School of Science and Engineering

Project N°1

Intro to Big Data Environment

Fall 2023

Submitted By :

Asmaa Dalil (75675)

Supervised by :

Dr. Tajjeeddine Rachidi

Table of Contents:

I. Introduction

II. Hadoop Architecture Used

1. Hadoop Architecture:

Using clusters of affordable technology, Hadoop is an open-source platform for the distributed archiving and analysis of massive datasets. Its design is made up of a number of essential parts that cooperate to offer fault tolerance, scalability, and effective data processing. Here is a list of the key elements of the Hadoop architecture.

- Hadoop HDFS to store data across slave machines.
- Hadoop YARN for resource management in the Hadoop cluster.
- Hadoop MapReduce to process data in a distributed fashion.
- Zookeeper to ensure synchronization across a cluster.

Under the scope of this project, we are going to use Hadoop Distributed File System (HDFS) and MapReduce model.

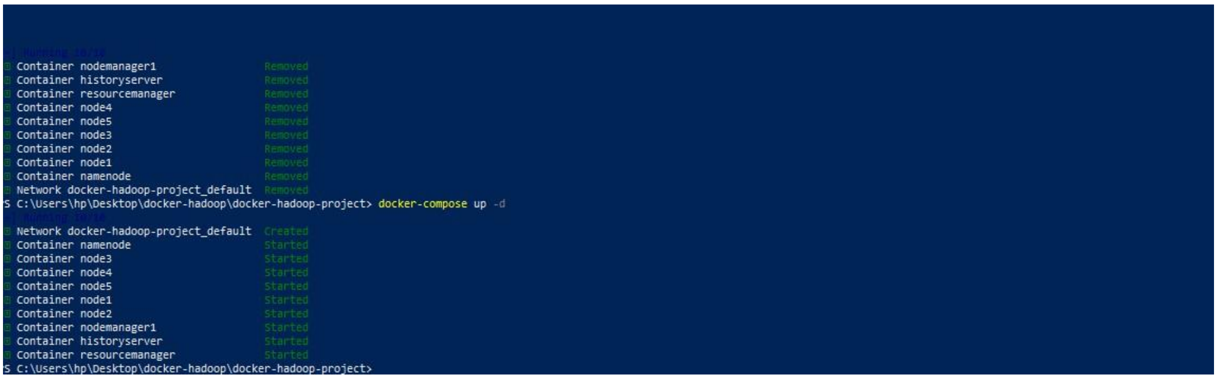
2. Docker Container

3. Hadoop Cluster Setup

Now , we are going to deploy Hadoop cluster by cloning from GitHub link that includes the docker-hadoop folder in order to install the Hadoop cluster with 5 namenodes.

After we get into the folder, we use the command:

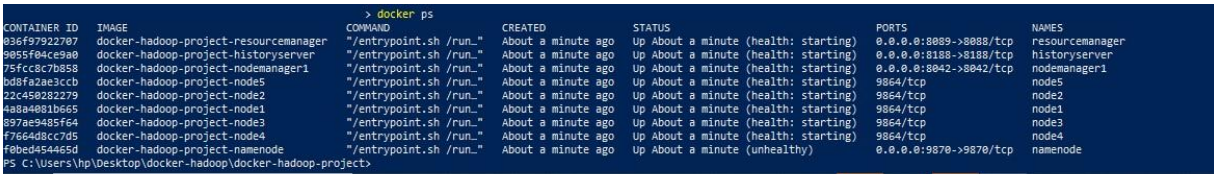
```
$ docker-compose up -d
```



```
+] Running 10/10
 Container nodemanager1              Removed
 Container historyserver             Removed
 Container resourcemanager           Removed
 Container node4                     Removed
 Container node5                     Removed
 Container node3                     Removed
 Container node2                     Removed
 Container node1                     Removed
 Container namenode                  Removed
 Network docker-hadoop-project_default  Removed
S C:\Users\hp\Desktop\docker-hadoop\docker-hadoop-project> docker-compose up -d
+] Running 10/10
 Network docker-hadoop-project_default  Created
 Container namenode                  Started
 Container node3                     Started
 Container node4                     Started
 Container node5                     Started
 Container node1                     Started
 Container node2                     Started
 Container nodemanager1              Started
 Container historyserver             Started
 Container resourcemanager           Started
S C:\Users\hp\Desktop\docker-hadoop\docker-hadoop-project>
```

Figure: creation of containers in docker

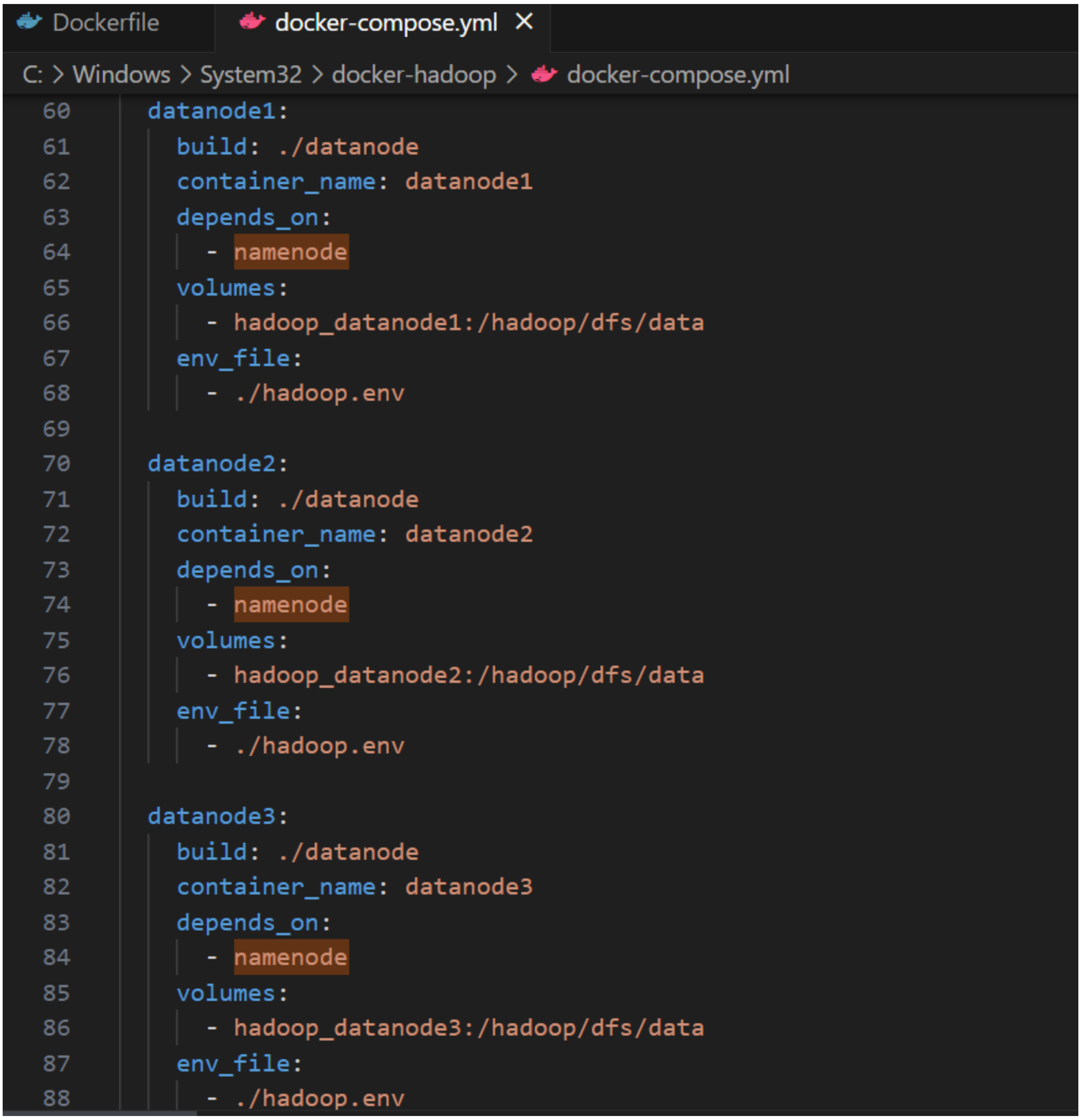
Then we use the command *docker ps* in order to pull the running containers



```
                                                          > docker ps
CONTAINER ID   IMAGE                                  COMMAND                  CREATED              STATUS                                 PORTS                      NAMES
036f97922707   docker-hadoop-project-resourcemanager   "/entrypoint.sh /run…"   About a minute ago   Up About a minute (health: starting)   0.0.0.0:8089->8088/tcp   resourcemanager
9055f04ce9a0   docker-hadoop-project-historyserver     "/entrypoint.sh /run…"   About a minute ago   Up About a minute (health: starting)   0.0.0.0:8188->8188/tcp   historyserver
75fcc8c7b858   docker-hadoop-project-nodemanager1      "/entrypoint.sh /run…"   About a minute ago   Up About a minute (health: starting)   0.0.0.0:8042->8042/tcp   nodemanager1
bd8fa2ae3ccb   docker-hadoop-project-node5             "/entrypoint.sh /run…"   About a minute ago   Up About a minute (health: starting)   9864/tcp                 node5
22c450282279   docker-hadoop-project-node2             "/entrypoint.sh /run…"   About a minute ago   Up About a minute (health: starting)   9864/tcp                 node2
4a8a4081b665   docker-hadoop-project-node1             "/entrypoint.sh /run…"   About a minute ago   Up About a minute (health: starting)   9864/tcp                 node1
897ae9485f64   docker-hadoop-project-node3             "/entrypoint.sh /run…"   About a minute ago   Up About a minute (health: starting)   9864/tcp                 node3
f7664d8cc7d5   docker-hadoop-project-node4             "/entrypoint.sh /run…"   About a minute ago   Up About a minute (health: starting)   9864/tcp                 node4
f0bed454465d   docker-hadoop-project-namenode          "/entrypoint.sh /run…"   About a minute ago   Up About a minute (unhealthy)          0.0.0.0:9870->9870/tcp   namenode
PS C:\Users\hp\Desktop\docker-hadoop\docker-hadoop-project>
```

Figure: the running containers

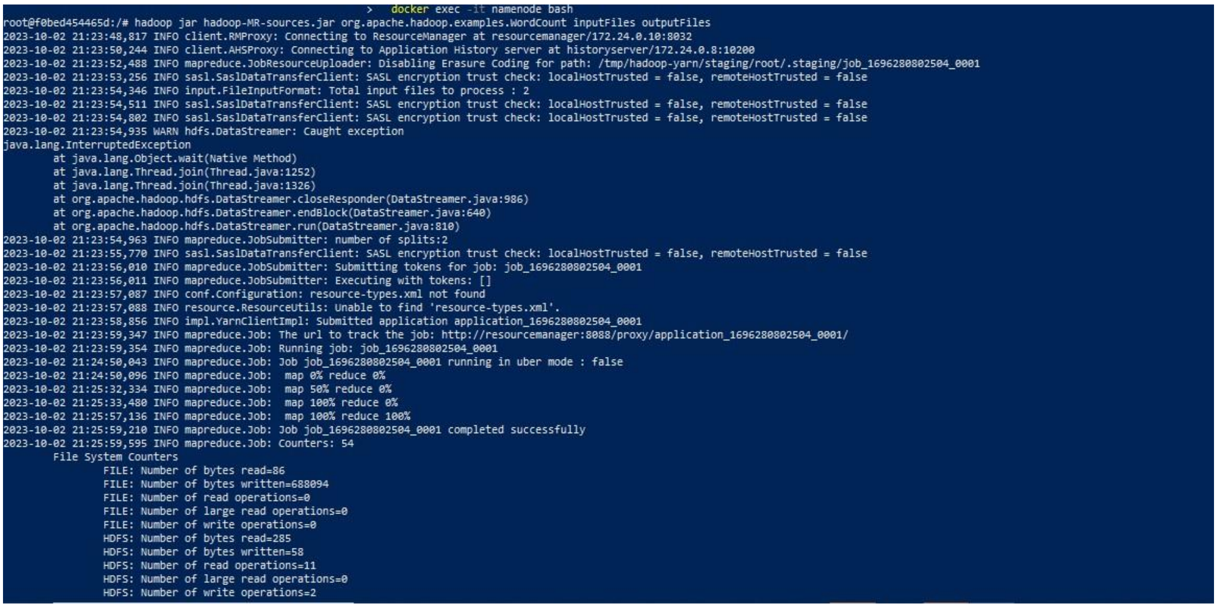
To get into the namenode container in order to test the 2 simple files as showing in the figure. This figure shows that we tested the wordcount program on the files and we got the results.


Dockerfile
docker-compose.yml ×
C: > Windows > System32 > docker-hadoop > docker-compose.yml

```
  datanode1:
    build: ./datanode
    container_name: datanode1
    depends_on:
      - namenode
    volumes:
      - hadoop_datanode1:/hadoop/dfs/data
    env_file:
      - ./hadoop.env

  datanode2:
    build: ./datanode
    container_name: datanode2
    depends_on:
      - namenode
    volumes:
      - hadoop_datanode2:/hadoop/dfs/data
    env_file:
      - ./hadoop.env

  datanode3:
    build: ./datanode
    container_name: datanode3
    depends_on:
      - namenode
    volumes:
      - hadoop_datanode3:/hadoop/dfs/data
    env_file:
      - ./hadoop.env
```

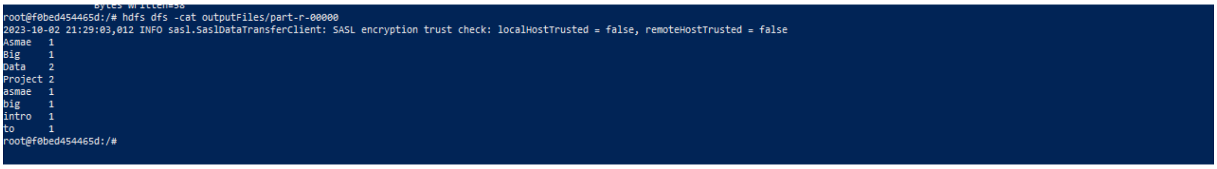
Figure: Compose.yml

we will create a directory with the same name in HDFS and copy all of its there. They are now spread over our 5 datanodes in the HDFS:



```
                                              >  docker exec -it namenode bash
root@f0bed454465d:/# hadoop jar hadoop-MR-sources.jar org.apache.hadoop.examples.WordCount inputFiles outputFiles
2023-10-02 21:23:48,817 INFO client.RMProxy: Connecting to ResourceManager at resourcemanager/172.24.0.10:8032
2023-10-02 21:23:50,244 INFO client.AHSProxy: Connecting to Application History server at historyserver/172.24.0.8:10200
2023-10-02 21:23:52,488 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/root/.staging/job_1696280802504_0001
2023-10-02 21:23:53,256 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
2023-10-02 21:23:54,346 INFO input.FileInputFormat: Total input files to process : 2
2023-10-02 21:23:54,511 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
2023-10-02 21:23:54,802 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
2023-10-02 21:23:54,935 WARN hdfs.DataStreamer: Caught exception
java.lang.InterruptedException
        at java.lang.Object.wait(Native Method)
        at java.lang.Thread.join(Thread.java:1252)
        at java.lang.Thread.join(Thread.java:1326)
        at org.apache.hadoop.hdfs.DataStreamer.closeResponder(DataStreamer.java:986)
        at org.apache.hadoop.hdfs.DataStreamer.endBlock(DataStreamer.java:640)
        at org.apache.hadoop.hdfs.DataStreamer.run(DataStreamer.java:810)
2023-10-02 21:23:54,963 INFO mapreduce.JobSubmitter: number of splits:2
2023-10-02 21:23:55,770 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
2023-10-02 21:23:56,010 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1696280802504_0001
2023-10-02 21:23:56,011 INFO mapreduce.JobSubmitter: Executing with tokens: []
2023-10-02 21:23:57,087 INFO conf.Configuration: resource-types.xml not found
2023-10-02 21:23:57,088 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2023-10-02 21:23:58,856 INFO impl.YarnClientImpl: Submitted application application_1696280802504_0001
2023-10-02 21:23:59,347 INFO mapreduce.Job: The url to track the job: http://resourcemanager:8088/proxy/application_1696280802504_0001/
2023-10-02 21:23:59,354 INFO mapreduce.Job: Running job: job_1696280802504_0001
2023-10-02 21:24:50,043 INFO mapreduce.Job: Job job_1696280802504_0001 running in uber mode : false
2023-10-02 21:24:50,096 INFO mapreduce.Job:  map 0% reduce 0%
2023-10-02 21:25:32,334 INFO mapreduce.Job:  map 50% reduce 0%
2023-10-02 21:25:33,480 INFO mapreduce.Job:  map 100% reduce 0%
2023-10-02 21:25:57,136 INFO mapreduce.Job:  map 100% reduce 100%
2023-10-02 21:25:59,210 INFO mapreduce.Job: Job job_1696280802504_0001 completed successfully
2023-10-02 21:25:59,595 INFO mapreduce.Job: Counters: 54
        File System Counters
                FILE: Number of bytes read=86
                FILE: Number of bytes written=688094
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=285
                HDFS: Number of bytes written=58
                HDFS: Number of read operations=11
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
```

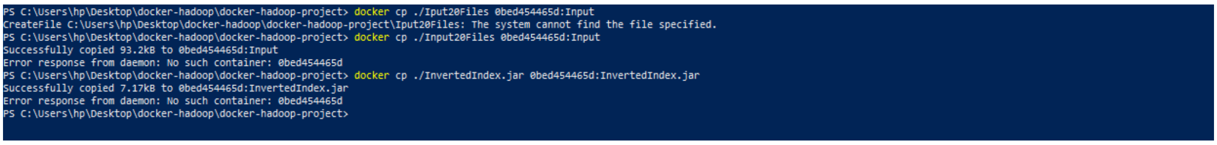
Figure: wordcount on simple files



```
                Bytes Written=58
root@f0bed454465d:/# hdfs dfs -cat outputFiles/part-r-00000
2023-10-02 21:29:03,012 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
Asmae   1
Big     1
Data    2
Project 2
asmae   1
big     1
intro   1
to      1
root@f0bed454465d:/#
```

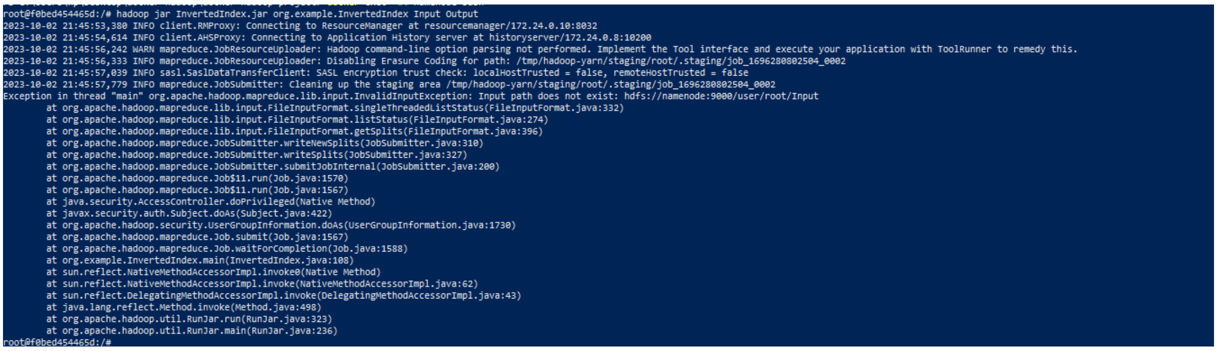
Figure: results

Inverted Index



```
PS C:\Users\hp\Desktop\docker-hadoop\docker-hadoop-project> docker cp ./Iput20Files 0bed454465d:Input
CreateFile C:\Users\hp\Desktop\docker-hadoop\docker-hadoop-project\Iput20Files: The system cannot find the file specified.
PS C:\Users\hp\Desktop\docker-hadoop\docker-hadoop-project> docker cp ./Input20Files 0bed454465d:Input
Successfully copied 93.2kB to 0bed454465d:Input
Error response from daemon: No such container: 0bed454465d
PS C:\Users\hp\Desktop\docker-hadoop\docker-hadoop-project> docker cp ./InvertedIndex.jar 0bed454465d:InvertedIndex.jar
Successfully copied 7.17kB to 0bed454465d:InvertedIndex.jar
Error response from daemon: No such container: 0bed454465d
PS C:\Users\hp\Desktop\docker-hadoop\docker-hadoop-project>
```

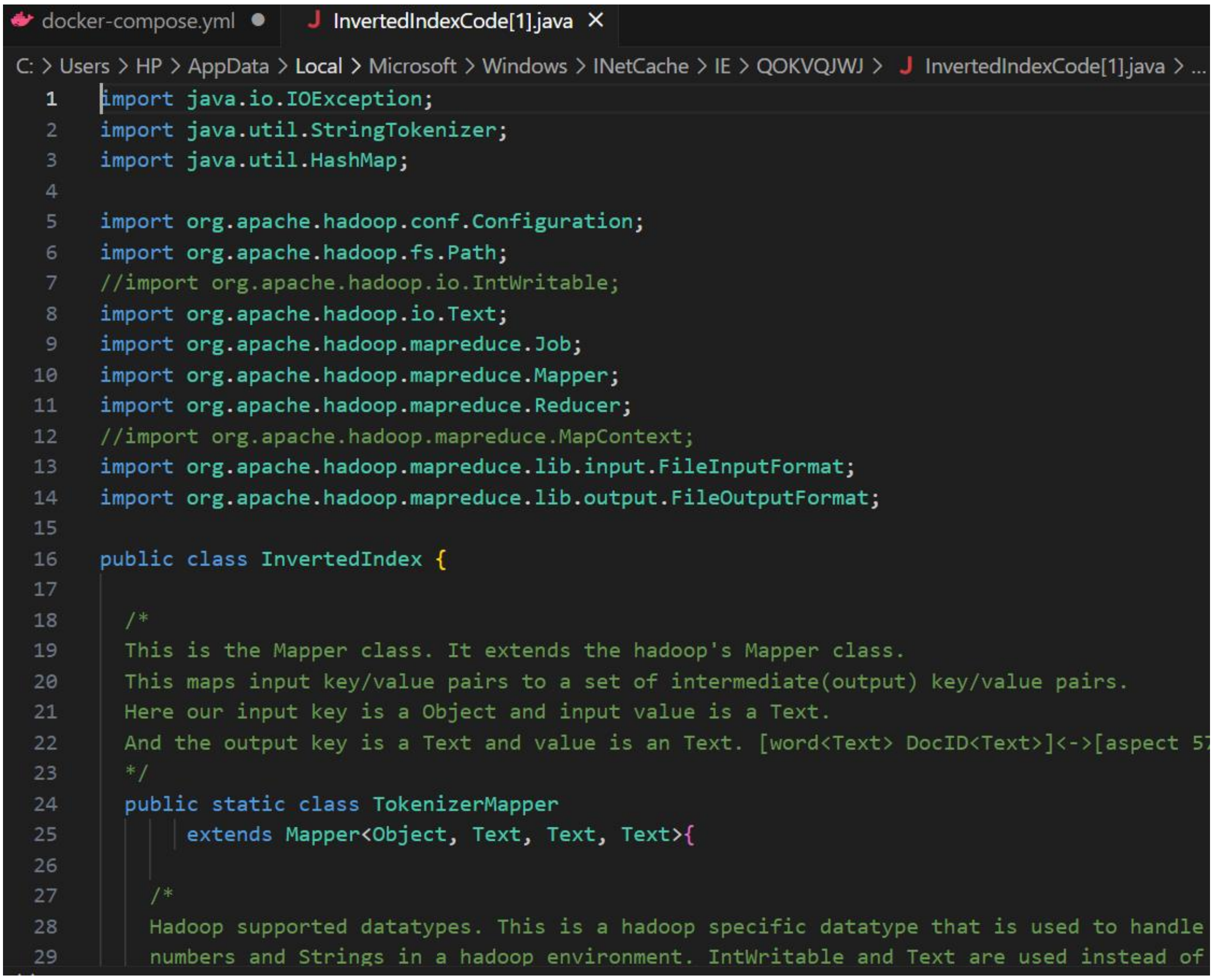
Figure : copying the 20 files of 1000 words to run inverted index



```
root@f0bed454465d:/# hadoop jar InvertedIndex.jar org.example.InvertedIndex Input Output
2023-10-02 21:45:53,380 INFO client.RMProxy: Connecting to ResourceManager at resourcemanager/172.24.0.10:8032
2023-10-02 21:45:54,614 INFO client.AHSProxy: Connecting to Application History server at historyserver/172.24.0.8:10200
2023-10-02 21:45:56,242 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
2023-10-02 21:45:56,333 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/root/.staging/job_1696280802504_0002
2023-10-02 21:45:57,039 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
2023-10-02 21:45:57,779 INFO mapreduce.JobSubmitter: Cleaning up the staging area /tmp/hadoop-yarn/staging/root/.staging/job_1696280802504_0002
Exception in thread "main" org.apache.hadoop.mapreduce.lib.input.InvalidInputException: Input path does not exist: hdfs://namenode:9000/user/root/Input
        at org.apache.hadoop.mapreduce.lib.input.FileInputFormat.singleThreadedListStatus(FileInputFormat.java:332)
        at org.apache.hadoop.mapreduce.lib.input.FileInputFormat.listStatus(FileInputFormat.java:274)
        at org.apache.hadoop.mapreduce.lib.input.FileInputFormat.getSplits(FileInputFormat.java:396)
        at org.apache.hadoop.mapreduce.JobSubmitter.writeNewSplits(JobSubmitter.java:310)
        at org.apache.hadoop.mapreduce.JobSubmitter.writeSplits(JobSubmitter.java:327)
        at org.apache.hadoop.mapreduce.JobSubmitter.submitJobInternal(JobSubmitter.java:200)
        at org.apache.hadoop.mapreduce.Job$11.run(Job.java:1570)
        at org.apache.hadoop.mapreduce.Job$11.run(Job.java:1567)
        at java.security.AccessController.doPrivileged(Native Method)
        at javax.security.auth.Subject.doAs(Subject.java:422)
        at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1730)
        at org.apache.hadoop.mapreduce.Job.submit(Job.java:1567)
        at org.apache.hadoop.mapreduce.Job.waitForCompletion(Job.java:1588)
        at org.example.InvertedIndex.main(InvertedIndex.java:108)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:498)
        at org.apache.hadoop.util.RunJar.run(RunJar.java:323)
        at org.apache.hadoop.util.RunJar.main(RunJar.java:236)
root@f0bed454465d:/#
```

Figure: running inverted index

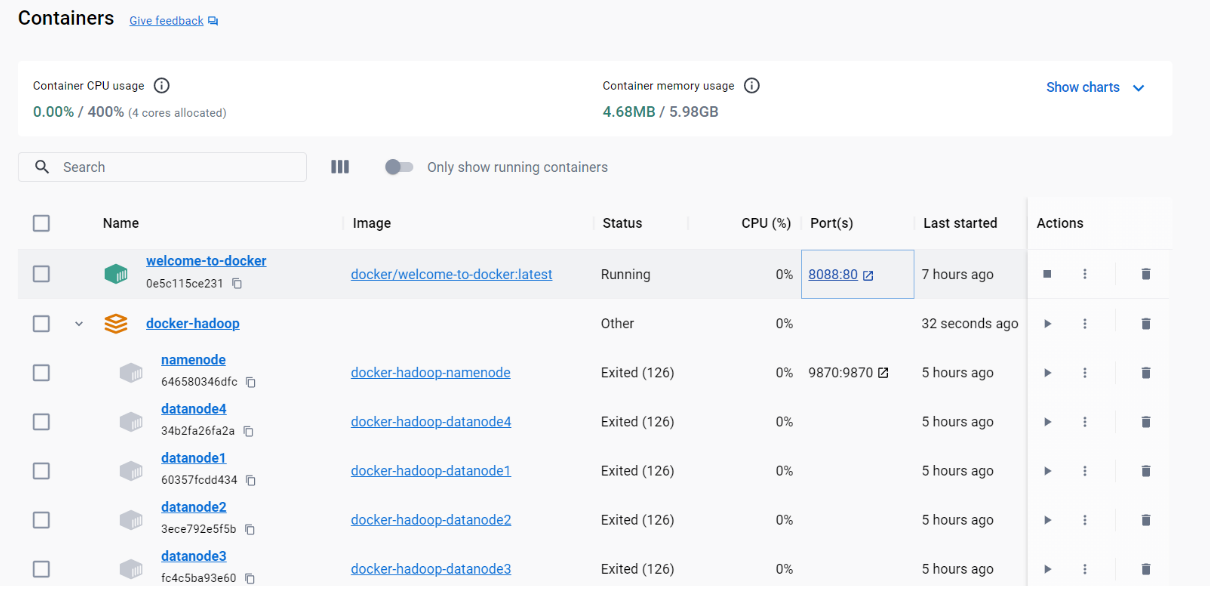
Source code:


docker-compose.yml
InvertedIndexCode[1].java
C: > Users > HP > AppData > Local > Microsoft > Windows > INetCache > IE > QOKVQJWJ > InvertedIndexCode[1].java > ...

```java
import java.io.IOException;
import java.util.StringTokenizer;
import java.util.HashMap;

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
//import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
//import org.apache.hadoop.mapreduce.MapContext;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

public class InvertedIndex {

  /*
  This is the Mapper class. It extends the hadoop's Mapper class.
  This maps input key/value pairs to a set of intermediate(output) key/value pairs.
  Here our input key is a Object and input value is a Text.
  And the output key is a Text and value is an Text. [word<Text> DocID<Text>]<->[aspect 57
  */
  public static class TokenizerMapper
      extends Mapper<Object, Text, Text, Text>{

    /*
    Hadoop supported datatypes. This is a hadoop specific datatype that is used to handle
    numbers and Strings in a hadoop environment. IntWritable and Text are used instead of
```

Dificulties:

I had a problem running the containers on powershell so I had to use Docket ToolBox to run them



Container CPU usage 0.00% / 400% (4 cores allocated)

Container memory usage 4.68MB / 5.98GB

□	Name	Image	Status	CPU (%)	Port(s)	Last started	Actions
□	welcome-to-docker 0e5c115ce231	docker/welcome-to-docker:latest	Running	0%	8088:80	7 hours ago	■ ⋮
□	docker-hadoop		Other	0%		32 seconds ago	▶ ⋮
□	namenode 646580346dfc	docker-hadoop-namenode	Exited (126)	0%	9870:9870	5 hours ago	▶ ⋮
□	datanode4 34b2fa26fa2a	docker-hadoop-datanode4	Exited (126)	0%		5 hours ago	▶ ⋮
□	datanode1 60357fcdd434	docker-hadoop-datanode1	Exited (126)	0%		5 hours ago	▶ ⋮
□	datanode2 3ece792e5f5b	docker-hadoop-datanode2	Exited (126)	0%		5 hours ago	▶ ⋮
□	datanode3 fc4c5ba93e60	docker-hadoop-datanode3	Exited (126)	0%		5 hours ago	▶ ⋮

References:

https://github.com/big-data-europe/docker-hadoop#supported-hadoop-versions

https://chat.openai.com/

https://www.simplilearn.com/tutorials/hadoop-tutorial/hadoop-architecture#:~:text=Hadoop%20is%20a%20framework%20permitting,management%20in%20the%20Hadoop%20cluster