent processing by splitting petabytes of data into smaller chunks, and processing them in parallel on Hadoop commodity servers.
- Aggregates all the data from multiple servers to return a consolidated output back to the application.
- A programming model and an associated implementation for processing and generating big data sets with a parallel, distributed algorithm on a cluster.

HARDWARE AND SOFTWARE REQUIRMENTS:

Hardware Requirement: High Configuration computer

Software Requirement: Hadoop-3.3.6, jdk1.8, notepad++**.**

THEORY:

MapReduce is a parallel programming model for writing distributed applications devised at Google for efficient processing of large amounts of data (multi-terabyte data-sets), on large clusters (thousands of nodes) of commodity hardware in a reliable, fault-tolerant manner. The MapReduce program runs on Hadoop which is an Apache open-source framework

WordCount is a simple program which counts the number of occurrences of each word in a given-text input data set. WordCount fits very well with the MapReduce programming model making it a great example to understand the Hadoop Map/Reduce programming style. Our implementation consists of two main parts:

1. Mapper

2. Reducer

MapReduce consists of 2 steps:

Map Function – It takes a set of data and converts it into another set of data, where individual elements are broken down into tuples (Key-Value pair).

Input (Set of Data) : Bus, Car, bus, car, train, car, bus, car, train, bus, TRAIN,BUS, buS, BUS, TRAIN

Output (Convert into another set of data) (Key,Value) :

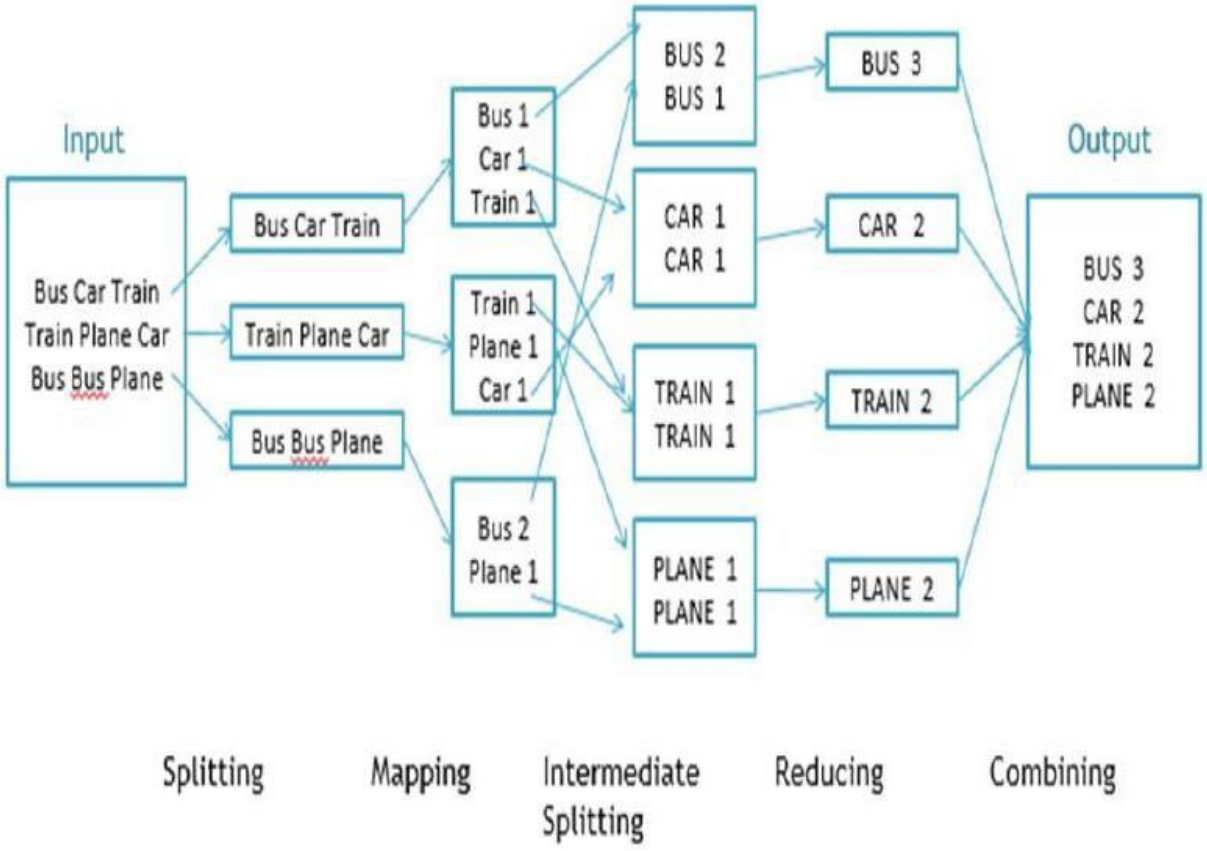
(Bus,1), (Car,1), (bus,1), (car,1), (train,1), (car,1), (bus,1), (car,1), (train,1), (bus,1), (TRAIN,1),(BUS,1), (buS,1), (caR,1), (CAR,1), (car,1), (BUS,1), (TRAIN,1)

Reduce Function – Takes the output from Map as an input and combines those data tuples into a smaller set of tuples.

Input (Set of tuples): (Bus,1), (Car,1), (bus,1), (car,1), (car,1), (bus,1), (car,1), (train,1),

(TRAIN,1),(BUS,1), (buS,1), (caR,1) (car,1), (BUS,1), (TRAIN,1)

Output (Converts into smaller set of tuples) (BUS,7), (CAR,7), (TRAIN,4)



Workflow of MapReduce consists of 5 steps:

1. Splitting – The splitting parameter can be anything, e.g. splitting by space, comma, semicolon, or even by a new line ('\n').

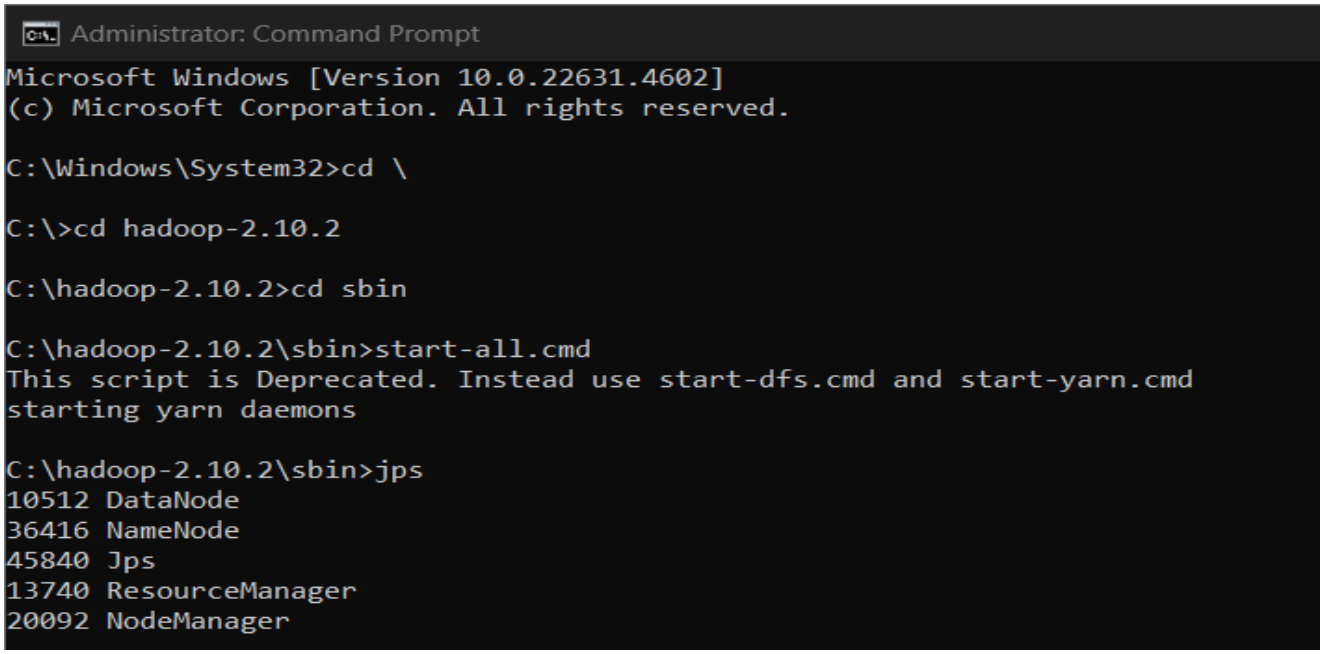
2. Mapping – as explained above.

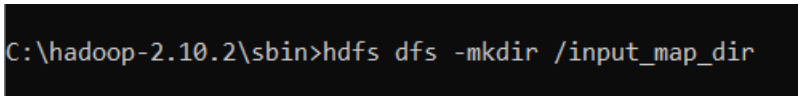
3. Intermediate splitting – the entire process in parallel on different clusters. In order to group them in "Reduce Phase" the similar KEY data should be on the same cluster.

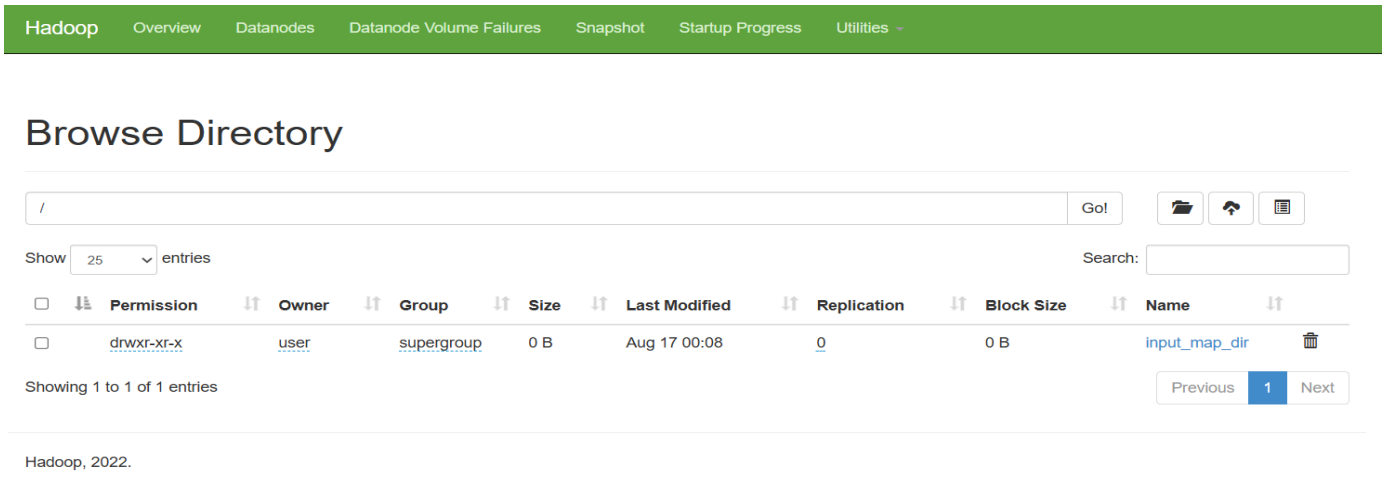
4. Reduce – it is nothing but mostly group by phase.

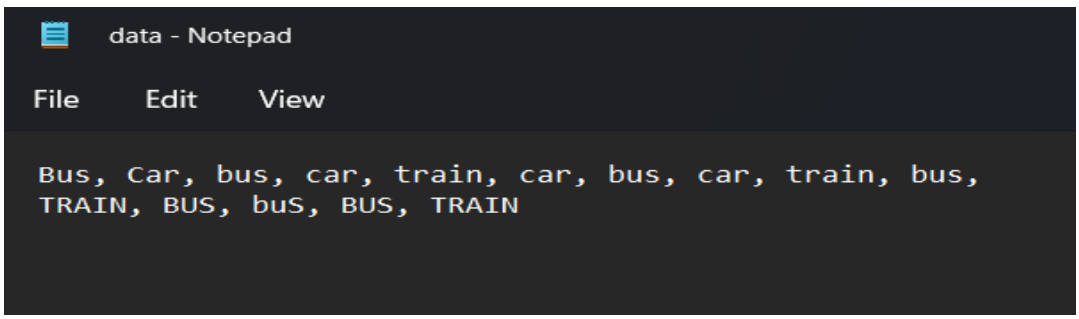
5. Combining – The last phase where all the data (individual result set from each cluster) is combined together to form a result.

INPUT / OUTPUT (SCREENSHOTS):

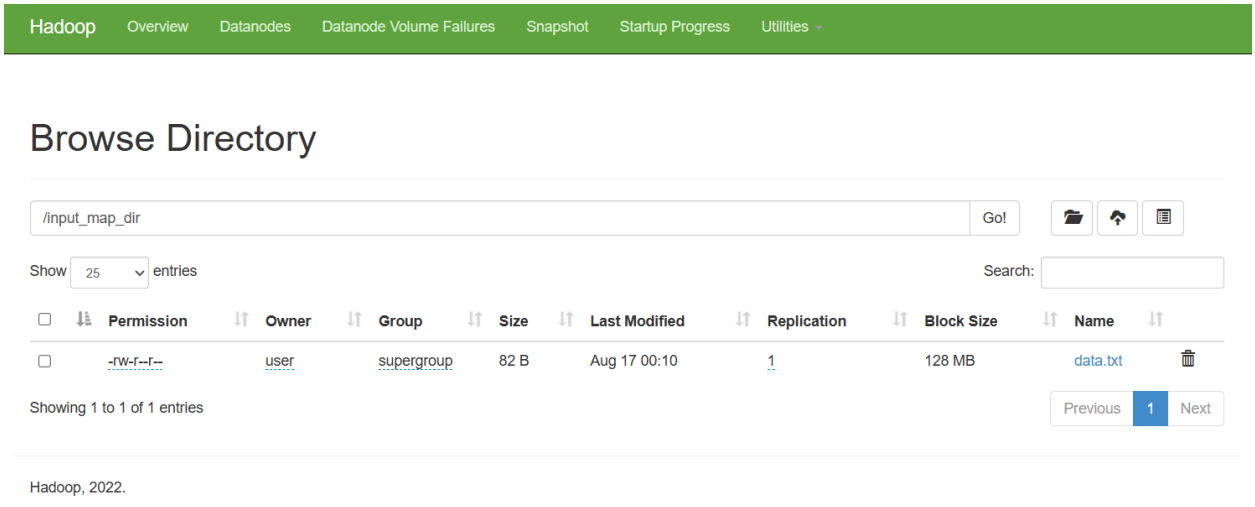








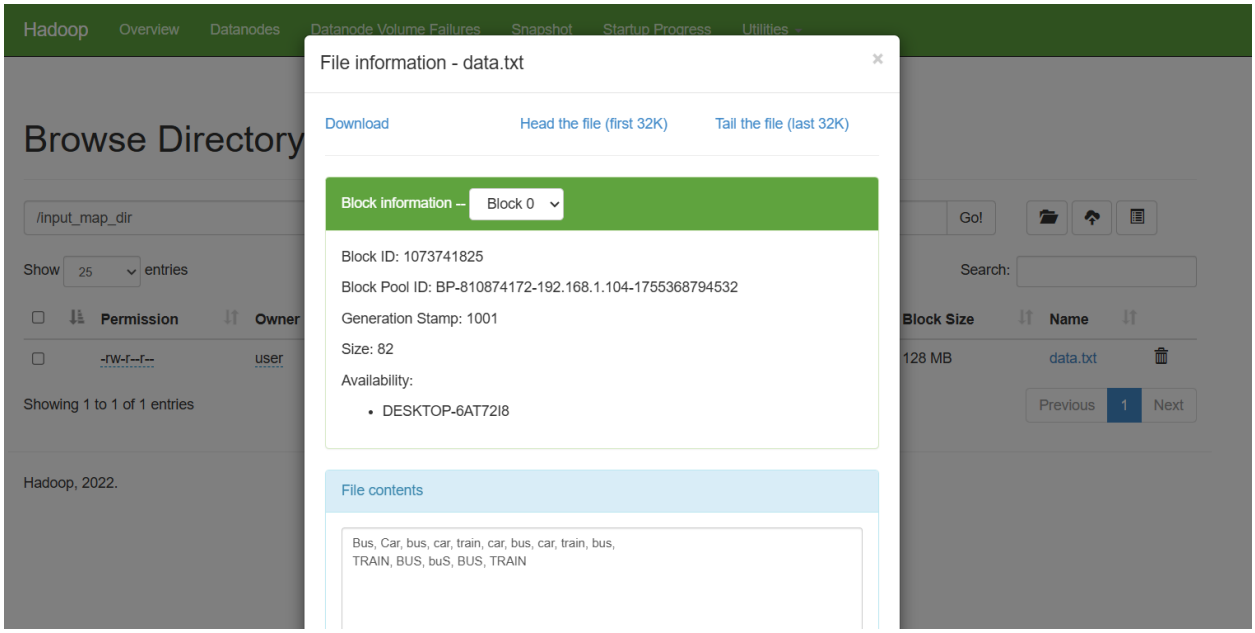
```
C:\hadoop-2.10.2\sbin>hdfs dfs -put D:\data.txt /input_map_dir
```




```
C:\hadoop-2.10.2\sbin>hadoop fs -ls /input_map_dir
Found 1 items
-rw-r--r--   1 user supergroup         82 2025-08-17 00:10 /input_map_dir/data.txt

C:\hadoop-2.10.2\sbin>hadoop fs -chmod 777 /input_map_dir/data.txt

C:\hadoop-2.10.2\sbin>hadoop fs -ls /input_map_dir
Found 1 items
-rwxrwxrwx   1 user supergroup         82 2025-08-17 00:10 /input_map_dir/data.txt
```



Select Administrator: Command Prompt

```
C:\hadoop-2.10.2\sbin>hadoop jar C:/hadoop-2.10.2/share/hadoop/mapreduce/hadoop-mapreduce-examples-2.10.
2.jar wordcount /input_map_dir /output_dirnew1
25/08/17 00:17:30 INFO Configuration.deprecation: session.id is deprecated. Instead, use dfs.metrics.ses
sion-id
25/08/17 00:17:30 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker, sessionId=
25/08/17 00:17:31 INFO input.FileInputFormat: Total input files to process : 1
25/08/17 00:17:31 INFO mapreduce.JobSubmitter: number of splits:1
25/08/17 00:17:31 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local512420856_0001
25/08/17 00:17:31 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
25/08/17 00:17:31 INFO mapreduce.Job: Running job: job_local512420856_0001
25/08/17 00:17:31 INFO mapred.LocalJobRunner: OutputCommitter set in config null
25/08/17 00:17:31 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
25/08/17 00:17:31 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders u
nder output directory:false, ignore cleanup failures: false
25/08/17 00:17:31 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.
FileOutputCommitter
25/08/17 00:17:31 INFO mapred.LocalJobRunner: Waiting for map tasks
25/08/17 00:17:31 INFO mapred.LocalJobRunner: Starting task: attempt_local512420856_0001_m_000000_0
25/08/17 00:17:31 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
25/08/17 00:17:31 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders u
nder output directory:false, ignore cleanup failures: false
25/08/17 00:17:31 INFO util.ProcfsBasedProcessTree: ProcfsBasedProcessTree currently is supported only o
n Linux.
25/08/17 00:17:31 INFO mapred.Task:  Using ResourceCalculatorProcessTree : org.apache.hadoop.yarn.util.W
indowsBasedProcessTree@7e4f9ed7
25/08/17 00:17:31 INFO mapred.MapTask: Processing split: hdfs://localhost:9000/input_map_dir/data.txt:0+
82
```

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities

Browse Directory

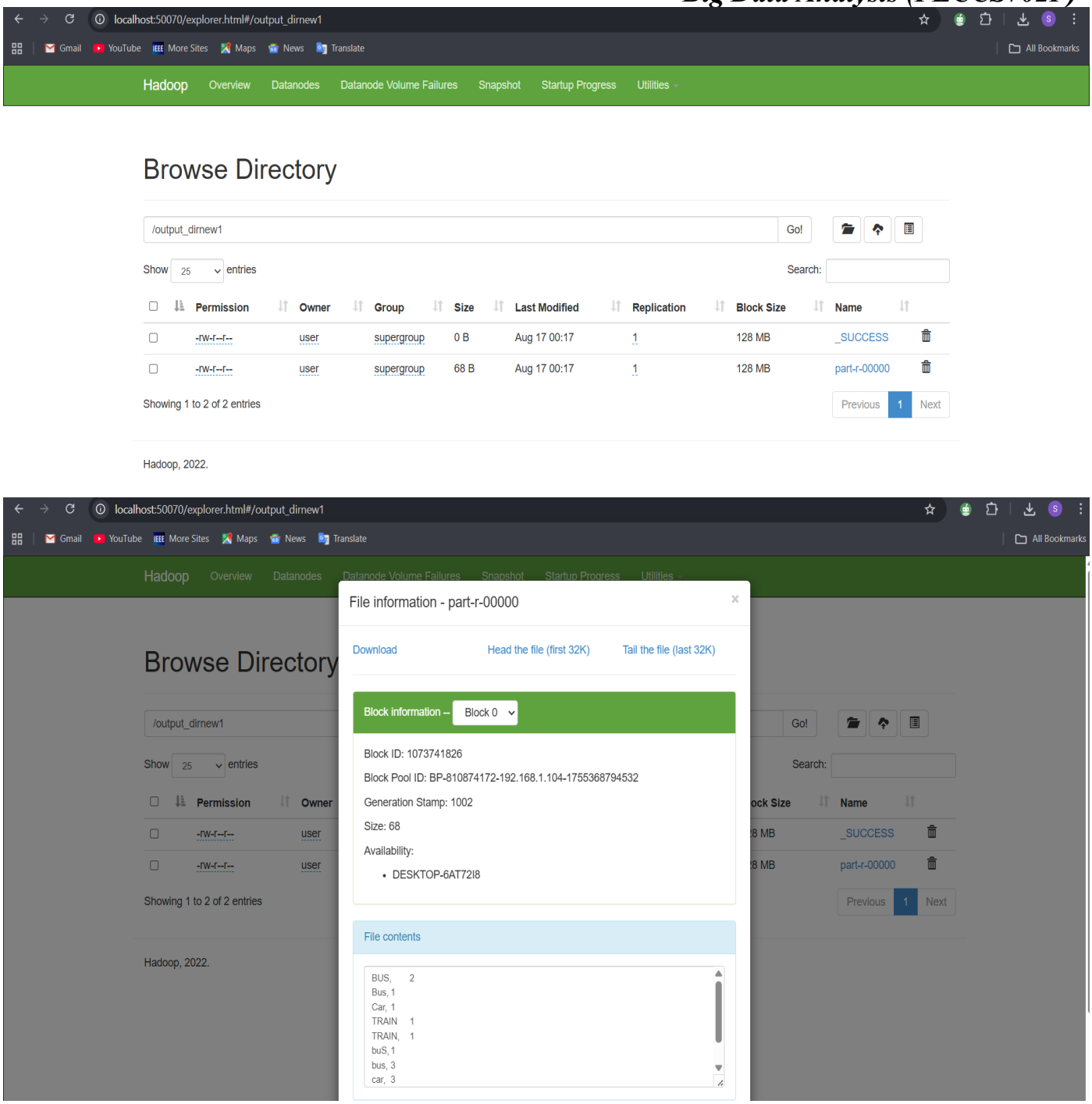
/ Go!

Show 25 entries Search:

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxr-xr-x	user	supergroup	0 B	Aug 17 00:10	0	0 B	input_map_dir
drwxr-xr-x	user	supergroup	0 B	Aug 17 00:17	0	0 B	output_dirnew1

Showing 1 to 2 of 2 entries Previous 1 Next

Hadoop, 2022.



CONCLUSION:

DISCUSSION AND VIVA VOCE:

- Explain the key components of a MapReduce program.
- What is the role of the Reducer in a Word Count program?

- What is shuffling and sorting in the context of MapReduce?
- What is combiner function, and how does it improve the Word Count program's efficiency?
- How would you optimize a Word Count program for large datasets or different file formats?

REFERENCE:

- https://www.ibm.com/topics/mapreduce
- https://www.google.com/search?q=objectives+and+expected+learning+outcome+of+mapreduce+program&sca_esv=565257361&ei=46wCZZW-NayV4-AcsCqgEDMy0xuAEByAEA-AEC-AEB4gMEGAAgQYgGAQ&sclient=gws-wiz-serp.

Observation book: (3)	Viva-Voce (3)	Quality of Submission and timely Evaluation (4)
Total:		**Sign with date:**