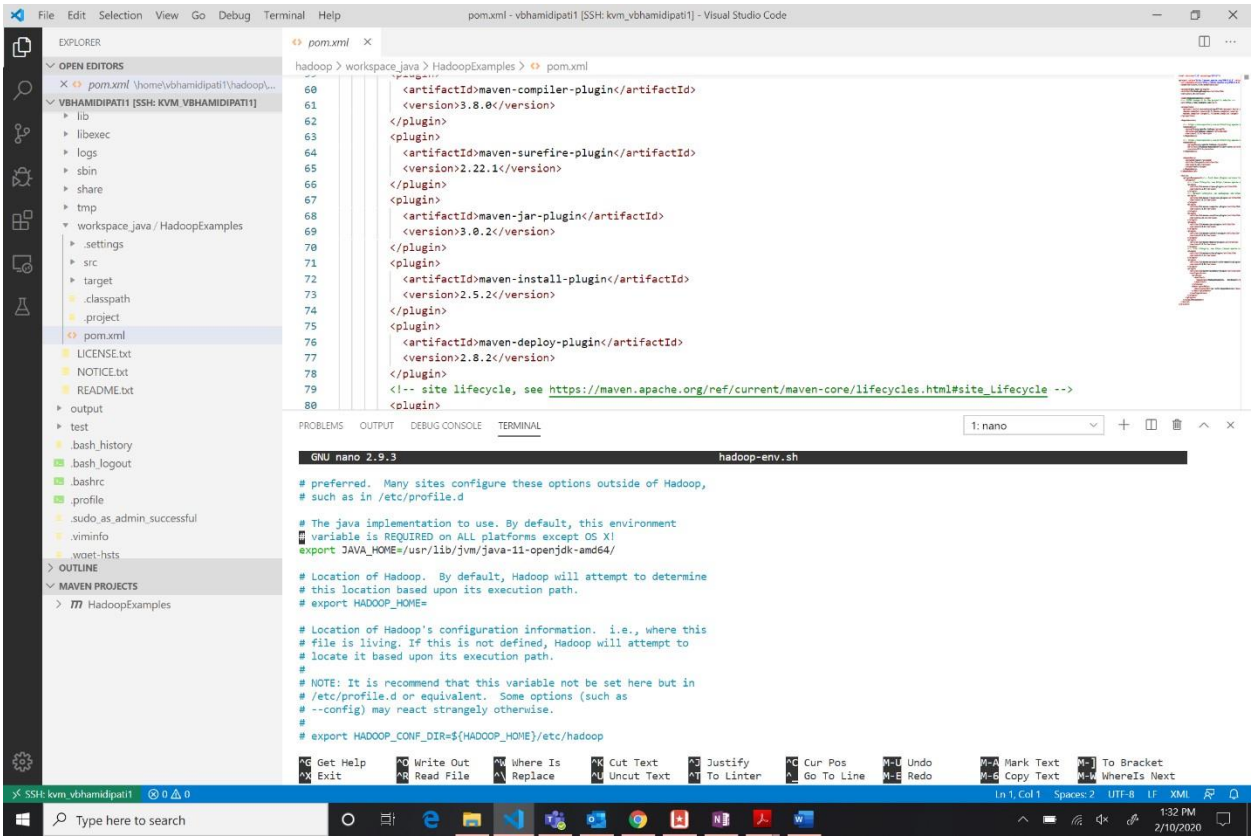
Setting up Hadoop on Ubuntu:

1. Install Java Using
    a. Sudo apt install default-jdk
2. Set the root of the java installation
    a. vim hadoop-env.sh
    b. Update the line port JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64



3. Core-site.xml
    a. This is used to specify the default file system and defaults to the local file system that's why it needs to be set to a HDFS address. This is important for client configuration as well so your local configuration file should include this element.
    b. When client request for the datanode information to the namenode then the namenode give them, the information about the datanode i.e hdfs details and then client write the dataset into the datanode.
    c. Add the following configurations
        i. <configuration>
        ii. <property>
        iii. <name>hadoop.tmp.dir</name>
        iv. <value>/home/vbhamidipati1/hadoop/tmp</value>
        v. </property>
        vi. <property> vii. <name>fs.defaultFS</name> viii. <value>hdfs://localhost:9000</value>

ix. <!-- <value>file:///</value> -->

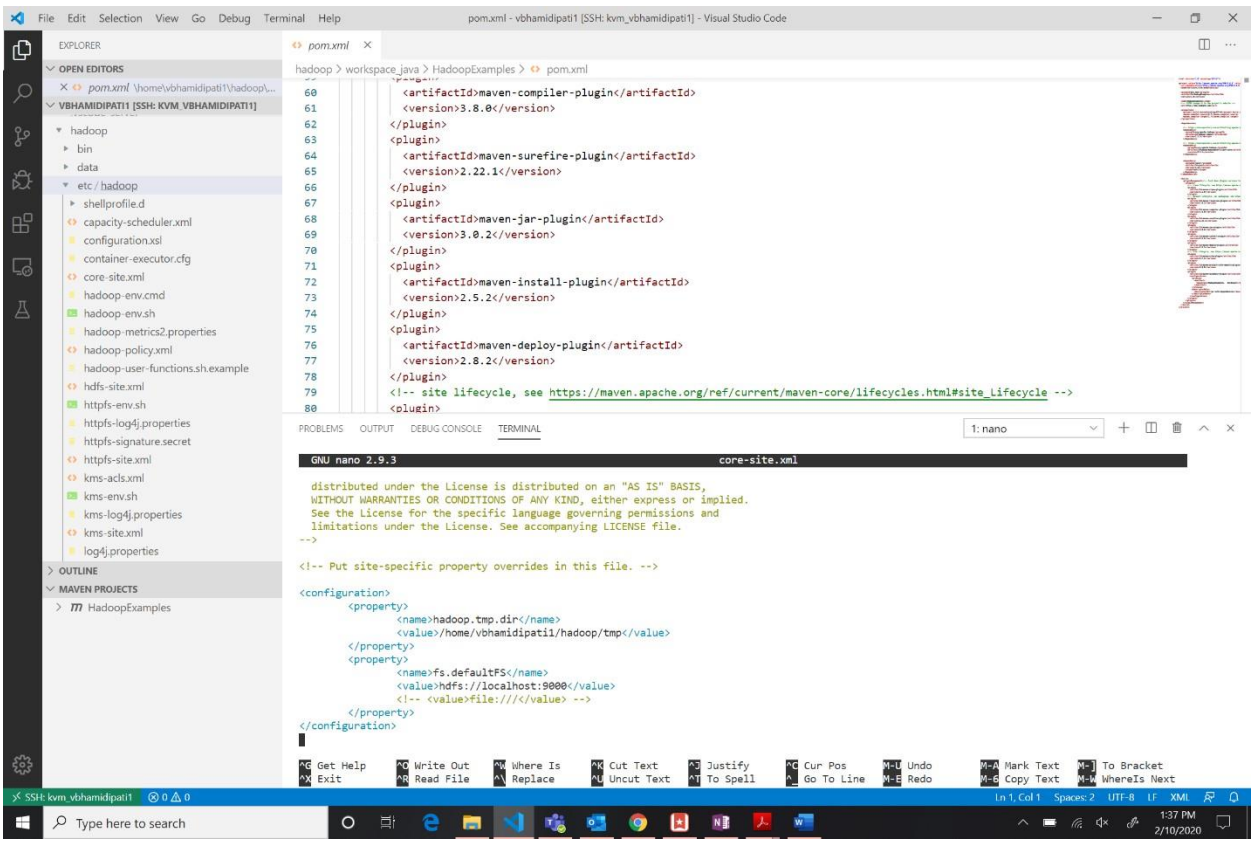
x. </property>

xi. </configuration>

d. Here fs denotes file system and default.name denotes namenode

e. Here 9000 denotes port on which datanode will send heartbeat to namenode. And full address is the machine name that is converted to hostname.

f. The fs.defaultFS makes HDFS a file abstraction over a cluster, so that its root is not the same as the local system's. You need to change the value in order to create the distributed file system. The fs.defaultFS in core-site.xml gives the datanode address of namenode. The datanode looks here for the namenode address and tries to contact it using RPC.



4. hdfs-site.xml

a. Here we set the following:

i. Minimum block size

ii. Default block size

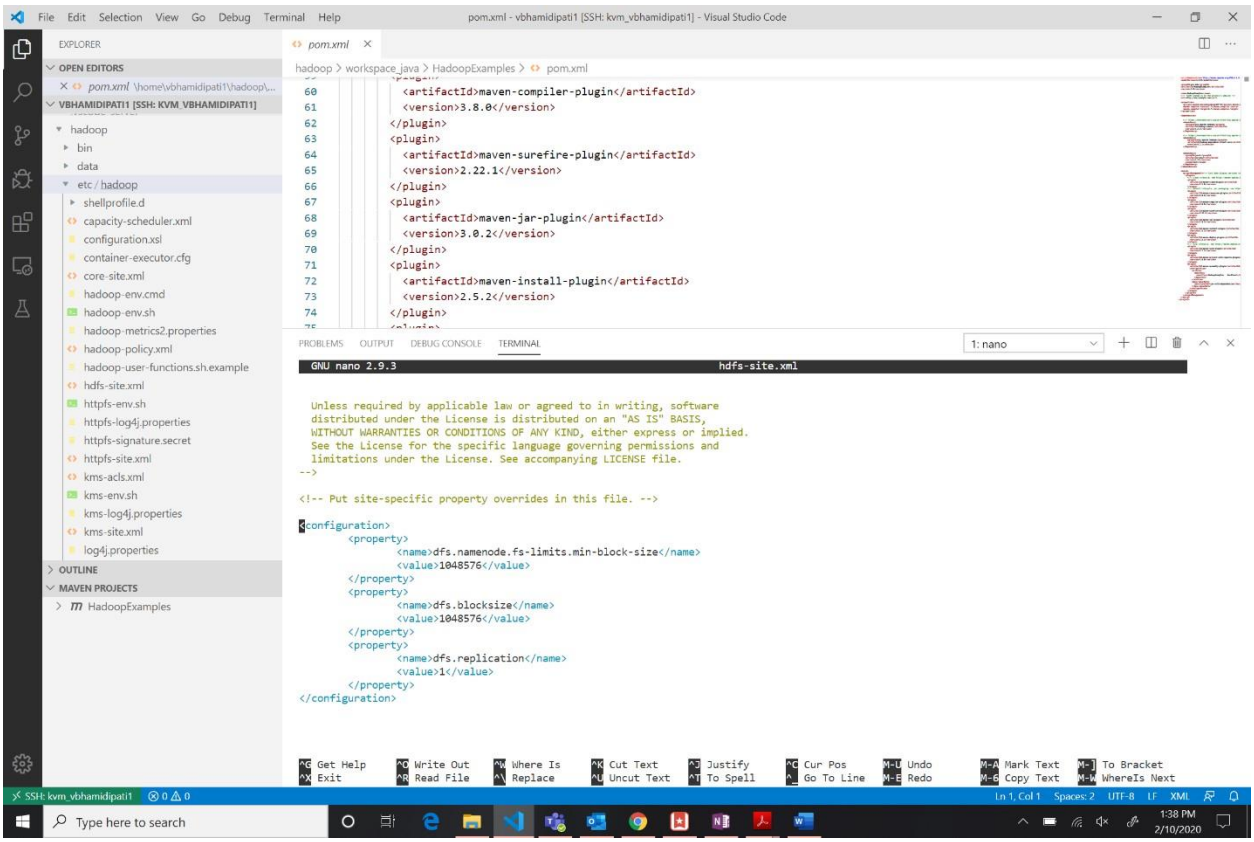
iii. Replication value

b. In Hadoop, HDFS splits huge files into small chunks known as data blocks. HDFS Data blocks are the smallest unit of data in a filesystem. We (client and admin) do not have any control over the data block like block location. Namenode decides all such things.

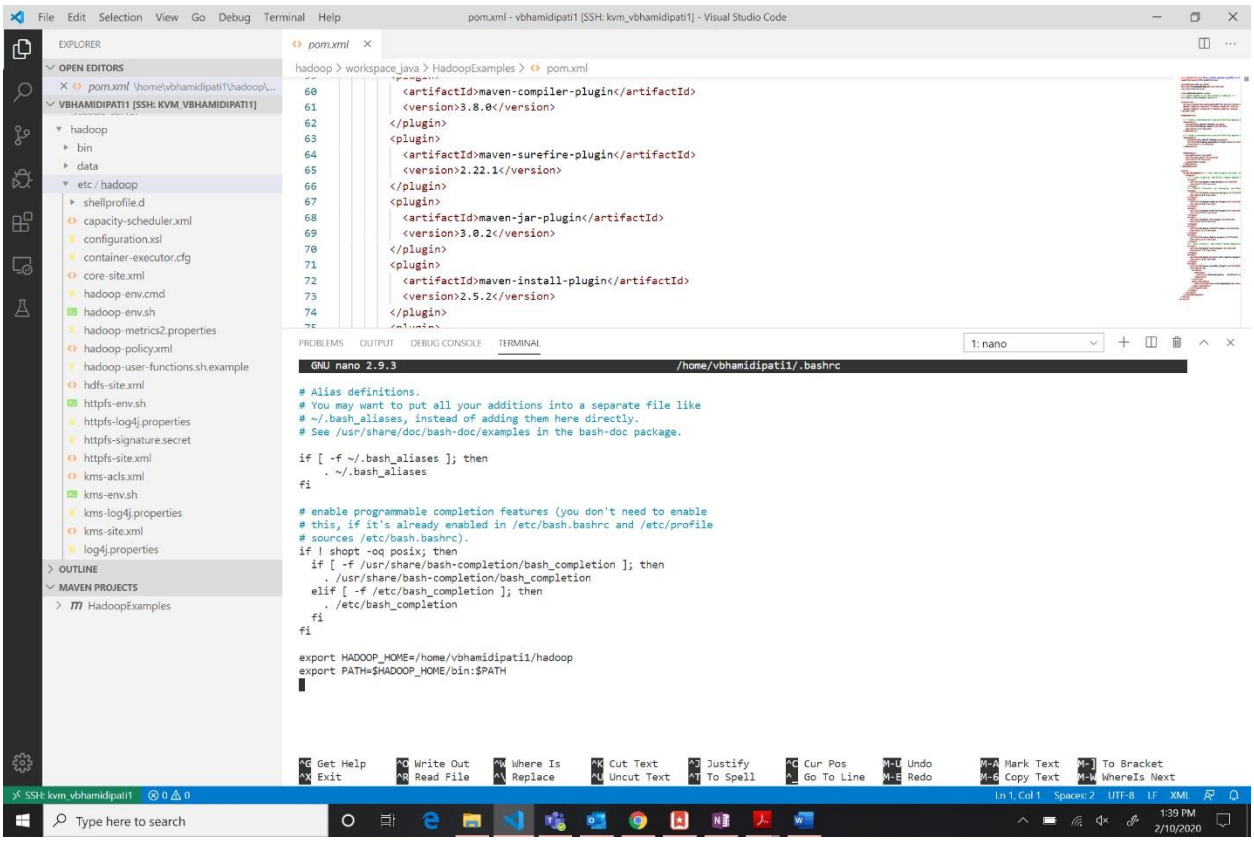
c. HDFS stores each file as a data block. However, the data block size in HDFS is very large. The default size of the HDFS block is 128MB which you can configure as per your

requirement. All blocks of the file are the same size except the last block, which can be either the same size or smaller. The files are split into 128 MB blocks and then stored into the Hadoop file system. The Hadoop application is responsible for distributing the data block across multiple nodes. Here in our example, the block size is 1 MB.

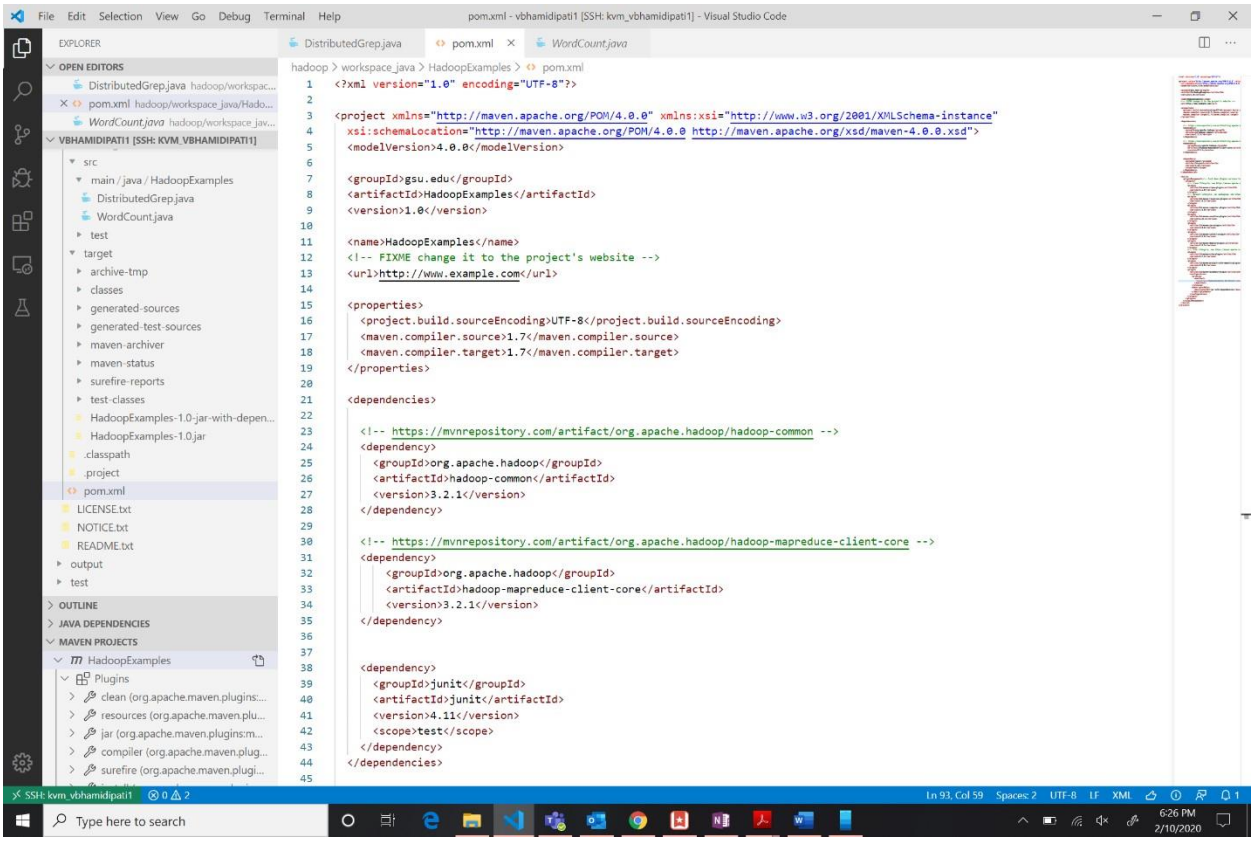
d. Replication factor dictates how many copies of a block should be kept in your cluster. The replication factor is 3 by default (there would be one original block and two replicas) and hence any file you create in HDFS will have a replication factor of 3 and each block from the file will be copied to 3 different nodes in your cluster.

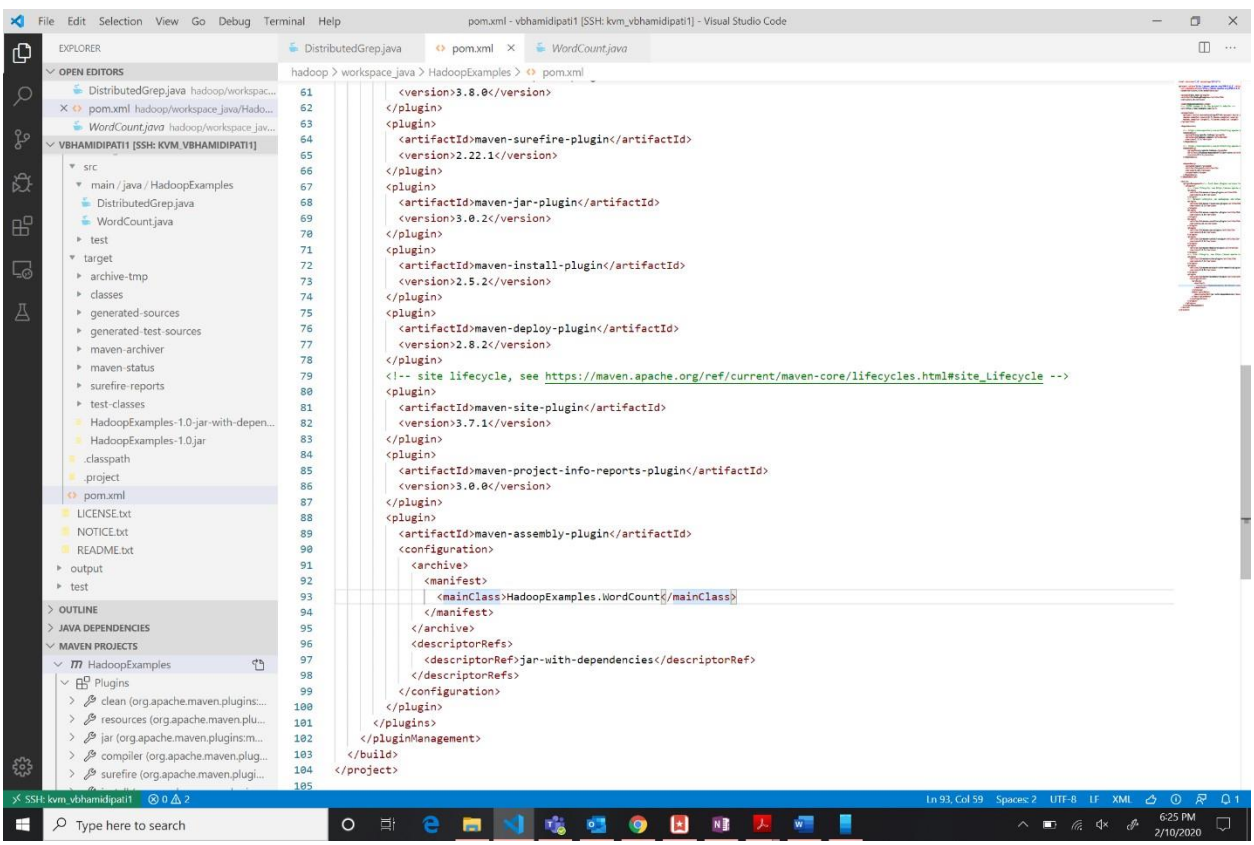


5. bashrc
   a. .bashrc is a shell script that Bash runs whenever it is started interactively. It initializes an interactive shell session. You can put any command in that file that you could type at the command prompt.
   b. Here we set the Hadoop path to identify it globally



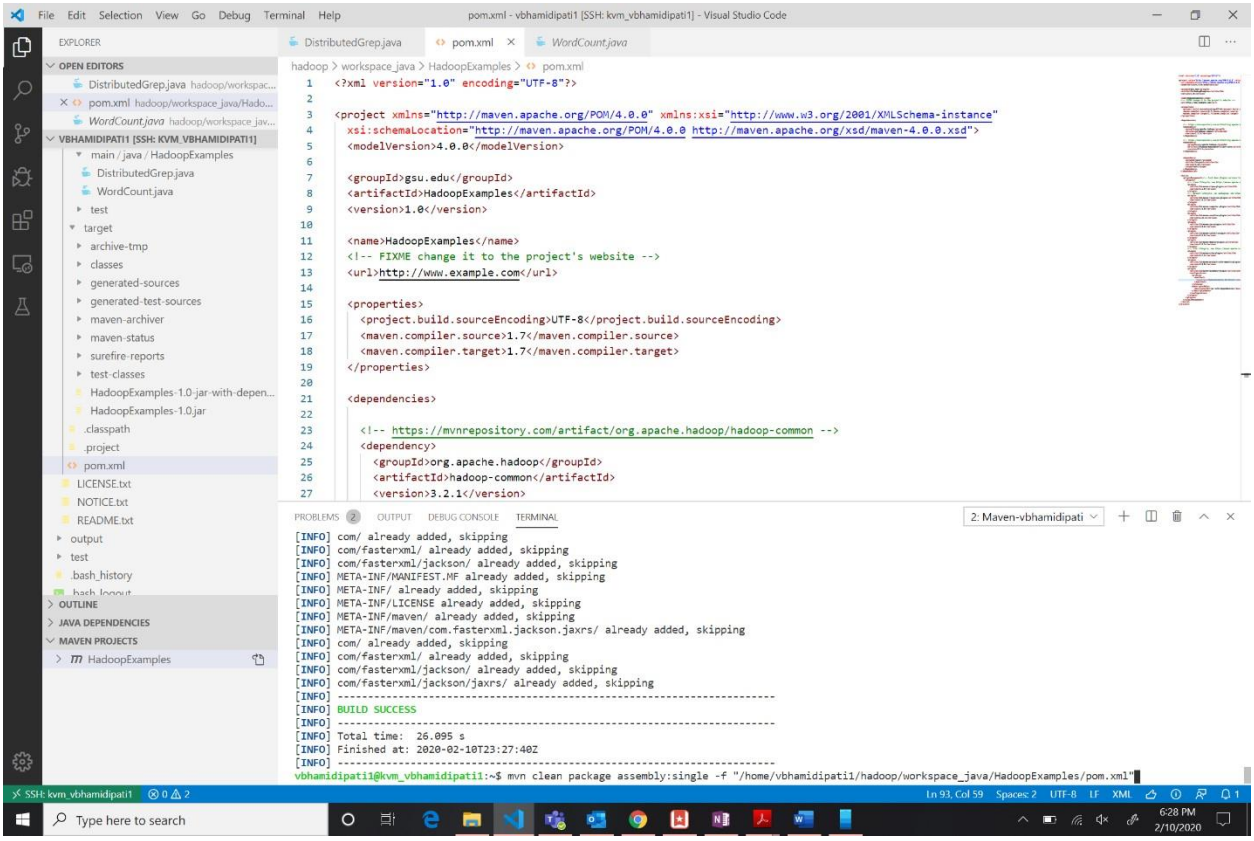
6. Updating the pom.xml file
    a. Add the following dependencies
        i. Apache Hadoop Common 3.2.1
        ii. Apache Hadoop MapReduce Core 3.2.1
    b. Add the dependencies by accessing the dependency code from the maven repository to the pom.xml file
    c. Also specify the main class file for execution as <PackageName.ClassName>
    d. In our example we used HadoopExamples.WordCount as our main class





7. Maven Assembly pulgin and build the project

a. It will create a runnable jar file for the execution of wordcount program
b. The jars can be found in the target folder



8. Running the jar file
   a. We run the jar file using the command

hadoop jar /home/vbhamidipati1/hadoop/workspace_java/HadoopExamples/target/HadoopExamples-1.0-jar-with-dependencies.jar file:///home/vbhamidipati1/hadoop/data/peterpan.txt file:///home/vbhamidipati1/hadoop/data/output

Explanation:

1. The command uses the complete path of jar as stated by "jar /home/vbhamidipati1/hadoop/workspace_java/HadoopExamples/target/HadoopExamples-1.0-jar-with-dependencies.jar"
2. It takes in the input text file as file:///home/vbhamidipati1/hadoop/data/peterpan.txt. It specifies that the file has to be taken from the local (Ubuntu) file system not the HDFS.
3. It outputs text file at file:///home/vbhamidipati1/hadoop/data/output. It specifies that the file is written to local (Ubuntu) file system not the HDFS.

File Edit Selection View Go Debug Terminal Help — pom.xml - vbhamidipati1 [SSH: kvm_vbhamidipati1] - Visual Studio Code

EXPLORER

OPEN EDITORS
- DistributedGrep.java hadoop/workspac...
- pom.xml hadoop/workspace_java/Hado...
- WordCount.java hadoop/workspace_jav...

VBHAMIDIPATI1 [SSH: KVM_VBHAMIDIPATI1]
- include
- lib
- libexec
- logs
- sbin
- share
- tmp
- workspace_java / HadoopExamples 2
- LICENSE.txt
- NOTICE.txt
- README.txt
- output
  - _SUCCESS
  - ._SUCCESS.crc
  - .part-r-00000.crc
  - part-r-00000
- test
- .bash_history
- .bash_logout
- .bashrc
- .core-site.xml.swp
- .profile
- .sudo_as_admin_successful
- .viminfo

OUTLINE

JAVA DEPENDENCIES

MAVEN PROJECTS
- HadoopExamples

DistributedGrep.java | pom.xml | WordCount.java

hadoop > workspace_java > HadoopExamples > pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>gsu.edu</groupId>
```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL — 2: Maven-vbhamidipati

```
vbhamidipati1@kvm_vbhamidipati1:~$ hadoop jar /home/vbhamidipati1/hadoop/workspace_java/HadoopExamples/target/HadoopExamples-1.0-jar-with-dependenc
ies.jar file:///home/vbhamidipati1/hadoop/data/peterpan.txt file:///home/vbhamidipati1/hadoop/data/output
2020-02-10 23:33:02,468 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2020-02-10 23:33:02,733 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2020-02-10 23:33:02,733 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2020-02-10 23:33:02,871 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and exec
ute your application with ToolRunner to remedy this.
2020-02-10 23:33:04,162 INFO input.FileInputFormat: Total input files to process : 1
2020-02-10 23:33:04,206 INFO mapreduce.JobSubmitter: number of splits:1
2020-02-10 23:33:04,573 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local775807566_0001
2020-02-10 23:33:04,576 INFO mapreduce.JobSubmitter: Executing with tokens: []
2020-02-10 23:33:04,796 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
2020-02-10 23:33:04,797 INFO mapreduce.Job: Running job: job_local775807566_0001
2020-02-10 23:33:04,801 INFO mapred.LocalJobRunner: OutputCommitter set in config null
2020-02-10 23:33:04,814 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2020-02-10 23:33:04,814 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore c
leanup failures: false
2020-02-10 23:33:04,815 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
2020-02-10 23:33:04,872 INFO mapred.LocalJobRunner: Waiting for map tasks
2020-02-10 23:33:04,873 INFO mapred.LocalJobRunner: Starting task: attempt_local775807566_0001_m_000000_0
2020-02-10 23:33:04,916 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2020-02-10 23:33:04,918 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore c
leanup failures: false
2020-02-10 23:33:04,961 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
2020-02-10 23:33:04,969 INFO mapred.MapTask: Processing split: file:/home/vbhamidipati1/hadoop/data/peterpan.txt:0+291927
2020-02-10 23:33:05,251 INFO mapred.MapTask: (EQUATOR) 0 kvi 26214396(104857584)
2020-02-10 23:33:05,252 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100
2020-02-10 23:33:05,252 INFO mapred.MapTask: soft limit at 83886080
2020-02-10 23:33:05,253 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600
2020-02-10 23:33:05,253 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
2020-02-10 23:33:05,267 INFO mapred.MapTask: Map output collector class = org.apache.hadoop.mapred.MapTask$MapOutputBuffer
2020-02-10 23:33:05,275 INFO input.LineRecordReader: Found UTF-8 BOM and skipped it
2020-02-10 23:33:05,488 INFO mapred.LocalJobRunner:
2020-02-10 23:33:05,488 INFO mapred.MapTask: Starting flush of map output
2020-02-10 23:33:05,489 INFO mapred.MapTask: Spilling map output
2020-02-10 23:33:05,489 INFO mapred.MapTask: bufstart = 0; bufend = 486076; bufvoid = 104857600
2020-02-10 23:33:05,489 INFO mapred.MapTask: kvstart = 26214396(104857584); kvend = 26011340(104045360); length = 203057/6553600
2020-02-10 23:33:05,716 INFO mapred.MapTask: Finished spill 0
2020-02-10 23:33:05,737 INFO mapred.Task: Task:attempt_local775807566_0001_m_000000_0 is done. And is in the process of committing
2020-02-10 23:33:05,739 INFO mapred.LocalJobRunner: map
2020-02-10 23:33:05,743 INFO mapred.Task: Task 'attempt_local775807566_0001_m_000000_0' done.
2020-02-10 23:33:05,750 INFO mapred.Task: Final Counters for attempt_local775807566_0001_m_000000_0: Counters: 24
```

SSH: kvm_vbhamidipati1 ⊗ 0 ⚠ 2 — Ln 93, Col 59 Spaces: 2 UTF-8 LF XML 1

Type here to search — 6:36 PM 2/10/2020

pom.xml - vbhamidipati1 [SSH: kvm_vbhamidipati1] - Visual Studio Code

hadoop > workspace_java > HadoopExamples > pom.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>gsu.edu</groupId>
  <artifactId>HadoopExamples</artifactId>
  <version>1.0</version>

  <name>HadoopExamples</name>
  <!-- FIXME change it to the project's website -->
  <url>http://www.example.com</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
  </properties>

  <dependencies>

    <!-- https://mvnrepository.com/artifact/org.apache.hadoop/hadoop-common -->
    <dependency>
      <groupId>org.apache.hadoop</groupId>
```

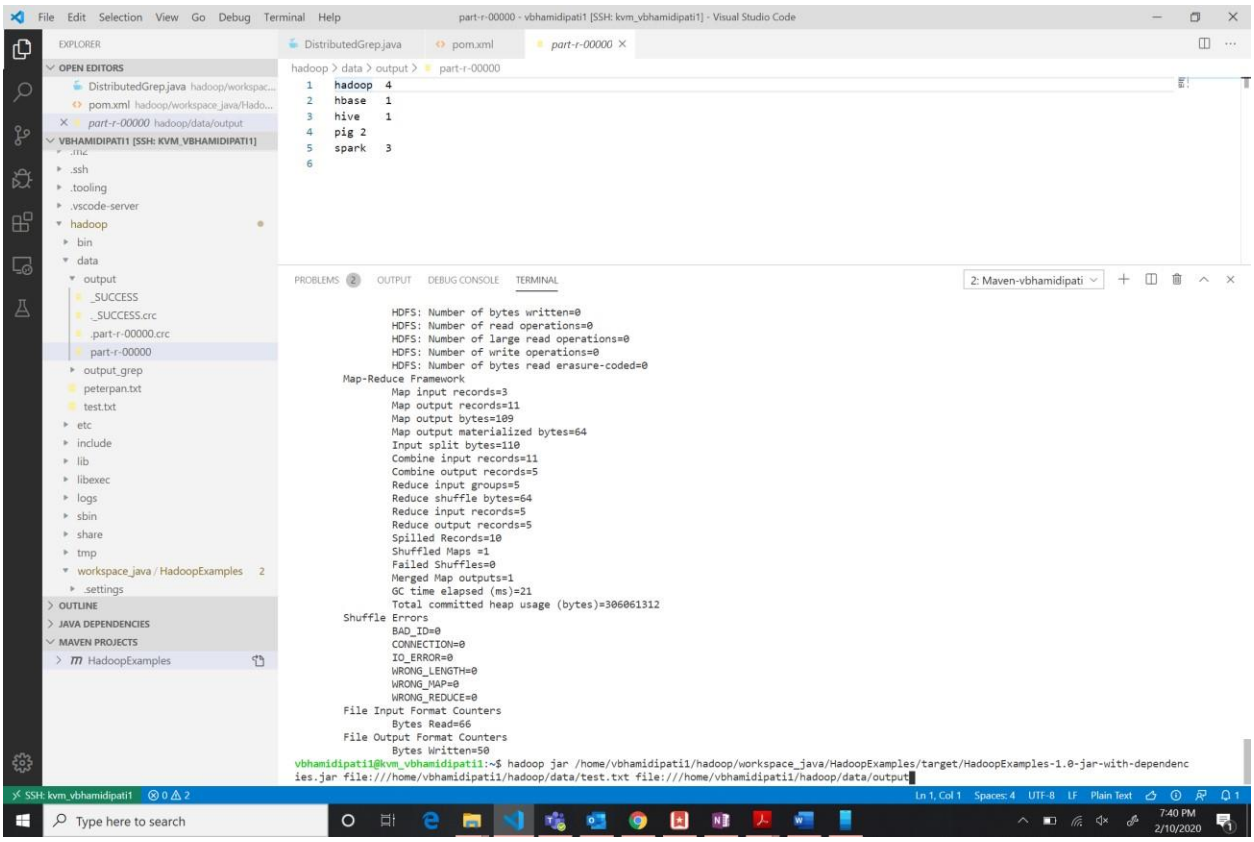
```
vbhamidipati1@kvm_vbhamidipati1:~$ hadoop jar /home/vbhamidipati1/hadoop/workspace_java/HadoopExamples/target/HadoopExamples-1.0-jar-with-dependenc
ies.jar file:///home/vbhamidipati1/hadoop/data/peterpan.txt file:///home/vbhamidipati1/hadoop/data/output
```

part-r-00000 - vbhamidipati1 [SSH: kvm_vbhamidipati1] - Visual Studio Code

output > part-r-00000

```
#10000, 2
#16]    1
$5,000) 1
($1 1
(2) 1
(3) 1
(4) 1
(801)   1
(I  1
(Morgan's   1
(Or 1
(Slightly   1
(a) 1
(and    2
(any    1
(as 1
(available  1
(b) 1
(but    2
(c) 1
(c)1991 1
(does   1
(he 1
(if 1
(or 3
(the    1
(though 1
(trademark/copyright)   1
(which  1
(winking    1
(www.gutenberg.org),    1
(zipped),   1
("the   1
**  4
*** 8
*****   2
-   7
/etext  1
00, 1
```

```
2020-02-10 23:33:05,737 INFO mapred.Task: Task:attempt_local775807566_0001_m_000000_0 is done. And is in the process of committing
2020-02-10 23:33:05,739 INFO mapred.LocalJobRunner: map
2020-02-10 23:33:05,743 INFO mapred.Task: Task 'attempt_local775807566_0001_m_000000_0' done.
2020-02-10 23:33:05,750 INFO mapred.Task: Final Counters for attempt_local775807566_0001_m_000000_0: Counters: 24
```


File Edit Selection View Go Debug Terminal Help
part-r-00000 - vbhamidipati1 [SSH: kvm_vbhamidipati1] - Visual Studio Code
EXPLORER
OPEN EDITORS
DistributedGrep.java hadoop/workspac...
pom.xml hadoop/workspace_java/Hado...
part-r-00000 hadoop/data/output
VBHAMIDIPATI1 [SSH: KVM_VBHAMIDIPATI1]
.ssh
.tooling
.vscode-server
hadoop
bin
data
output
_SUCCESS
_SUCCESS.crc
.part-r-00000.crc
part-r-00000
output_grep
peterpan.txt
test.txt
etc
include
lib
libexec
logs
sbin
share
tmp
workspace_java / HadoopExamples 2
.settings
OUTLINE
JAVA DEPENDENCIES
MAVEN PROJECTS
HadoopExamples
DistributedGrep.java
pom.xml
part-r-00000
hadoop > data > output > part-r-00000
hadoop 4
hbase 1
hive 1
pig 2
spark 3
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL
2: Maven-vbhamidipati
HDFS: Number of bytes written=0
HDFS: Number of read operations=0
HDFS: Number of large read operations=0
HDFS: Number of write operations=0
HDFS: Number of bytes read erasure-coded=0
Map-Reduce Framework
Map input records=3
Map output records=11
Map output bytes=109
Map output materialized bytes=64
Input split bytes=110
Combine input records=11
Combine output records=5
Reduce input groups=5
Reduce shuffle bytes=64
Reduce input records=5
Reduce output records=5
Spilled Records=10
Shuffled Maps =1
Failed Shuffles=0
Merged Map outputs=1
GC time elapsed (ms)=21
Total committed heap usage (bytes)=306061312
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=66
File Output Format Counters
Bytes Written=50
vbhamidipati1@kvm_vbhamidipati1:~$ hadoop jar /home/vbhamidipati1/hadoop/workspace_java/HadoopExamples/target/HadoopExamples-1.0-jar-with-dependencies.jar file:///home/vbhamidipati1/hadoop/data/test.txt file:///home/vbhamidipati1/hadoop/data/output
SSH: kvm_vbhamidipati1
Ln 1, Col 1 Spaces: 4 UTF-8 LF Plain Text
Type here to search
7:40 PM
2/10/2020