BIG DATA CLOUD LAB MANUAL

NAME : PRAGADEESHWARAN K J

ROLL.NO: 20BIS031

COURSE : BIGDATA TECHNOLOGIES

CODE : U18ISI5202T

LAB EXERCISE-1

AIM:

Perform setting up and Installing Hadoop in its three operating modes: Standalone, Pseudo distributed, Fully distributed

INSTALLATION STEPS:

Step 1: Install OpenJDK on Ubuntu-$ sudo apt install

openjdk-8-jdk -y

Verify the installed version of Java- $ java -version

Step 2: Create Hadoop User and Configure Password-less SSH

Add a new user hadoop - $ sudo adduser hadoop

Add the hadoop user to the sudo group-$ sudo usermod -aG sudo hadoop

Switch to the created user - $ sudo su - hadoop

Install the OpenSSH server and client- $ apt install openssh-server openssh-client -y

$ sudo su - hadoop

Generate public and private key pairs- $ ssh-keygen -t rsa

Add the generated public key from id_rsa.pub to authorized_keys

$ sudo cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys

Change the permissions of the authorized_keys file

$ sudo chmod 640 ~/.ssh/authorized_keys

Verify if the password-less SSH is functional

$ ssh localhost

Step 3: Install Apache Hadoop

Log in with hadoop user- $ sudo su - hadoop

Download the latest stable version of Hadoop-

$ wget https://downloads.apache.org/hadoop/common/hadoop-3.3.1/hadoop-3.3.1.tar.gz

Extract the downloaded file- $ tar -xvzf hadoop-3.3.1.tar.gz

Move the extracted directory to the /usr/local/ directory.

$ sudo mv hadoop-3.3.1 /usr/local/hadoop

Create directory to store system logs-

$ sudo mkdir /usr/local/hadoop/logs

Change the ownership of the hadoop directory

$ sudo chown -R hadoop:hadoop /usr/local/hadoop

Step 4: Configure Hadoop

Edit file ~/.bashrc to configure the Hadoop environment variables

$ sudo nano ~/.bashrc

Add the following lines to the file. Save and close the file.

export HADOOP_HOME=/usr/local/hadoop

export HADOOP_INSTALL=$HADOOP_HOME

export HADOOP_MAPRED_HOME=$HADOOP_HOME

export HADOOP_COMMON_HOME=$HADOOP_HOME

export HADOOP_HDFS_HOME=$HADOOP_HOME

export YARN_HOME=$HADOOP_HOME

export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native

export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin

export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"

Activate the environment variables.

$ source ~/.bashrc

Step 5: Configure Java Environment Variables

Find the OpenJDK directory- $ readlink -f /usr/bin/javac

Edit the hadoop-env.sh file-

$ sudo nano $HADOOP_HOME/etc/hadoop/hadoop-env.sh

Add the following lines to the file. Then, close and save the file-export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64

export HADOOP_CLASSPATH+=" $HADOOP_HOME/lib/*.jar"

Browse to the hadoop lib directory-$ cd /usr/local/hadoop/lib

Download the Javax activation file- $ sudo wget https://jcenter.bintray.com/javax/activation/javax.activation-api/1.2.0/javax.activation-api-1.2.0.jar

Verify the Hadoop version-$ hadoop version

Edit the core-site.xml configuration file to specify the URL for your NameNode-$ sudo nano $HADOOP_HOME/etc/hadoop/core-site.xml

Add the following lines. Save and close the file.

<configuration>

  <property>

    <name>fs.default.name</name>

    <value>hdfs://0.0.0.0:9000</value>

```
    <description>The default file system URI</description>
  </property>
</configuration>
```

Create a directory for storing node metadata and change the ownership to hadoop.

```
$ sudo mkdir -p /home/hadoop/hdfs/{namenode,datanode}
$ sudo nano $HADOOP_HOME/etc/hadoop/hdfs-site.xml
```

Add the following lines. Close and save the file.

```
<configuration>
  <property>
    <name>dfs.replication</name>
    <value>1</value>
  </property>
  <property>
    <name>dfs.name.dir</name>
    <value>file:///home/hadoop/hdfs/namenode</value>
  </property>
  <property>
    <name>dfs.data.dir</name>
    <value>file:///home/hadoop/hdfs/datanode</value>
  </property>
</configuration>
```

Edit mapred-site.xml configuration file to define MapReduce values- $ sudo nano $HADOOP_HOME/etc/hadoop/mapred-site.xml

Add the following lines. Save and close the file.

```
<configuration>
  <property>
    <name>mapreduce.framework.name</name>
```

<value>yarn</value>

</property>

</configuration>

Edit the yarn-site.xml configuration file and define YARN-related settings- $ sudo nano $HADOOP_HOME/etc/hadoop/yarn-site.xml

Add the following lines. Save and close the file.

<configuration>

<property>

<name>yarn.nodemanager.aux-services</name>

<value>mapreduce_shuffle</value>

</property>

</configuration>

Log in with hadoop user-$ sudo su - hadoop

Validate the Hadoop configuration and format the HDFS NameNode-$ hdfs namenode -format

Step 6: Start the Apache Hadoop Cluster

Start the NameNode and DataNode-$ start-dfs.sh

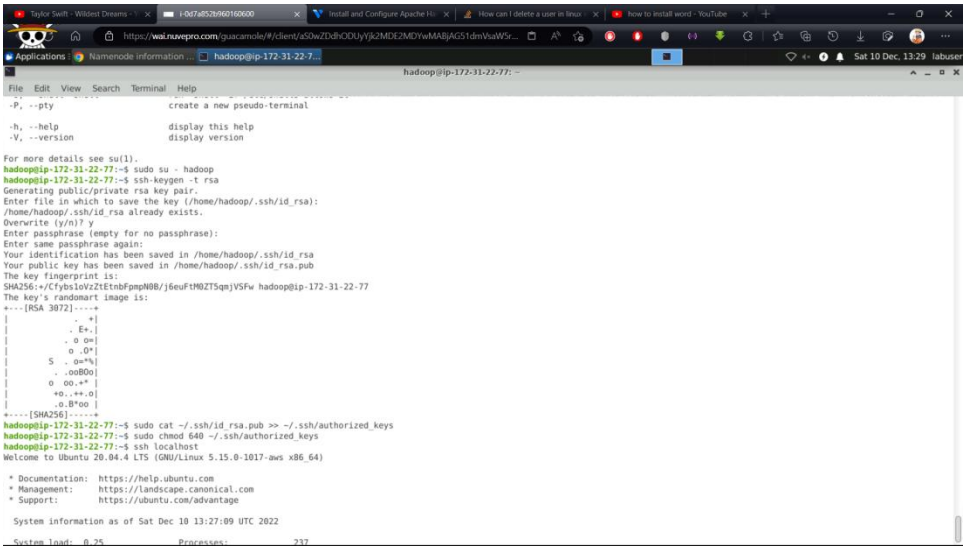
Start the YARN resource and node managers-$ start-yarn.sh

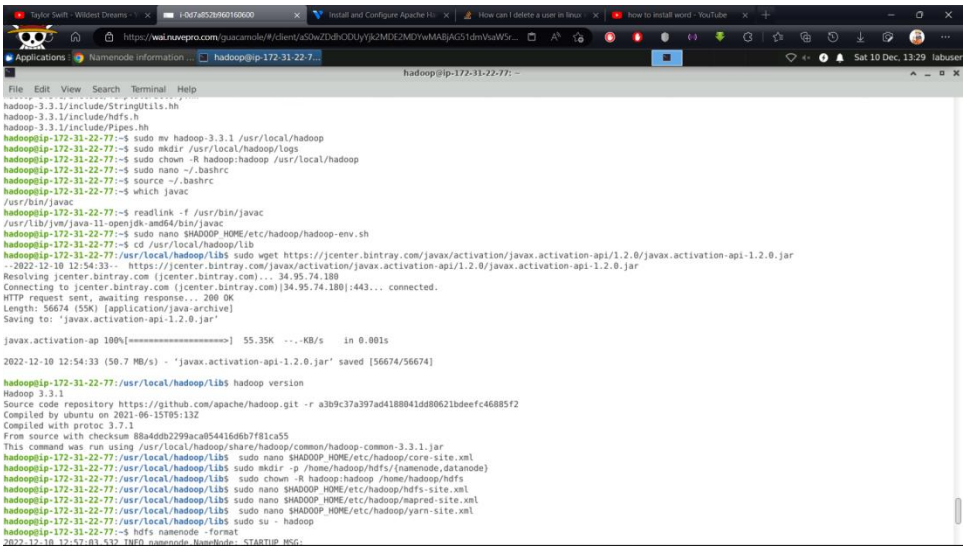
Verify all the running components-$ jps

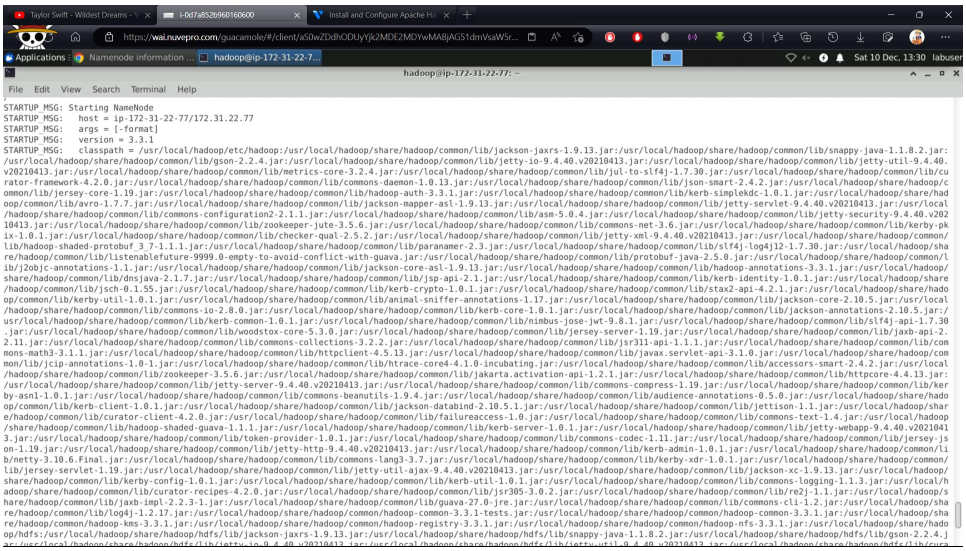
Step 7: Access Apache Hadoop Web Interface

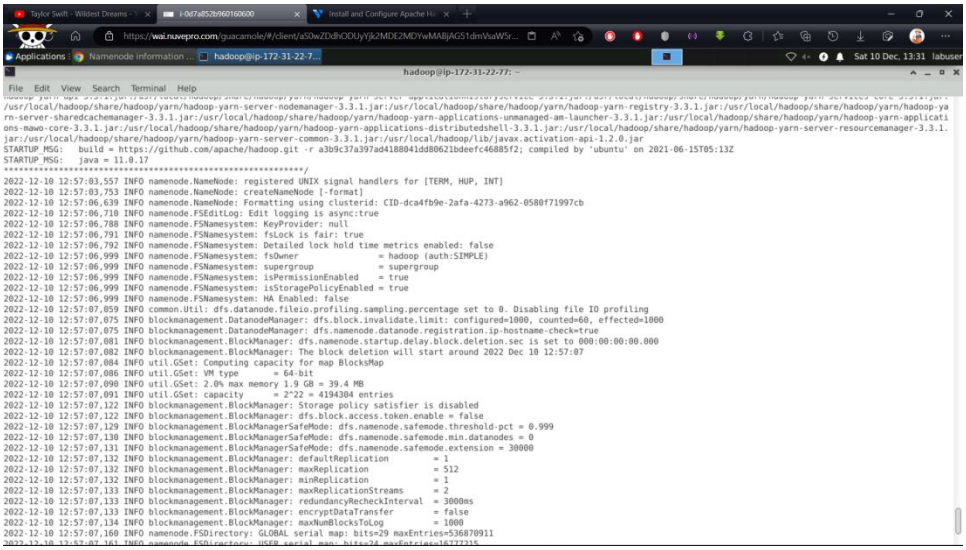
http://localhost:9870

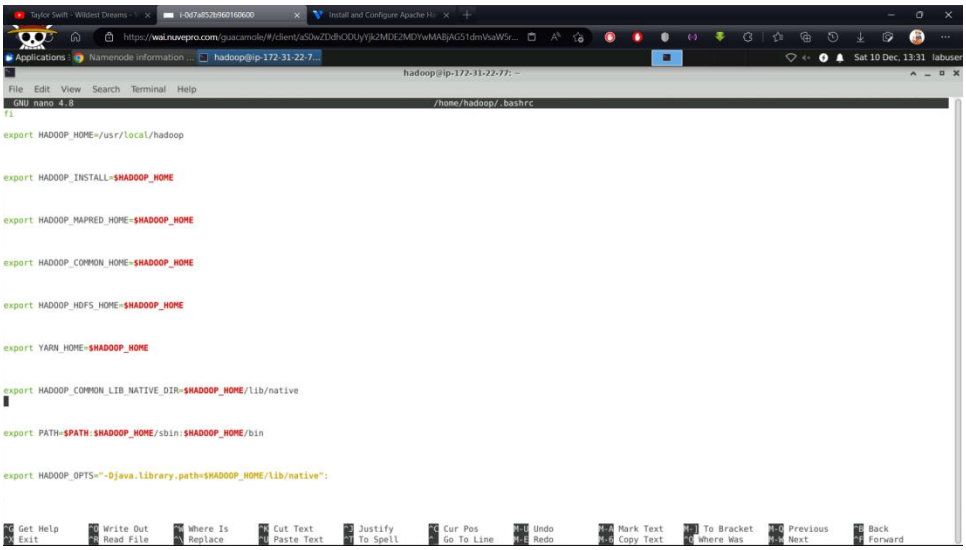
Screenshots:

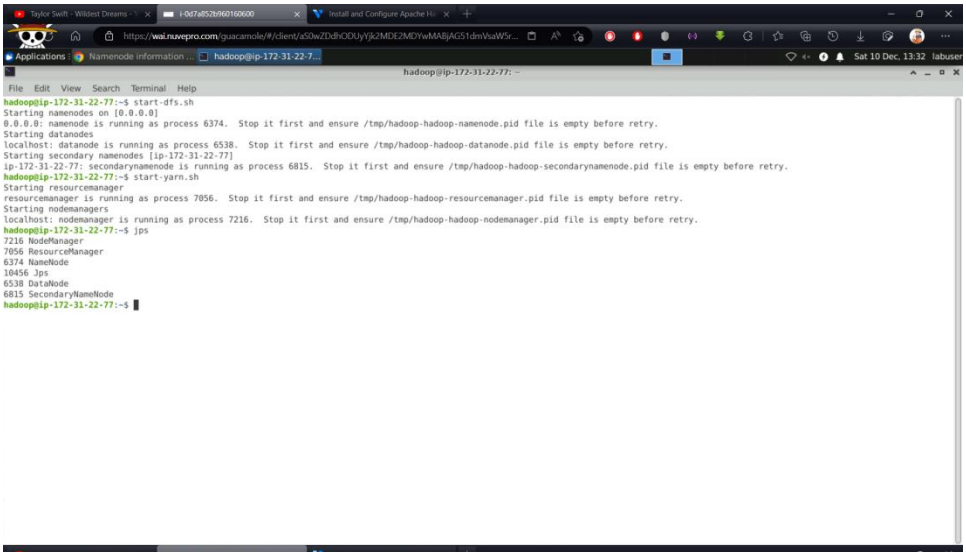

Taylor Swift - Wildest Dreams
i-0d7a852b960160600
Install and Configure Apache Ha
How can I delete a user in linux
how to install word - YouTube
https://wai.nuvepro.com/guacamole/#/client/aS0wZDdhODUyYjk2MDE2MDYwMABjAG51dmVsaW5r...
Applications
Namenode information ...
hadoop@ip-172-31-22-7...
Sat 10 Dec, 13:29
labuser
hadoop@ip-172-31-22-77: ~
File Edit View Search Terminal Help
-P, --pty create a new pseudo-terminal
-h, --help display this help
-V, --version display version
For more details see su(1).
hadoop@ip-172-31-22-77:~$ sudo su - hadoop
hadoop@ip-172-31-22-77:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hadoop/.ssh/id_rsa):
/home/hadoop/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/hadoop/.ssh/id_rsa
Your public key has been saved in /home/hadoop/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:+/CfybsIoVz2tEtnbFpmpN0B/j6euFtM0ZT5qmjVSFw hadoop@ip-172-31-22-77
The key's randomart image is:
+---[RSA 3072]----+
| . +|
| . E+.|
| . o o=|
| o .O*|
| S . o=*%|
| . .ooBOo|
| o oo.+* |
| +o..++.o|
| .o.B*oo |
+----[SHA256]-----+
hadoop@ip-172-31-22-77:~$ sudo cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
hadoop@ip-172-31-22-77:~$ sudo chmod 640 ~/.ssh/authorized_keys
hadoop@ip-172-31-22-77:~$ ssh localhost
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.15.0-1017-aws x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage
System information as of Sat Dec 10 13:27:09 UTC 2022
System load: 0.25 Processes: 237

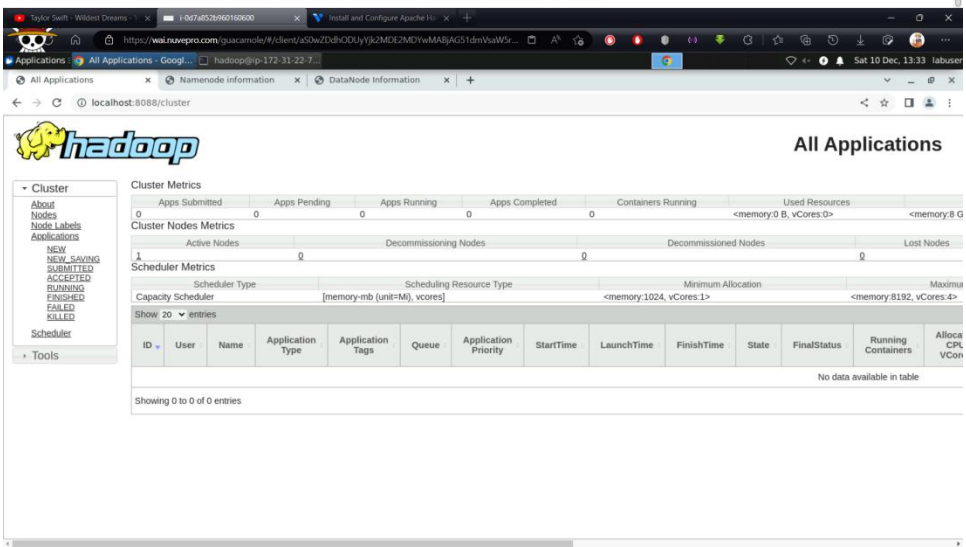

Taylor Swift - Wildest Dreams
i-0d7a852b960160600
Install and Configure Apache Ha
How can I delete a user in linux
how to install word - YouTube
https://wai.nuvepro.com/guacamole/#/client/aS0wZDdhODUyYjk2MDE2MDYwMABjAG51dmVsaW5r...
Applications
Namenode information ...
hadoop@ip-172-31-22-7...
Sat 10 Dec, 13:29
labuser
hadoop@ip-172-31-22-77: ~
File Edit View Search Terminal Help
hadoop-3.3.1/include/StringUtils.hh
hadoop-3.3.1/include/hdfs.h
hadoop-3.3.1/include/Pipes.hh
hadoop@ip-172-31-22-77:~$ sudo mv hadoop-3.3.1 /usr/local/hadoop
hadoop@ip-172-31-22-77:~$ sudo mkdir /usr/local/hadoop/logs
hadoop@ip-172-31-22-77:~$ sudo chown -R hadoop:hadoop /usr/local/hadoop
hadoop@ip-172-31-22-77:~$ sudo nano ~/.bashrc
hadoop@ip-172-31-22-77:~$ source ~/.bashrc
hadoop@ip-172-31-22-77:~$ which javac
/usr/bin/javac
hadoop@ip-172-31-22-77:~$ readlink -f /usr/bin/javac
/usr/lib/jvm/java-11-openjdk-amd64/bin/javac
hadoop@ip-172-31-22-77:~$ sudo nano $HADOOP_HOME/etc/hadoop/hadoop-env.sh
hadoop@ip-172-31-22-77:~$ cd /usr/local/hadoop/lib
hadoop@ip-172-31-22-77:/usr/local/hadoop/lib$ sudo wget https://jcenter.bintray.com/javax/activation/javax.activation-api/1.2.0/javax.activation-api-1.2.0.jar
--2022-12-10 12:54:33-- https://jcenter.bintray.com/javax/activation/javax.activation-api/1.2.0/javax.activation-api-1.2.0.jar
Resolving jcenter.bintray.com (jcenter.bintray.com)... 34.95.74.180
Connecting to jcenter.bintray.com (jcenter.bintray.com)|34.95.74.180|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 56674 (55K) [application/java-archive]
Saving to: 'javax.activation-api-1.2.0.jar'
javax.activation-ap 100%[==========>] 55.35K --.-KB/s in 0.001s
2022-12-10 12:54:33 (50.7 MB/s) - 'javax.activation-api-1.2.0.jar' saved [56674/56674]
hadoop@ip-172-31-22-77:/usr/local/hadoop/lib$ hadoop version
Hadoop 3.3.1
Source code repository https://github.com/apache/hadoop.git -r a3b9c37a397ad4188041dd80621bdeefc46885f2
Compiled by ubuntu on 2021-06-15T05:13Z
Compiled with protoc 3.7.1
From source with checksum 88a4ddb2299aca054416d6b7f81ca55
This command was run using /usr/local/hadoop/share/hadoop/common/hadoop-common-3.3.1.jar
hadoop@ip-172-31-22-77:/usr/local/hadoop/lib$ sudo nano $HADOOP_HOME/etc/hadoop/core-site.xml
hadoop@ip-172-31-22-77:/usr/local/hadoop/lib$ sudo mkdir -p /home/hadoop/hdfs/{namenode,datanode}
hadoop@ip-172-31-22-77:/usr/local/hadoop/lib$ sudo chown -R hadoop:hadoop /home/hadoop/hdfs
hadoop@ip-172-31-22-77:/usr/local/hadoop/lib$ sudo nano $HADOOP_HOME/etc/hadoop/hdfs-site.xml
hadoop@ip-172-31-22-77:/usr/local/hadoop/lib$ sudo nano $HADOOP_HOME/etc/hadoop/mapred-site.xml
hadoop@ip-172-31-22-77:/usr/local/hadoop/lib$ sudo nano $HADOOP_HOME/etc/hadoop/yarn-site.xml
hadoop@ip-172-31-22-77:/usr/local/hadoop/lib$ sudo su - hadoop
hadoop@ip-172-31-22-77:~$ hdfs namenode -format
2022-12-10 12:57:03,532 INFO namenode.NameNode: STARTUP_MSG:

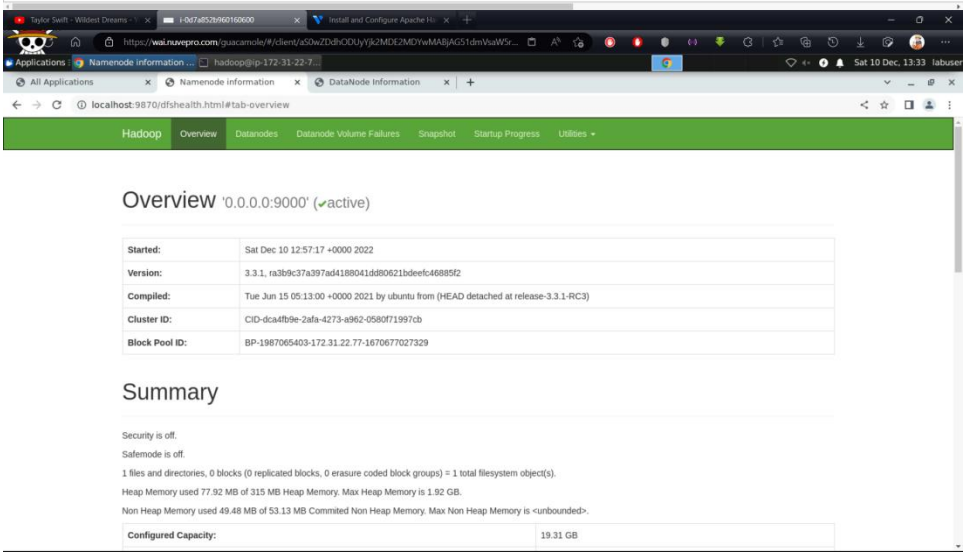

Taylor Swift - Wildest Dreams
i-0d7a852b960160600
Install and Configure Apache Ha
https://wai.nuvepro.com/guacamole/#/client/aS0wZDdhODUyYjk2MDE2MDYwMABjAG51dmVsaW5r...
Applications
Namenode information ...
hadoop@ip-172-31-22-7...
Sat 10 Dec, 13:30
labuser
hadoop@ip-172-31-22-77: ~
File Edit View Search Terminal Help
STARTUP_MSG: Starting NameNode
STARTUP_MSG: host = ip-172-31-22-77/172.31.22.77
STARTUP_MSG: args = [-format]
STARTUP_MSG: version = 3.3.1
STARTUP_MSG: classpath = /usr/local/hadoop/etc/hadoop:/usr/local/hadoop/share/hadoop/common/lib/jackson-jaxrs-1.9.13.jar:/usr/local/hadoop/share/hadoop/common/lib/snappy-java-1.1.8.2.jar:
/usr/local/hadoop/share/hadoop/common/lib/gson-2.2.4.jar:/usr/local/hadoop/share/hadoop/common/lib/jetty-io-9.4.40.v20210413.jar:/usr/local/hadoop/share/hadoop/common/lib/jetty-util-9.4.40.
v20210413.jar:/usr/local/hadoop/share/hadoop/common/lib/metrics-core-3.2.4.jar:/usr/local/hadoop/share/hadoop/common/lib/jul-to-slf4j-1.7.30.jar:/usr/local/hadoop/share/hadoop/common/lib/cu
rator-framework-4.2.0.jar:/usr/local/hadoop/share/hadoop/common/lib/commons-daemon-1.0.13.jar:/usr/local/hadoop/share/hadoop/common/lib/json-smart-2.4.2.jar:/usr/local/hadoop/share/hadoop/c
ommon/lib/jersey-core-1.19.jar:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-3.3.1.jar:/usr/local/hadoop/share/hadoop/common/lib/kerb-simplekdc-1.0.1.jar:/usr/local/hadoop/share/had
oop/common/lib/avro-1.7.7.jar:/usr/local/hadoop/share/hadoop/common/lib/jackson-mapper-asl-1.9.13.jar:/usr/local/hadoop/share/hadoop/common/lib/jetty-servlet-9.4.40.v20210413.jar:/usr/local
/hadoop/share/hadoop/common/lib/commons-configuration2-2.1.1.jar:/usr/local/hadoop/share/hadoop/common/lib/asm-5.0.4.jar:/usr/local/hadoop/share/hadoop/common/lib/jetty-security-9.4.40.v202
10413.jar:/usr/local/hadoop/share/hadoop/common/lib/zookeeper-jute-3.5.6.jar:/usr/local/hadoop/share/hadoop/common/lib/commons-net-3.6.jar:/usr/local/hadoop/share/hadoop/common/lib/kerby-pk
ix-1.0.1.jar:/usr/local/hadoop/share/hadoop/common/lib/checker-qual-2.5.2.jar:/usr/local/hadoop/share/hadoop/common/lib/jetty-xml-9.4.40.v20210413.jar:/usr/local/hadoop/share/hadoop/common/
lib/hadoop-shaded-protobuf_3_7-1.1.1.jar:/usr/local/hadoop/share/hadoop/common/lib/paranamer-2.3.jar:/usr/local/hadoop/share/hadoop/common/lib/slf4j-log4j12-1.7.30.jar:/usr/local/hadoop/sha
re/hadoop/common/lib/listenablefuture-9999.0-empty-to-avoid-conflict-with-guava.jar:/usr/local/hadoop/share/hadoop/common/lib/protobuf-java-2.5.0.jar:/usr/local/hadoop/share/hadoop/common/l
ib/j2objc-annotations-1.1.jar:/usr/local/hadoop/share/hadoop/common/lib/jackson-core-asl-1.9.13.jar:/usr/local/hadoop/share/hadoop/common/lib/hadoop-annotations-3.3.1.jar:/usr/local/hadoop/
share/hadoop/common/lib/dnsjava-2.1.7.jar:/usr/local/hadoop/share/hadoop/common/lib/jsp-api-2.1.jar:/usr/local/hadoop/share/hadoop/common/lib/kerb-identity-1.0.1.jar:/usr/local/hadoop/share
/hadoop/common/lib/jsch-0.1.55.jar:/usr/local/hadoop/share/hadoop/common/lib/kerb-crypto-1.0.1.jar:/usr/local/hadoop/share/hadoop/common/lib/stax2-api-4.2.1.jar:/usr/local/hadoop/share/hado
op/common/lib/kerby-util-1.0.1.jar:/usr/local/hadoop/share/hadoop/common/lib/animal-sniffer-annotations-1.17.jar:/usr/local/hadoop/share/hadoop/common/lib/jackson-core-2.10.5.jar:/usr/local
/hadoop/share/hadoop/common/lib/commons-io-2.8.0.jar:/usr/local/hadoop/share/hadoop/common/lib/kerb-core-1.0.1.jar:/usr/local/hadoop/share/hadoop/common/lib/jackson-annotations-2.10.5.jar:/
usr/local/hadoop/share/hadoop/common/lib/kerb-common-1.0.1.jar:/usr/local/hadoop/share/hadoop/common/lib/nimbus-jose-jwt-9.8.1.jar:/usr/local/hadoop/share/hadoop/common/lib/slf4j-api-1.7.30
.jar:/usr/local/hadoop/share/hadoop/common/lib/woodstox-core-5.3.0.jar:/usr/local/hadoop/share/hadoop/common/lib/jersey-server-1.19.jar:/usr/local/hadoop/share/hadoop/common/lib/jaxb-api-2.
2.11.jar:/usr/local/hadoop/share/hadoop/common/lib/commons-collections-3.2.2.jar:/usr/local/hadoop/share/hadoop/common/lib/jsr311-api-1.1.1.jar:/usr/local/hadoop/share/hadoop/common/lib/com
mons-math3-3.1.1.jar:/usr/local/hadoop/share/hadoop/common/lib/httpclient-4.5.13.jar:/usr/local/hadoop/share/hadoop/common/lib/javax.servlet-api-3.1.0.jar:/usr/local/hadoop/share/hadoop/com
mon/lib/jcip-annotations-1.0-1.jar:/usr/local/hadoop/share/hadoop/common/lib/htrace-core4-4.1.0-incubating.jar:/usr/local/hadoop/share/hadoop/common/lib/accessors-smart-2.4.2.jar:/usr/local
/hadoop/share/hadoop/common/lib/zookeeper-3.5.6.jar:/usr/local/hadoop/share/hadoop/common/lib/jakarta.activation-api-1.2.1.jar:/usr/local/hadoop/share/hadoop/common/lib/httpcore-4.4.13.jar:
/usr/local/hadoop/share/hadoop/common/lib/jetty-server-9.4.40.v20210413.jar:/usr/local/hadoop/share/hadoop/common/lib/commons-compress-1.19.jar:/usr/local/hadoop/share/hadoop/common/lib/ker
by-asn1-1.0.1.jar:/usr/local/hadoop/share/hadoop/common/lib/commons-beanutils-1.9.4.jar:/usr/local/hadoop/share/hadoop/common/lib/audience-annotations-0.5.0.jar:/usr/local/hadoop/share/hado
op/common/lib/kerb-client-1.0.1.jar:/usr/local/hadoop/share/hadoop/common/lib/jackson-databind-2.10.5.1.jar:/usr/local/hadoop/share/hadoop/common/lib/jettison-1.1.jar:/usr/local/hadoop/shar
e/hadoop/common/lib/curator-client-4.2.0.jar:/usr/local/hadoop/share/hadoop/common/lib/failureaccess-1.0.jar:/usr/local/hadoop/share/hadoop/common/lib/commons-text-1.4.jar:/usr/local/hadoop
/share/hadoop/common/lib/hadoop-shaded-guava-1.1.1.jar:/usr/local/hadoop/share/hadoop/common/lib/kerb-server-1.0.1.jar:/usr/local/hadoop/share/hadoop/common/lib/jetty-webapp-9.4.40.v2021041
3.jar:/usr/local/hadoop/share/hadoop/common/lib/token-provider-1.0.1.jar:/usr/local/hadoop/share/hadoop/common/lib/commons-codec-1.11.jar:/usr/local/hadoop/share/hadoop/common/lib/jersey-js
on-1.19.jar:/usr/local/hadoop/share/hadoop/common/lib/jetty-http-9.4.40.v20210413.jar:/usr/local/hadoop/share/hadoop/common/lib/kerb-admin-1.0.1.jar:/usr/local/hadoop/share/hadoop/common/li
b/netty-3.10.6.Final.jar:/usr/local/hadoop/share/hadoop/common/lib/commons-lang3-3.7.jar:/usr/local/hadoop/share/hadoop/common/lib/kerby-xdr-1.0.1.jar:/usr/local/hadoop/share/hadoop/common/
lib/jersey-servlet-1.19.jar:/usr/local/hadoop/share/hadoop/common/lib/jetty-util-ajax-9.4.40.v20210413.jar:/usr/local/hadoop/share/hadoop/common/lib/jackson-xc-1.9.13.jar:/usr/local/hadoop/
share/hadoop/common/lib/kerby-config-1.0.1.jar:/usr/local/hadoop/share/hadoop/common/lib/kerb-util-1.0.1.jar:/usr/local/hadoop/share/hadoop/common/lib/commons-logging-1.1.3.jar:/usr/local/h
adoop/share/hadoop/common/lib/curator-recipes-4.2.0.jar:/usr/local/hadoop/share/hadoop/common/lib/jsr305-3.0.2.jar:/usr/local/hadoop/share/hadoop/common/lib/re2j-1.1.jar:/usr/local/hadoop/s
hare/hadoop/common/lib/javax.activation-api-1.2.0.jar:/usr/local/hadoop/share/hadoop/common/lib/guava-27.0-jre.jar:/usr/local/hadoop/share/hadoop/common/lib/commons-cli-1.2.jar:/usr/local/hadoop/sha
re/hadoop/common/lib/log4j-1.2.17.jar:/usr/local/hadoop/share/hadoop/common/hadoop-common-3.3.1-tests.jar:/usr/local/hadoop/share/hadoop/common/hadoop-common-3.3.1.jar:/usr/local/hadoop/sha
re/hadoop/common/hadoop-kms-3.3.1.jar:/usr/local/hadoop/share/hadoop/common/hadoop-registry-3.3.1.jar:/usr/local/hadoop/share/hadoop/common/hadoop-nfs-3.3.1.jar:/usr/local/hadoop/share/hado
op/hdfs:/usr/local/hadoop/share/hadoop/hdfs/lib/jackson-jaxrs-1.9.13.jar:/usr/local/hadoop/share/hadoop/hdfs/lib/snappy-java-1.1.8.2.jar:/usr/local/hadoop/share/hadoop/hdfs/lib/gson-2.2.4.j
ar:/usr/local/hadoop/share/hadoop/hdfs/lib/jetty-io-9.4.40.v20210413.jar:/usr/local/hadoop/share/hadoop/hdfs/lib/jetty-util-9.4.40.v20210413.jar:/usr/local/hadoop/share/hadoop/hdfs/lib/jcip-

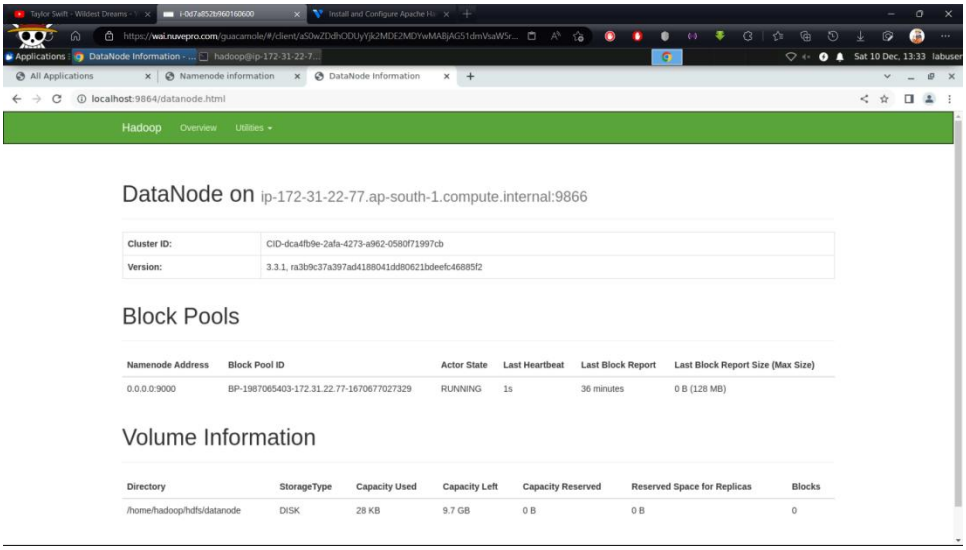

Taylor Swift - Wildest Dreams
i-0d7a852b960160600
Install and Configure Apache Ha
https://wai.nuvepro.com/guacamole/#/client/aS0wZDdhODUyYjk2MDE2MDYwMABjAG51dmVsaW5r...
Applications
Namenode information ...
hadoop@ip-172-31-22-7...
Sat 10 Dec, 13:31 labuser
hadoop@ip-172-31-22-77: ~
File Edit View Search Terminal Help
/usr/local/hadoop/share/hadoop/yarn/hadoop-yarn-server-nodemanager-3.3.1.jar:/usr/local/hadoop/share/hadoop/yarn/hadoop-yarn-registry-3.3.1.jar:/usr/local/hadoop/share/hadoop/yarn/hadoop-yarn-server-sharedcachemanager-3.3.1.jar:/usr/local/hadoop/share/hadoop/yarn/hadoop-yarn-applications-unmanaged-am-launcher-3.3.1.jar:/usr/local/hadoop/share/hadoop/yarn/hadoop-yarn-applications-mawo-core-3.3.1.jar:/usr/local/hadoop/share/hadoop/yarn/hadoop-yarn-applications-distributedshell-3.3.1.jar:/usr/local/hadoop/share/hadoop/yarn/hadoop-yarn-server-resourcemanager-3.3.1.jar:/usr/local/hadoop/share/hadoop/yarn/hadoop-yarn-server-common-3.3.1.jar:/usr/local/hadoop/lib/javax.activation-api-1.2.0.jar
STARTUP_MSG: build = https://github.com/apache/hadoop.git -r a3b9c37a397ad4188041dd80621bdeefc46885f2; compiled by 'ubuntu' on 2021-06-15T05:13Z
STARTUP_MSG: java = 11.0.17
**/
2022-12-10 12:57:03,557 INFO namenode.NameNode: registered UNIX signal handlers for [TERM, HUP, INT]
2022-12-10 12:57:03,753 INFO namenode.NameNode: createNameNode [-format]
2022-12-10 12:57:06,639 INFO namenode.NameNode: Formatting using clusterid: CID-dca4fb9e-2afa-4273-a962-0580f71997cb
2022-12-10 12:57:06,710 INFO namenode.FSEditLog: Edit logging is async:true
2022-12-10 12:57:06,788 INFO namenode.FSNamesystem: KeyProvider: null
2022-12-10 12:57:06,791 INFO namenode.FSNamesystem: fsLock is fair: true
2022-12-10 12:57:06,792 INFO namenode.FSNamesystem: Detailed lock hold time metrics enabled: false
2022-12-10 12:57:06,999 INFO namenode.FSNamesystem: fsOwner = hadoop (auth:SIMPLE)
2022-12-10 12:57:06,999 INFO namenode.FSNamesystem: supergroup = supergroup
2022-12-10 12:57:06,999 INFO namenode.FSNamesystem: isPermissionEnabled = true
2022-12-10 12:57:06,999 INFO namenode.FSNamesystem: isStoragePolicyEnabled = true
2022-12-10 12:57:06,999 INFO namenode.FSNamesystem: HA Enabled: false
2022-12-10 12:57:07,059 INFO common.Util: dfs.datanode.fileio.profiling.sampling.percentage set to 0. Disabling file IO profiling
2022-12-10 12:57:07,075 INFO blockmanagement.DatanodeManager: dfs.block.invalidate.limit: configured=1000, counted=60, effected=1000
2022-12-10 12:57:07,075 INFO blockmanagement.DatanodeManager: dfs.namenode.datanode.registration.ip-hostname-check=true
2022-12-10 12:57:07,081 INFO blockmanagement.BlockManager: dfs.namenode.startup.delay.block.deletion.sec is set to 000:00:00:00.000
2022-12-10 12:57:07,082 INFO blockmanagement.BlockManager: The block deletion will start around 2022 Dec 10 12:57:07
2022-12-10 12:57:07,084 INFO util.GSet: Computing capacity for map BlocksMap
2022-12-10 12:57:07,086 INFO util.GSet: VM type = 64-bit
2022-12-10 12:57:07,090 INFO util.GSet: 2.0% max memory 1.9 GB = 39.4 MB
2022-12-10 12:57:07,091 INFO util.GSet: capacity = 2^22 = 4194304 entries
2022-12-10 12:57:07,122 INFO blockmanagement.BlockManager: Storage policy satisfier is disabled
2022-12-10 12:57:07,122 INFO blockmanagement.BlockManager: dfs.block.access.token.enable = false
2022-12-10 12:57:07,129 INFO blockmanagement.BlockManagerSafeMode: dfs.namenode.safemode.threshold-pct = 0.999
2022-12-10 12:57:07,130 INFO blockmanagement.BlockManagerSafeMode: dfs.namenode.safemode.min.datanodes = 0
2022-12-10 12:57:07,131 INFO blockmanagement.BlockManagerSafeMode: dfs.namenode.safemode.extension = 30000
2022-12-10 12:57:07,132 INFO blockmanagement.BlockManager: defaultReplication = 1
2022-12-10 12:57:07,132 INFO blockmanagement.BlockManager: maxReplication = 512
2022-12-10 12:57:07,132 INFO blockmanagement.BlockManager: minReplication = 1
2022-12-10 12:57:07,133 INFO blockmanagement.BlockManager: maxReplicationStreams = 2
2022-12-10 12:57:07,133 INFO blockmanagement.BlockManager: redundancyRecheckInterval = 3000ms
2022-12-10 12:57:07,133 INFO blockmanagement.BlockManager: encryptDataTransfer = false
2022-12-10 12:57:07,134 INFO blockmanagement.BlockManager: maxNumBlocksToLog = 1000
2022-12-10 12:57:07,160 INFO namenode.FSDirectory: GLOBAL serial map: bits=29 maxEntries=536870911


Taylor Swift - Wildest Dreams
i-0d7a852b960160600
Install and Configure Apache Ha
https://wai.nuvepro.com/guacamole/#/client/aS0wZDdhODUyYjk2MDE2MDYwMABjAG51dmVsaW5r...
Applications
Namenode information ...
hadoop@ip-172-31-22-7...
Sat 10 Dec, 13:31 labuser
hadoop@ip-172-31-22-77: ~
File Edit View Search Terminal Help
GNU nano 4.8 /home/hadoop/.bashrc
fi
export HADOOP_HOME=/usr/local/hadoop
export HADOOP_INSTALL=$HADOOP_HOME
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos ^/ Undo ^6 Mark Text ^] To Bracket ^B Previous ^P Back
^X Exit ^R Read File ^\ Replace ^U Paste Text ^T To Spell ^_ Go To Line ^E Redo ^^ Copy Text ^Q Where Was ^N Next ^F Forward


hadoop@ip-172-31-22-77: ~
File Edit View Search Terminal Help
hadoop@ip-172-31-22-77:~$ start-dfs.sh
Starting namenodes on [0.0.0.0]
0.0.0.0: namenode is running as process 6374. Stop it first and ensure /tmp/hadoop-hadoop-namenode.pid file is empty before retry.
Starting datanodes
localhost: datanode is running as process 6538. Stop it first and ensure /tmp/hadoop-hadoop-datanode.pid file is empty before retry.
Starting secondary namenodes [ip-172-31-22-77]
ip-172-31-22-77: secondarynamenode is running as process 6815. Stop it first and ensure /tmp/hadoop-hadoop-secondarynamenode.pid file is empty before retry.
hadoop@ip-172-31-22-77:~$ start-yarn.sh
Starting resourcemanager
resourcemanager is running as process 7056. Stop it first and ensure /tmp/hadoop-hadoop-resourcemanager.pid file is empty before retry.
Starting nodemanagers
localhost: nodemanager is running as process 7216. Stop it first and ensure /tmp/hadoop-hadoop-nodemanager.pid file is empty before retry.
hadoop@ip-172-31-22-77:~$ jps
7216 NodeManager
7056 ResourceManager
6374 NameNode
10456 Jps
6538 DataNode
6815 SecondaryNameNode
hadoop@ip-172-31-22-77:~$


All Applications
Namenode information
DataNode Information
localhost:8088/cluster
All Applications
Cluster
About
Nodes
Node Labels
Applications
NEW
NEW_SAVING
SUBMITTED
ACCEPTED
RUNNING
FINISHED
FAILED
KILLED
Scheduler
Tools
Cluster Metrics
Apps Submitted
Apps Pending
Apps Running
Apps Completed
Containers Running
Used Resources
Cluster Nodes Metrics
Active Nodes
Decommissioning Nodes
Decommissioned Nodes
Lost Nodes
Scheduler Metrics
Scheduler Type
Scheduling Resource Type
Minimum Allocation
Capacity Scheduler
[memory-mb (unit=Mi), vcores]
<memory:1024, vCores:1>
Show 20 entries
ID
User
Name
Application Type
Application Tags
Queue
Application Priority
StartTime
LaunchTime
FinishTime
State
FinalStatus
Running Containers
No data available in table
Showing 0 to 0 of 0 entries


All Applications
Namenode information
DataNode Information
localhost:9870/dfshealth.html#tab-overview
Hadoop
Overview
Datanodes
Datanode Volume Failures
Snapshot
Startup Progress
Utilities
Overview '0.0.0.0:9000' (active)
Started: Sat Dec 10 12:57:17 +0000 2022
Version: 3.3.1, ra3b9c37a397ad4188041dd80621bdeefc46885f2
Compiled: Tue Jun 15 05:13:00 +0000 2021 by ubuntu from (HEAD detached at release-3.3.1-RC3)
Cluster ID: CID-dca4fb9e-2afa-4273-a962-0580f71997cb
Block Pool ID: BP-1987065403-172.31.22.77-1670677027329
Summary
Security is off.
Safemode is off.
1 files and directories, 0 blocks (0 replicated blocks, 0 erasure coded block groups) = 1 total filesystem object(s).
Heap Memory used 77.92 MB of 315 MB Heap Memory. Max Heap Memory is 1.92 GB.
Non Heap Memory used 49.48 MB of 53.13 MB Commited Non Heap Memory. Max Non Heap Memory is <unbounded>.
Configured Capacity: 19.31 GB


Hadoop
Overview
Utilities
DataNode on ip-172-31-22-77.ap-south-1.compute.internal:9866
Cluster ID: CID-dca4fb9e-2afa-4273-a962-0580f71997cb
Version: 3.3.1, ra3b9c37a397ad4188041dd80621bdeefc46885f2
Block Pools
Namenode Address | Block Pool ID | Actor State | Last Heartbeat | Last Block Report | Last Block Report Size (Max Size)
0.0.0.0:9000 | BP-1987065403-172.31.22.77-1670677027329 | RUNNING | 1s | 36 minutes | 0 B (128 MB)
Volume Information
Directory | StorageType | Capacity Used | Capacity Left | Capacity Reserved | Reserved Space for Replicas | Blocks
/home/hadoop/hdfs/datanode | DISK | 28 KB | 9.7 GB | 0 B | 0 B | 0

LAB EXERCISE-2

AIM:

Implement the following file management tasks in Hadoop:

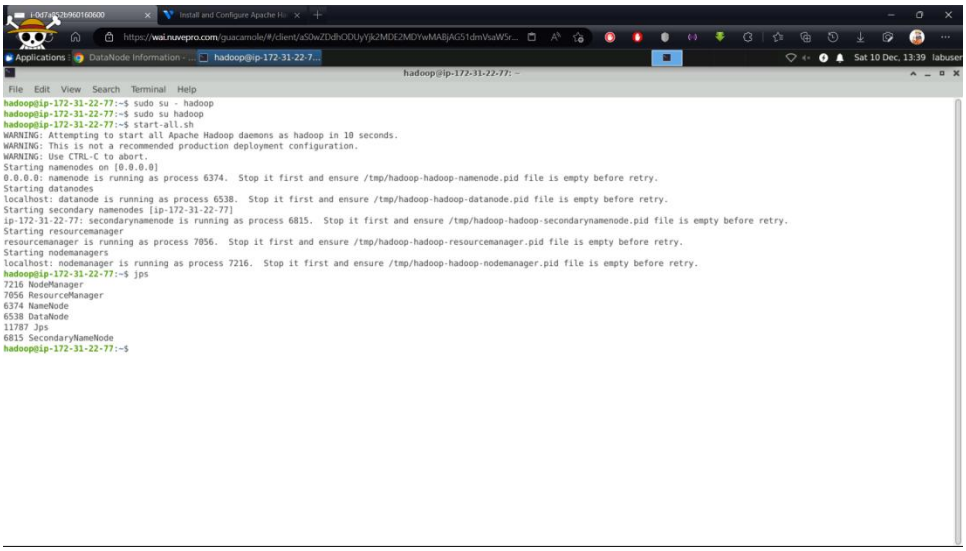
•Adding files and directories

•Retrieving files

•Deleting files

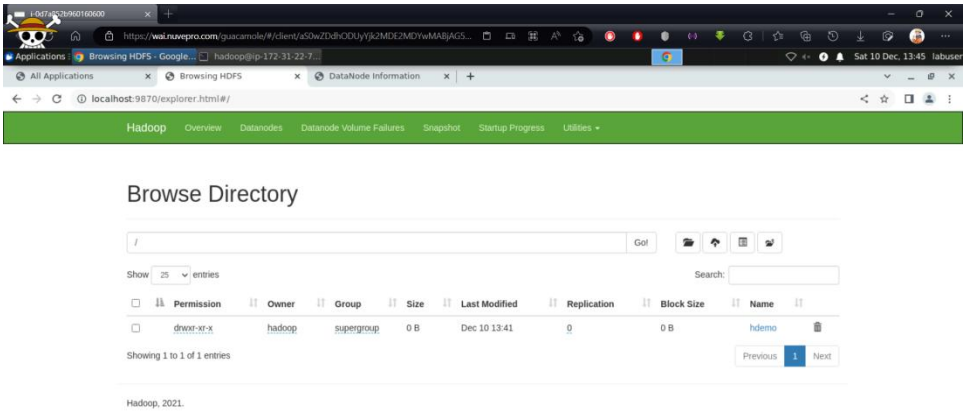
FILE OPERATIONS:

Step 1.Open terminal in Ubuntu and start a cluster



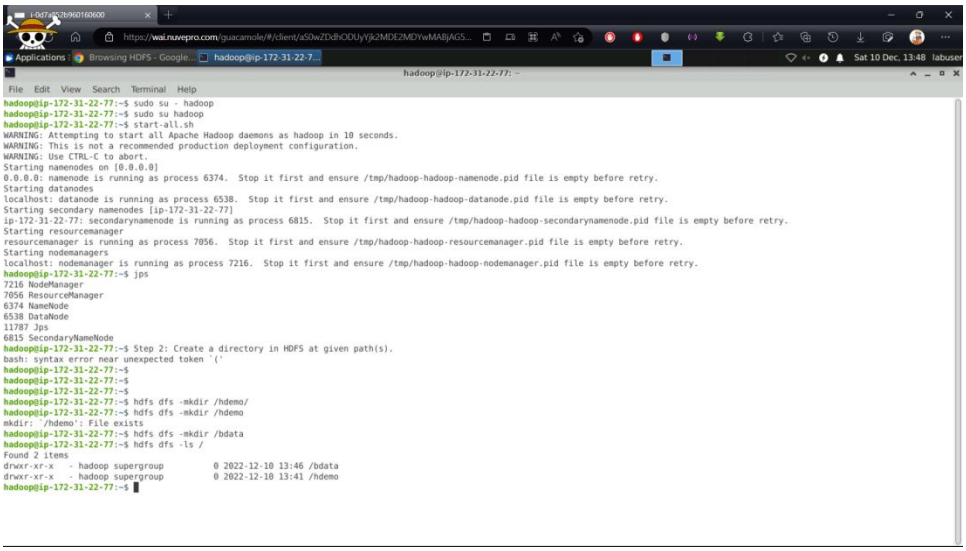
Step 2: Create a directory in HDFS at given path(s).

hdfs dfs -mkdir /hdemo/



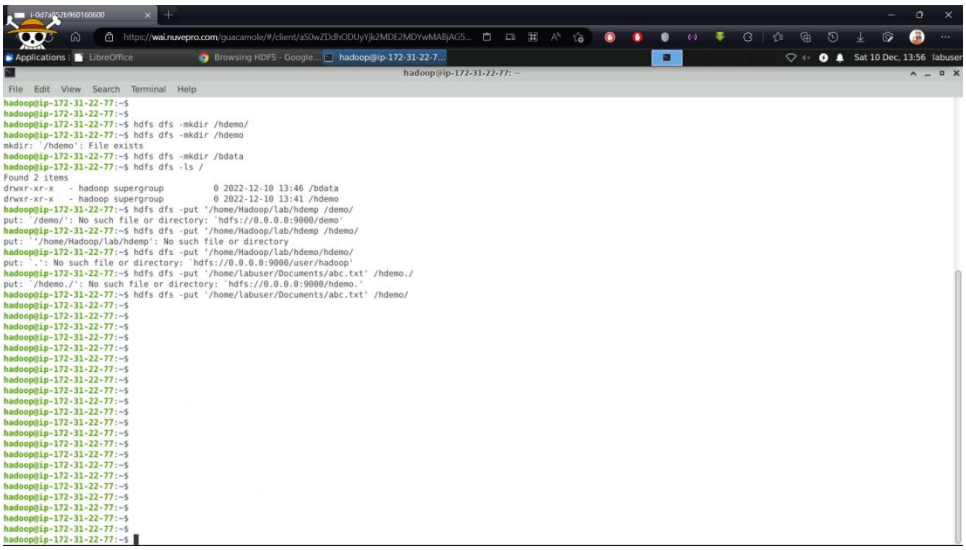
Step 3: List the contents of a directory.

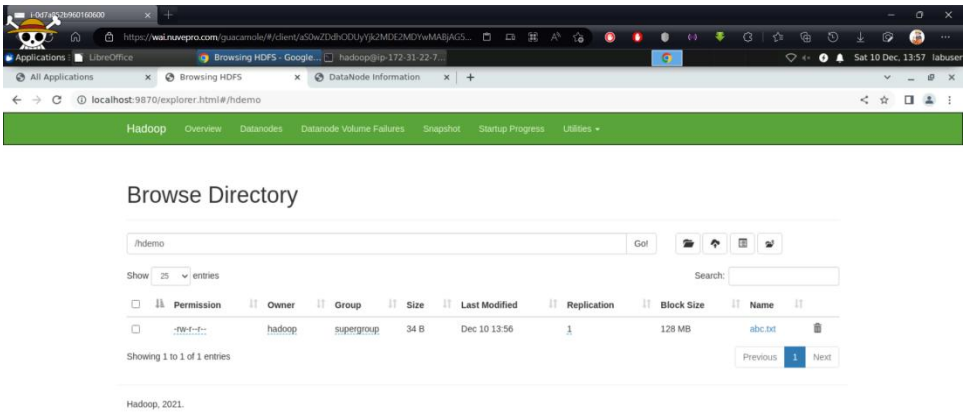
hdfs dfs -ls /



Step 4: Upload and download a file in H Upload

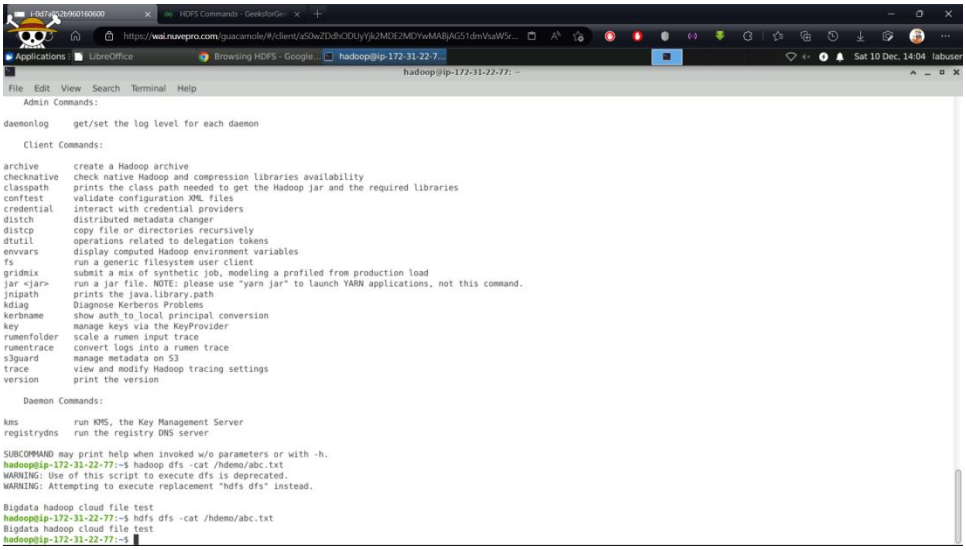
hdfs dfs -put '/home/labuser/Documents/abc.txt' /hdemo/





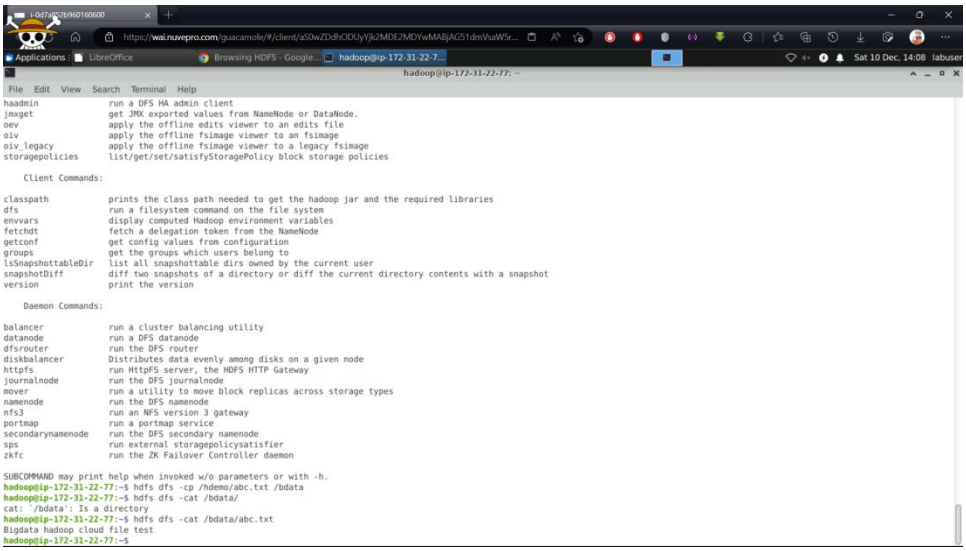
Step 5: See contents of a file:

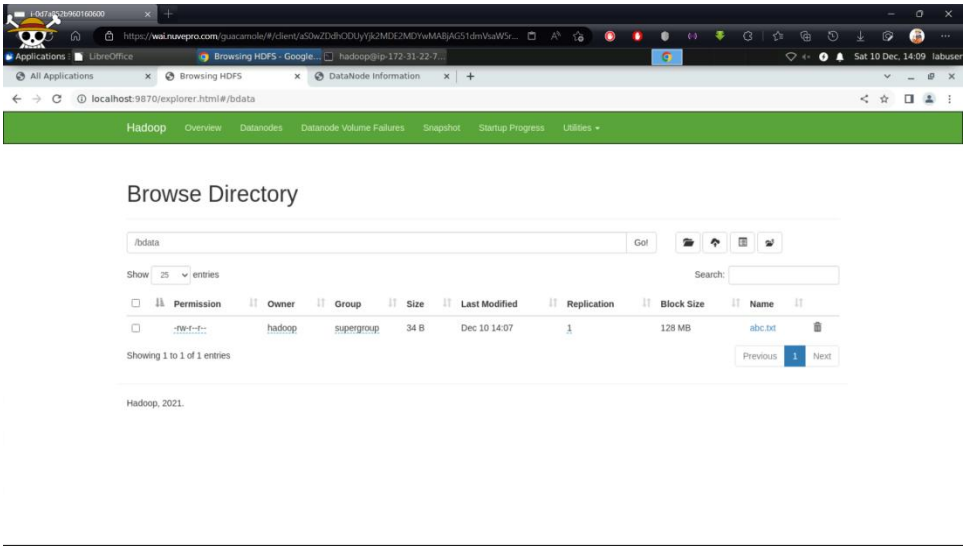
hdfs dfs -cat /hdemo/abc.txt



Step 6: Copy a file from source to destination

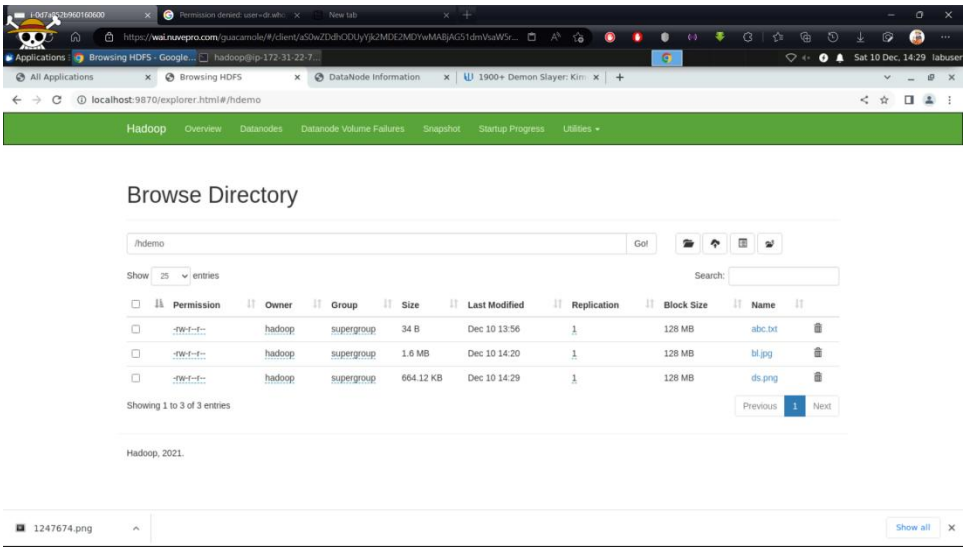
hdfs dfs -cp /hdemo/abc.txt /bdata

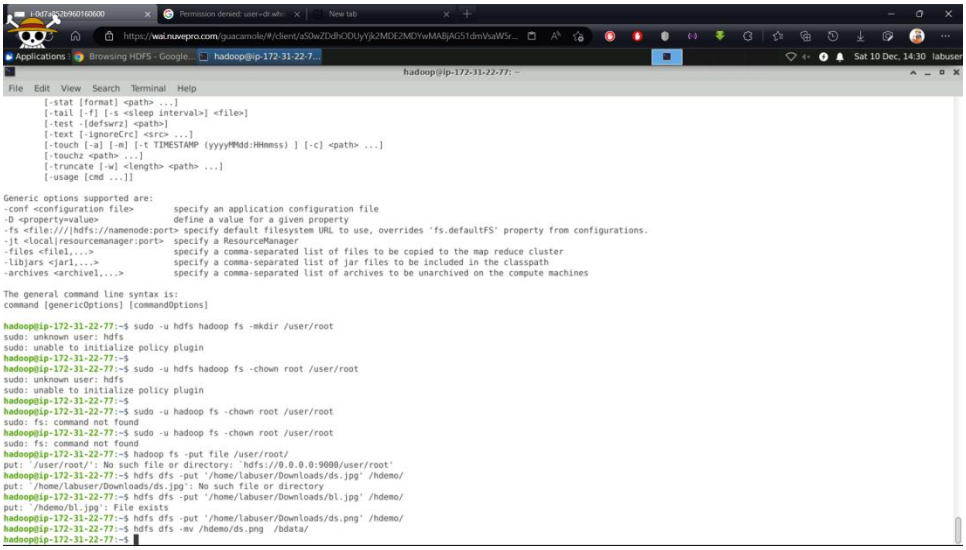


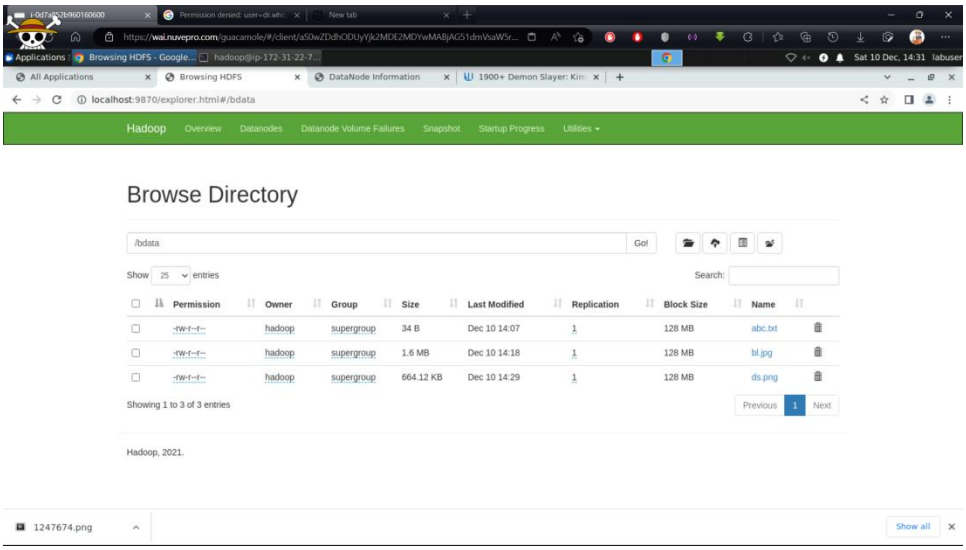


Step 7: Move file from source to destination.

hdfs dfs -mv /hdemo/ds.png /bdata/

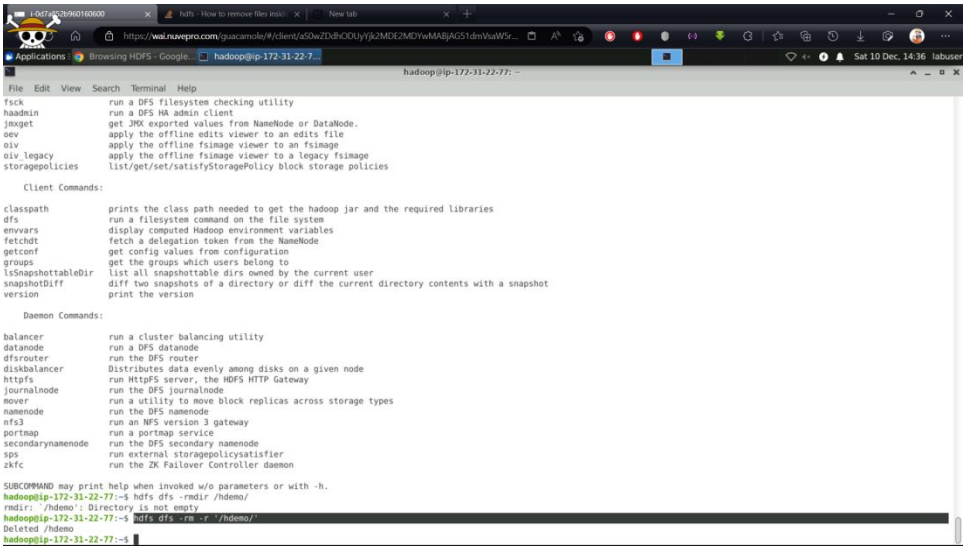


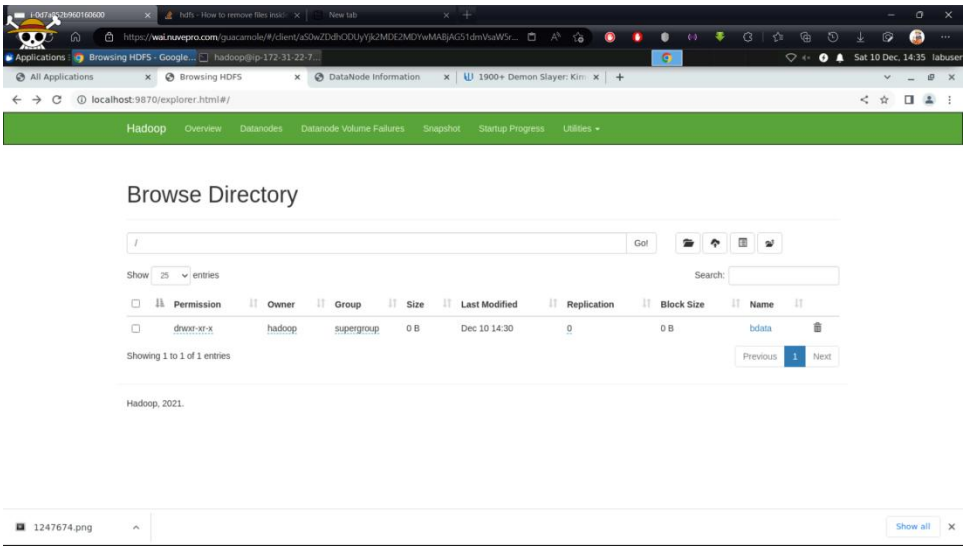




Step 8: Remove a file or directory in HDFS.

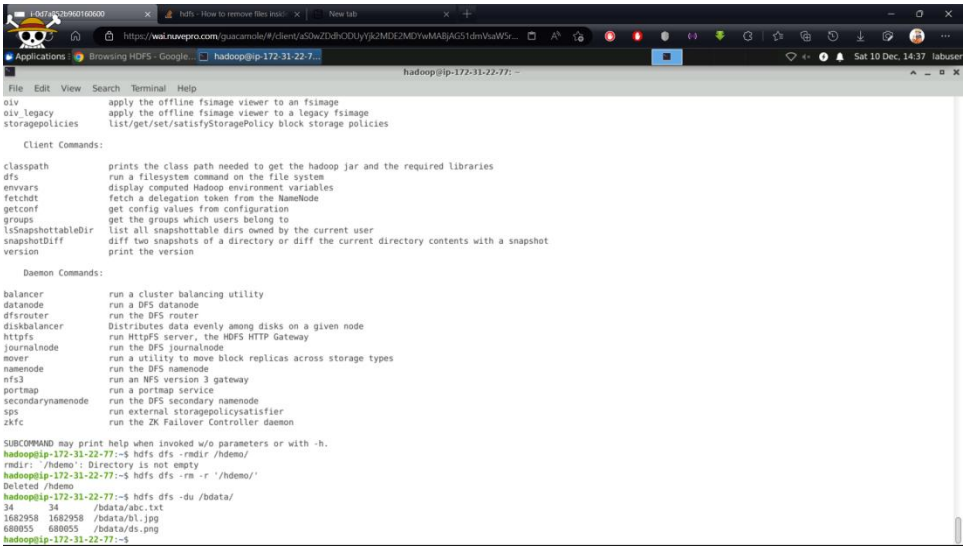
hdfs dfs -rm -r '/hdemo/'





Step 9: Display the aggregate length of a file.

hdfs dfs -du /bdata/


hadoop@ip-172-31-22-77: ~
File Edit View Search Terminal Help
oiv apply the offline fsimage viewer to an fsimage
oiv_legacy apply the offline fsimage viewer to a legacy fsimage
storagepolicies list/get/set/satisfyStoragePolicy block storage policies
Client Commands:
classpath prints the class path needed to get the hadoop jar and the required libraries
dfs run a filesystem command on the file system
envvars display computed Hadoop environment variables
fetchdt fetch a delegation token from the NameNode
getconf get config values from configuration
groups get the groups which users belong to
lsSnapshottableDir list all snapshottable dirs owned by the current user
snapshotDiff diff two snapshots of a directory or diff the current directory contents with a snapshot
version print the version
Daemon Commands:
balancer run a cluster balancing utility
datanode run a DFS datanode
dfsrouter run the DFS router
diskbalancer Distributes data evenly among disks on a given node
httpfs run HttpFS server, the HDFS HTTP Gateway
journalnode run the DFS journalnode
mover run a utility to move block replicas across storage types
namenode run the DFS namenode
nfs3 run an NFS version 3 gateway
portmap run a portmap service
secondarynamenode run the DFS secondary namenode
sps run external storagepolicysatisfier
zkfc run the ZK Failover Controller daemon
SUBCOMMAND may print help when invoked w/o parameters or with -h.
hadoop@ip-172-31-22-77:~$ hdfs dfs -rmdir /hdemo/
rmdir: `/hdemo': Directory is not empty
hadoop@ip-172-31-22-77:~$ hdfs dfs -rm -r '/hdemo/'
Deleted /hdemo
hadoop@ip-172-31-22-77:~$ hdfs dfs -du /bdata/
34 34 /bdata/abc.txt
1682958 1682958 /bdata/bl.jpg
680055 680055 /bdata/ds.png
hadoop@ip-172-31-22-77:~$

LAB EXCERSICE -3

WORD COUNT

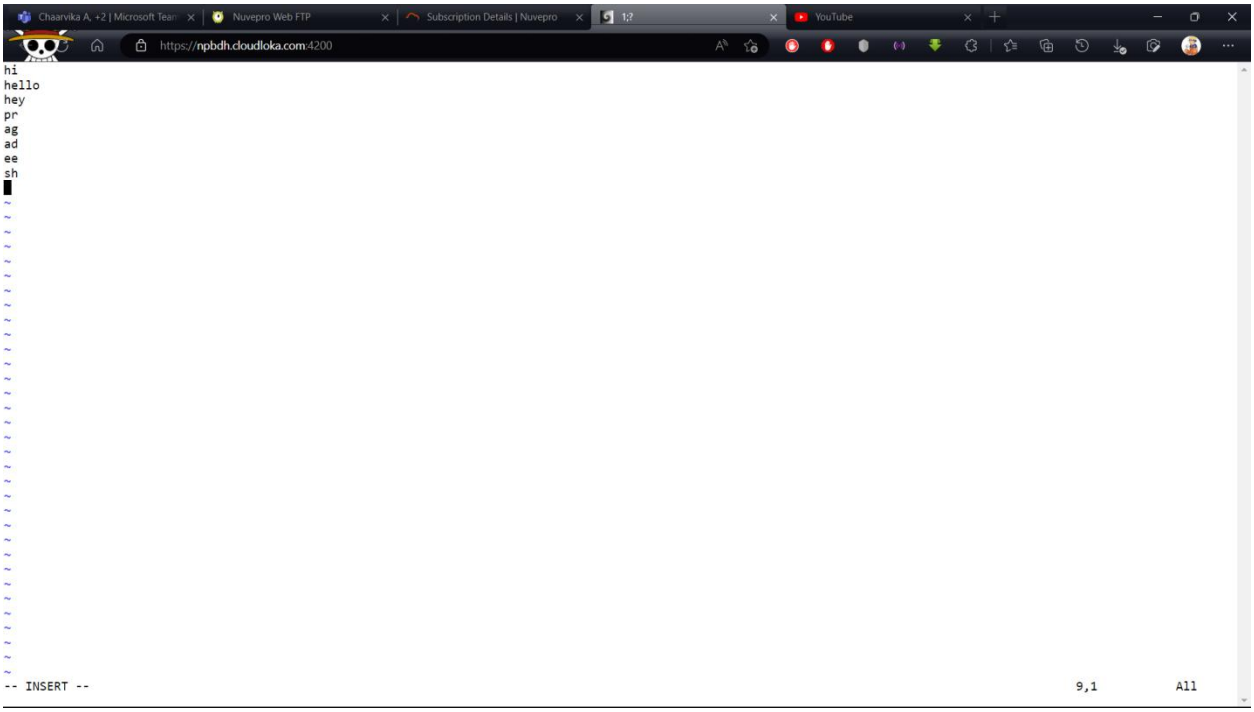
STEP1-LOGIN TO THE WEB SHELL USING THE USERNAME AND PASWORD

STEP2-CREATE A FOLDER/ DIRECTORY FOR weather IN HOME

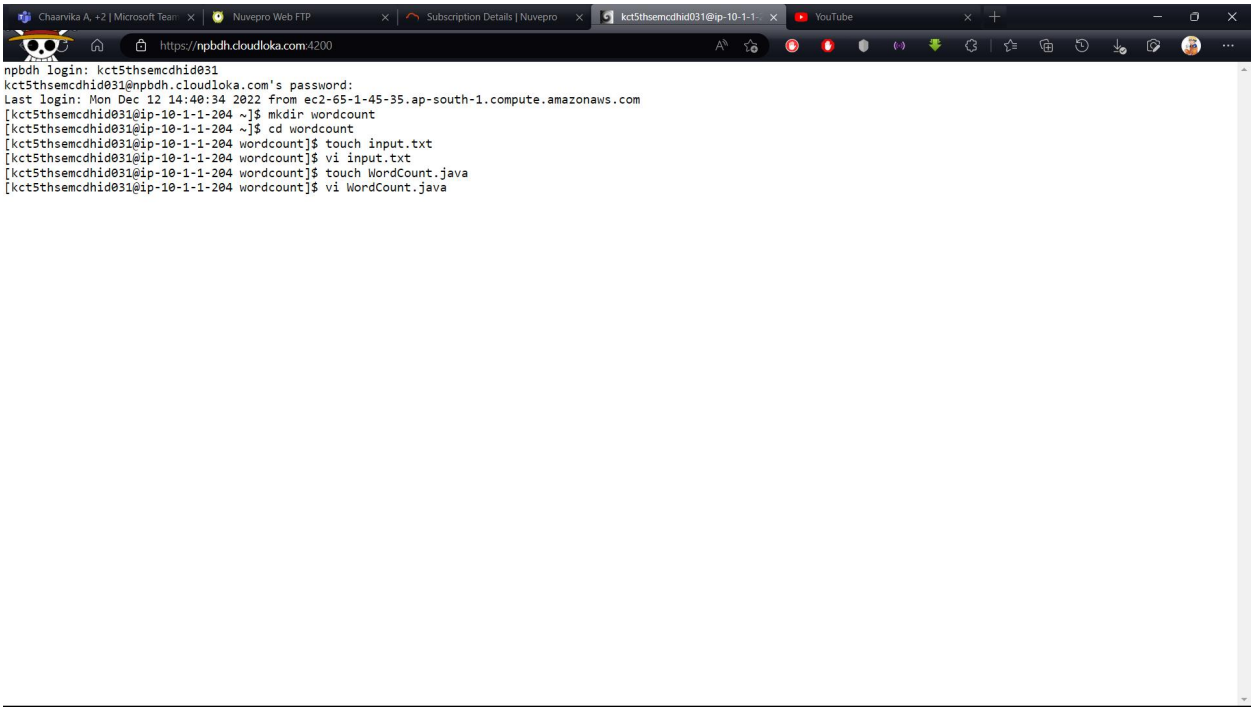
STEP3-MOVE TO THAT FOLDER/DIRECTORY

STEP4-CREATE A input.txt FILE INSIDE THE weather FOLDER

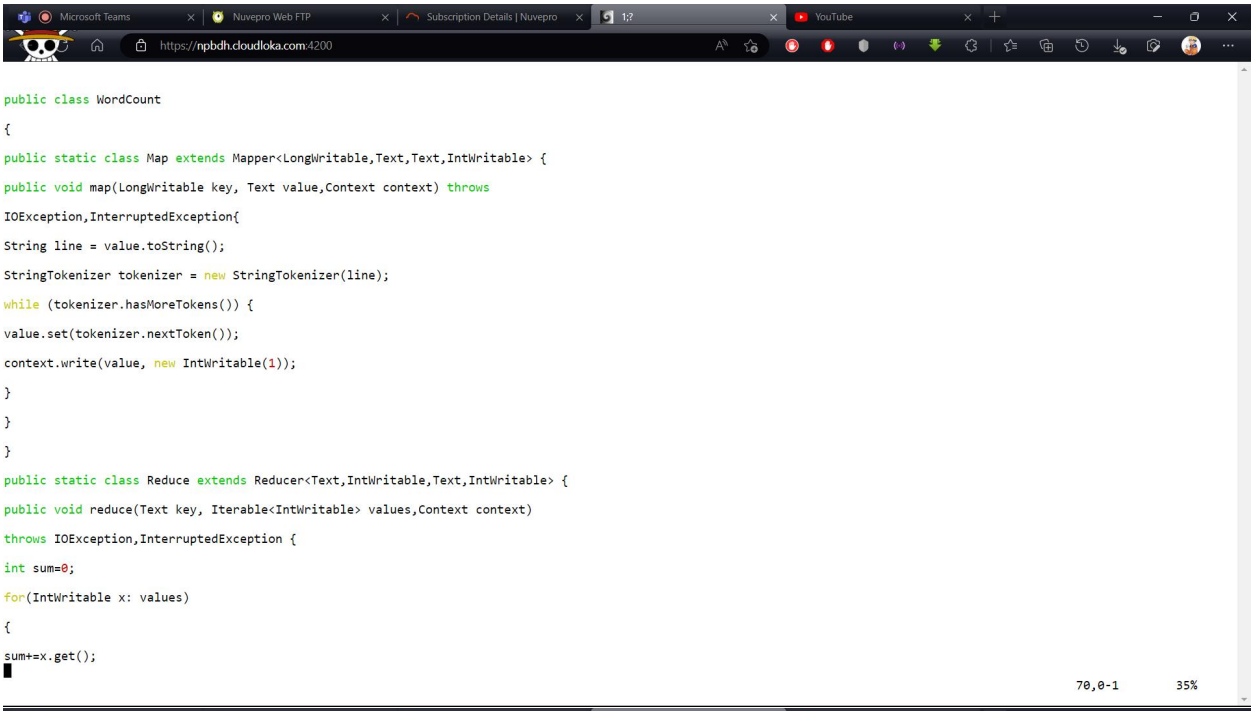
STEP5-PASTE THE INPUT IN THE INPUT FILE

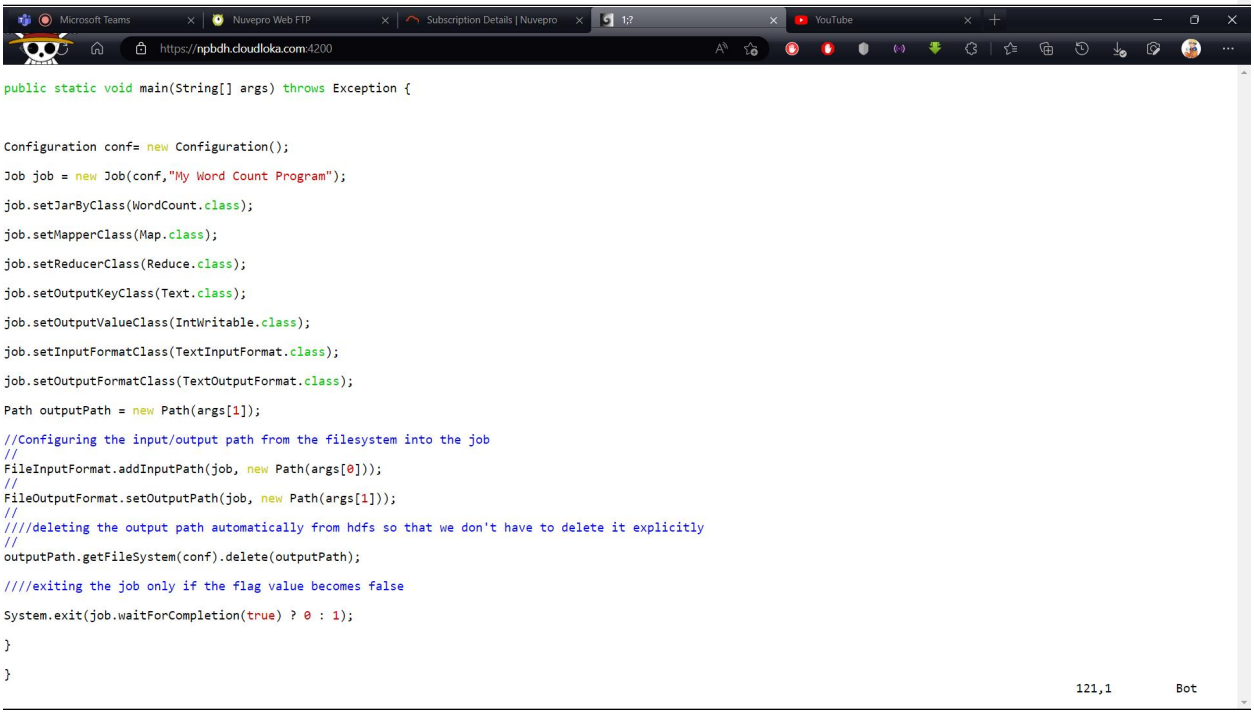


STEP6-CREATE A WORDCOUNT JAVA FILE



STEP7-PASTE THE SOURCE CODE IN THE FILE





SOURCE CODE:

import java.io.IOException;

```
import java.util.StringTokenizer;

import org.apache.hadoop.io.IntWritable;

import org.apache.hadoop.io.LongWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapreduce.Mapper;

import org.apache.hadoop.mapreduce.Reducer;

import org.apache.hadoop.conf.Configuration;

import org.apache.hadoop.mapreduce.Job;

import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;

import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;

import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

import org.apache.hadoop.fs.Path;
```

```
public class WordCount

{

public static class Map extends Mapper<LongWritable,Text,Text,IntWritable> {

public void map(LongWritable key, Text value,Context context) throws

IOException,InterruptedException{

String line = value.toString();

StringTokenizer tokenizer = new StringTokenizer(line);

while (tokenizer.hasMoreTokens()) {

value.set(tokenizer.nextToken());

context.write(value, new IntWritable(1));

}

}

}
```

```
public static class Reduce extends Reducer<Text,IntWritable,Text,IntWritable> {

public void reduce(Text key, Iterable<IntWritable> values,Context context)

throws IOException,InterruptedException {

int sum=0;

for(IntWritable x: values)

{

sum+=x.get();

}

context.write(key, new IntWritable(sum));

}

}



public static void main(String[] args) throws Exception {
```

```
Configuration conf= new Configuration();

Job job = new Job(conf,"My Word Count Program");

job.setJarByClass(WordCount.class);

job.setMapperClass(Map.class);

job.setReducerClass(Reduce.class);

job.setOutputKeyClass(Text.class);

job.setOutputValueClass(IntWritable.class);

job.setInputFormatClass(TextInputFormat.class);

job.setOutputFormatClass(TextOutputFormat.class);

Path outputPath = new Path(args[1]);

//Configuring the input/output path from the filesystem into the job
//
FileInputFormat.addInputPath(job, new Path(args[0]));
//
FileOutputFormat.setOutputPath(job, new Path(args[1]));
//
```

////deleting the output path automatically from hdfs so that we don't have to delete it explicitly

//

outputPath.getFileSystem(conf).delete(outputPath);

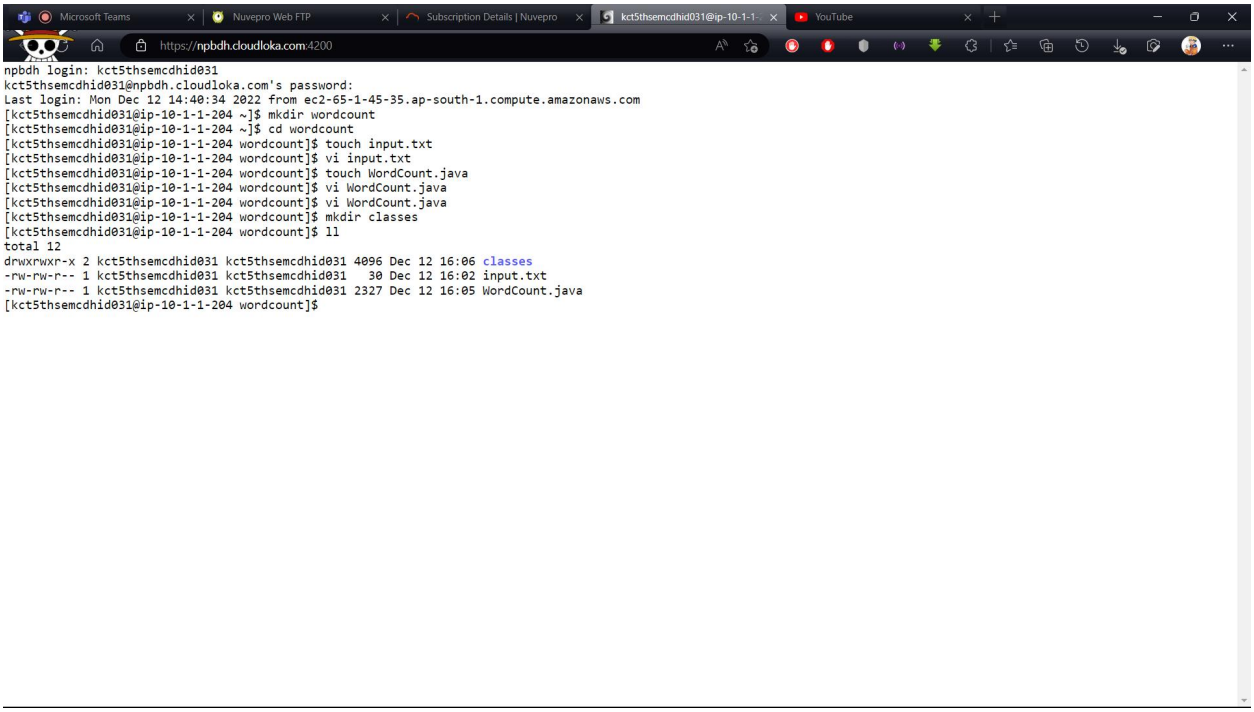
////exiting the job only if the flag value becomes false

System.exit(job.waitForCompletion(true) ? 0 : 1);

}

}

STEP8-CREATE A DIRECTORY/ FOLDER CLASSES INSIDE THE WORDCOUNT FOLDER



STEP9-SET THE PATH FOR JAVA FILE

COMMAND: export HADOOP_CLASSPATH=$(hadoop classpath)

echo $HADOOP_CLASSPATH

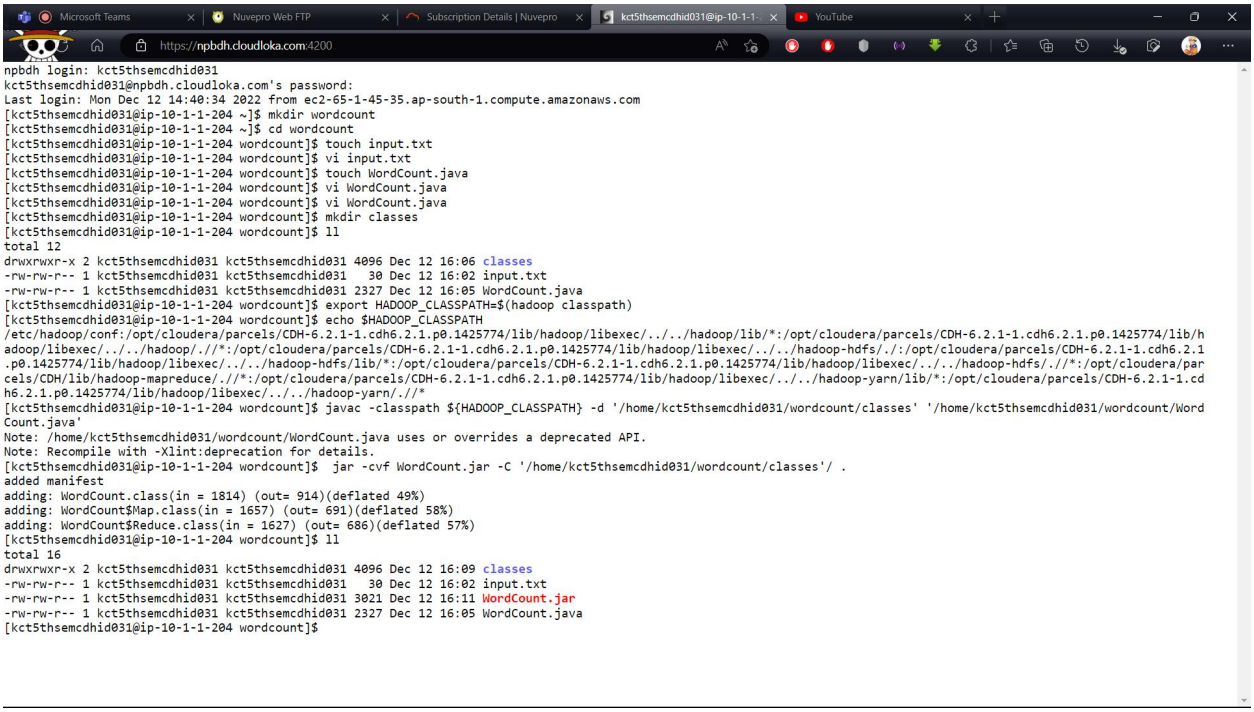
STEP10-COMPILE THE JAVAFILE

COMMAND: javac -classpath ${HADOOP_CLASSPATH} -d ‘/home/<NAME>/wordcount/classes’ ‘/home/<NAME>/wordcount/WordCount.java’

STEP11-CREATE A JAR FILE

COMMAND: jar -cvf WordCount.jar -C ‘/home/<NAME>/wordcount/classes’/ .

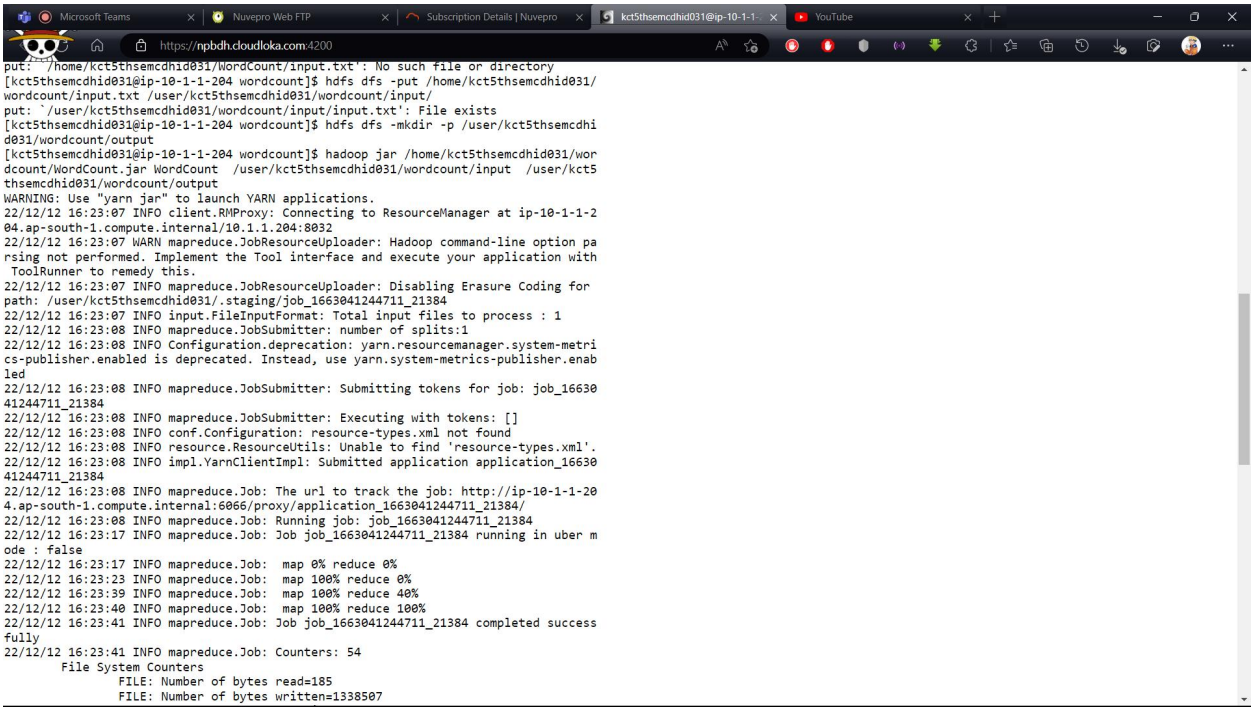
STEP12-CREATE A DIRECTORY IN HADOOP



```
npbdh login: kct5thsemcdhid031
kct5thsemcdhid031@npbdh.cloudloka.com's password:
Last login: Mon Dec 12 14:40:34 2022 from ec2-65-1-45-35.ap-south-1.compute.amazonaws.com
[kct5thsemcdhid031@ip-10-1-1-204 ~]$ mkdir wordcount
[kct5thsemcdhid031@ip-10-1-1-204 ~]$ cd wordcount
[kct5thsemcdhid031@ip-10-1-1-204 wordcount]$ touch input.txt
[kct5thsemcdhid031@ip-10-1-1-204 wordcount]$ vi input.txt
[kct5thsemcdhid031@ip-10-1-1-204 wordcount]$ touch WordCount.java
[kct5thsemcdhid031@ip-10-1-1-204 wordcount]$ vi WordCount.java
[kct5thsemcdhid031@ip-10-1-1-204 wordcount]$ vi WordCount.java
[kct5thsemcdhid031@ip-10-1-1-204 wordcount]$ mkdir classes
[kct5thsemcdhid031@ip-10-1-1-204 wordcount]$ ll
total 12
drwxrwxr-x 2 kct5thsemcdhid031 kct5thsemcdhid031 4096 Dec 12 16:06 classes
-rw-rw-r-- 1 kct5thsemcdhid031 kct5thsemcdhid031   30 Dec 12 16:02 input.txt
-rw-rw-r-- 1 kct5thsemcdhid031 kct5thsemcdhid031 2327 Dec 12 16:05 WordCount.java
[kct5thsemcdhid031@ip-10-1-1-204 wordcount]$ export HADOOP_CLASSPATH=$(hadoop classpath)
[kct5thsemcdhid031@ip-10-1-1-204 wordcount]$ echo $HADOOP_CLASSPATH
/etc/hadoop/conf:/opt/cloudera/parcels/CDH-6.2.1-1.cdh6.2.1.p0.1425774/lib/hadoop/libexec/../../hadoop/lib/*:/opt/cloudera/parcels/CDH-6.2.1-1.cdh6.2.1.p0.1425774/lib/h
adoop/libexec/../../hadoop/.//*:/opt/cloudera/parcels/CDH-6.2.1-1.cdh6.2.1.p0.1425774/lib/hadoop/libexec/../../hadoop-hdfs/./:/opt/cloudera/parcels/CDH-6.2.1-1.cdh6.2.1
.p0.1425774/lib/hadoop/libexec/../../hadoop-hdfs/lib/*:/opt/cloudera/parcels/CDH-6.2.1-1.cdh6.2.1.p0.1425774/lib/hadoop/libexec/../../hadoop-hdfs/.//*:/opt/cloudera/par
cels/CDH/lib/hadoop-mapreduce/.//*:/opt/cloudera/parcels/CDH-6.2.1-1.cdh6.2.1.p0.1425774/lib/hadoop/libexec/../../hadoop-yarn/lib/*:/opt/cloudera/parcels/CDH-6.2.1-1.cd
h6.2.1.p0.1425774/lib/hadoop/libexec/../../hadoop-yarn/.//*
[kct5thsemcdhid031@ip-10-1-1-204 wordcount]$ javac -classpath ${HADOOP_CLASSPATH} -d '/home/kct5thsemcdhid031/wordcount/classes' '/home/kct5thsemcdhid031/wordcount/Word
Count.java'
Note: /home/kct5thsemcdhid031/wordcount/WordCount.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
[kct5thsemcdhid031@ip-10-1-1-204 wordcount]$  jar -cvf WordCount.jar -C '/home/kct5thsemcdhid031/wordcount/classes'/ .
added manifest
adding: WordCount.class(in = 1814) (out= 914)(deflated 49%)
adding: WordCount$Map.class(in = 1657) (out= 691)(deflated 58%)
adding: WordCount$Reduce.class(in = 1627) (out= 686)(deflated 57%)
[kct5thsemcdhid031@ip-10-1-1-204 wordcount]$ ll
total 16
drwxrwxr-x 2 kct5thsemcdhid031 kct5thsemcdhid031 4096 Dec 12 16:09 classes
-rw-rw-r-- 1 kct5thsemcdhid031 kct5thsemcdhid031   30 Dec 12 16:02 input.txt
-rw-rw-r-- 1 kct5thsemcdhid031 kct5thsemcdhid031 3021 Dec 12 16:11 WordCount.jar
-rw-rw-r-- 1 kct5thsemcdhid031 kct5thsemcdhid031 2327 Dec 12 16:05 WordCount.java
[kct5thsemcdhid031@ip-10-1-1-204 wordcount]$
```

STEP13-PUT THE INPUT FILE IN LOCAL SYSTEM TO HADOOP DIRECTORY

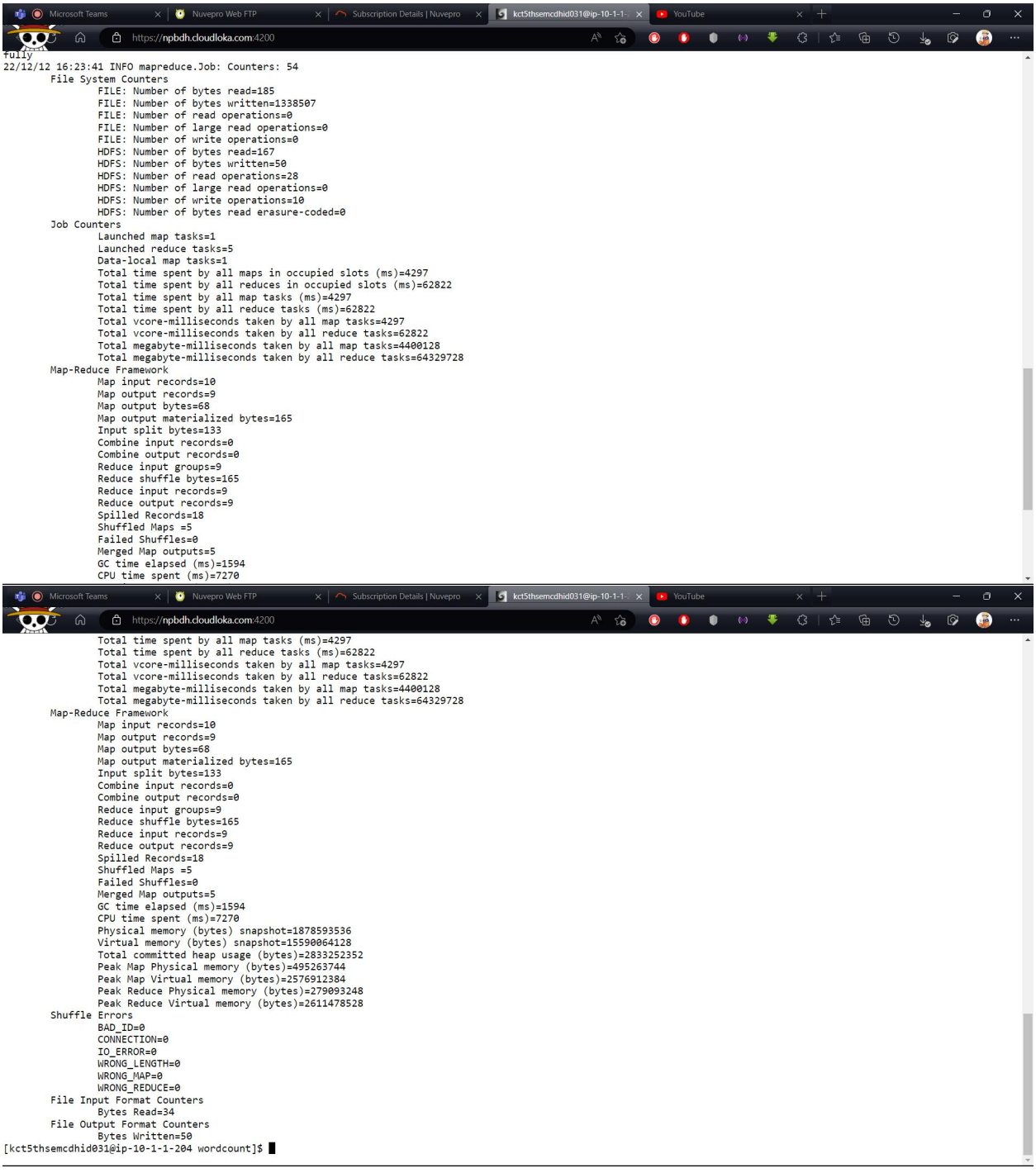
STEP14-CREATE A OUTPUT DIRECTORY IN HDFS INSIDE THE WORDCOUNT



```
put: `/home/kct5thsemcdhid031/WordCount/input.txt': No such file or directory
[kct5thsemcdhid031@ip-10-1-1-204 wordcount]$ hdfs dfs -put /home/kct5thsemcdhid031/
wordcount/input.txt /user/kct5thsemcdhid031/wordcount/input/
put: `/user/kct5thsemcdhid031/wordcount/input/input.txt': File exists
[kct5thsemcdhid031@ip-10-1-1-204 wordcount]$ hdfs dfs -mkdir -p /user/kct5thsemcdhi
d031/wordcount/output
[kct5thsemcdhid031@ip-10-1-1-204 wordcount]$ hadoop jar /home/kct5thsemcdhid031/wor
dcount/WordCount.jar WordCount  /user/kct5thsemcdhid031/wordcount/input  /user/kct5
thsemcdhid031/wordcount/output
WARNING: Use "yarn jar" to launch YARN applications.
22/12/12 16:23:07 INFO client.RMProxy: Connecting to ResourceManager at ip-10-1-1-2
04.ap-south-1.compute.internal/10.1.1.204:8032
22/12/12 16:23:07 WARN mapreduce.JobResourceUploader: Hadoop command-line option pa
rsing not performed. Implement the Tool interface and execute your application with
 ToolRunner to remedy this.
22/12/12 16:23:07 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for
path: /user/kct5thsemcdhid031/.staging/job_1663041244711_21384
22/12/12 16:23:07 INFO input.FileInputFormat: Total input files to process : 1
22/12/12 16:23:08 INFO mapreduce.JobSubmitter: number of splits:1
22/12/12 16:23:08 INFO Configuration.deprecation: yarn.resourcemanager.system-metri
cs-publisher.enabled is deprecated. Instead, use yarn.system-metrics-publisher.enab
led
22/12/12 16:23:08 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_16630
41244711_21384
22/12/12 16:23:08 INFO mapreduce.JobSubmitter: Executing with tokens: []
22/12/12 16:23:08 INFO conf.Configuration: resource-types.xml not found
22/12/12 16:23:08 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
22/12/12 16:23:08 INFO impl.YarnClientImpl: Submitted application application_16630
41244711_21384
22/12/12 16:23:08 INFO mapreduce.Job: The url to track the job: http://ip-10-1-1-20
4.ap-south-1.compute.internal:6066/proxy/application_1663041244711_21384/
22/12/12 16:23:08 INFO mapreduce.Job: Running job: job_1663041244711_21384
22/12/12 16:23:17 INFO mapreduce.Job: Job job_1663041244711_21384 running in uber m
ode : false
22/12/12 16:23:17 INFO mapreduce.Job:  map 0% reduce 0%
22/12/12 16:23:23 INFO mapreduce.Job:  map 100% reduce 0%
22/12/12 16:23:39 INFO mapreduce.Job:  map 100% reduce 40%
22/12/12 16:23:40 INFO mapreduce.Job:  map 100% reduce 100%
22/12/12 16:23:41 INFO mapreduce.Job: Job job_1663041244711_21384 completed success
fully
22/12/12 16:23:41 INFO mapreduce.Job: Counters: 54
        File System Counters
                FILE: Number of bytes read=185
                FILE: Number of bytes written=1338507
```

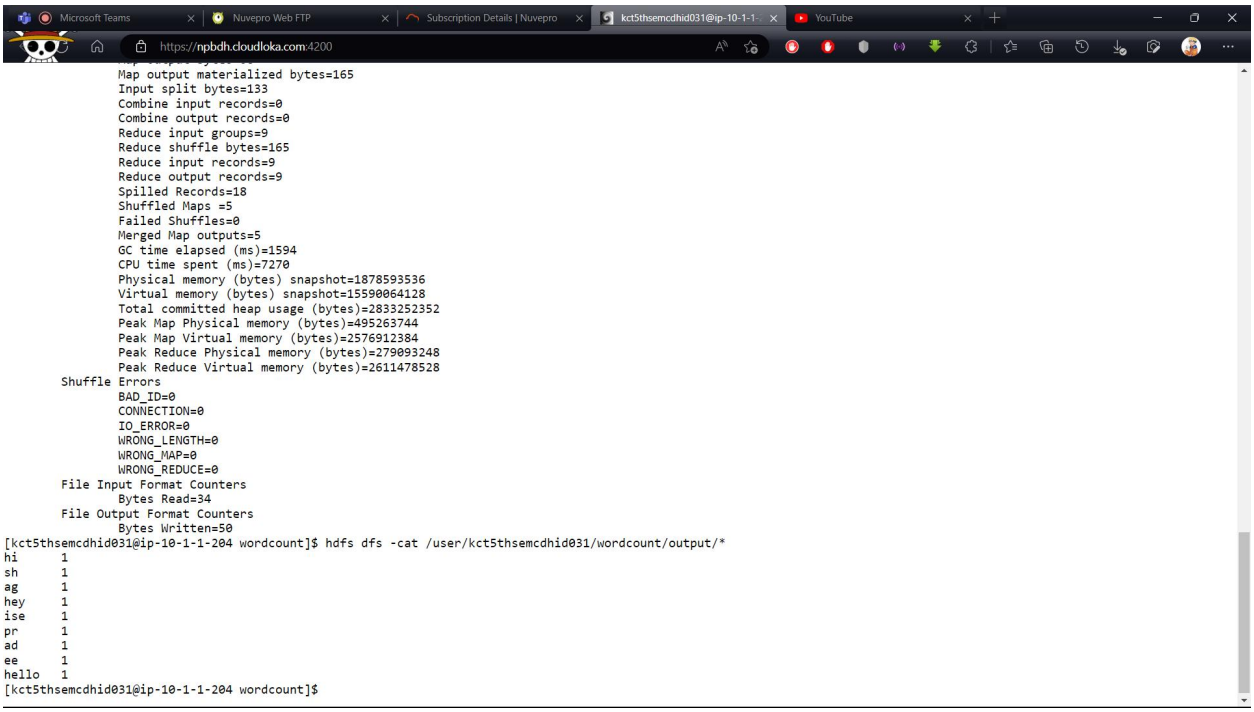
STEP15-RUN THE MAP REDUCE PROGRAM

COMMAND: hadoop jar /home/<NAME>/wordcount/WordCount.jar’ WordCount /user/<NAME>/wordcount/input /user/<NAME>/wordcount/output



STEP16-VERIFY THE OUTPUT:

COMMAND: hdfs dfs -cat /user/<Name>/wordcount/output/*



LAB EXCERSICE -4

WEATHER

AIM:
Implement a Map Reduce Program to analyse time-temperature statistics and generate report with max/min temperature

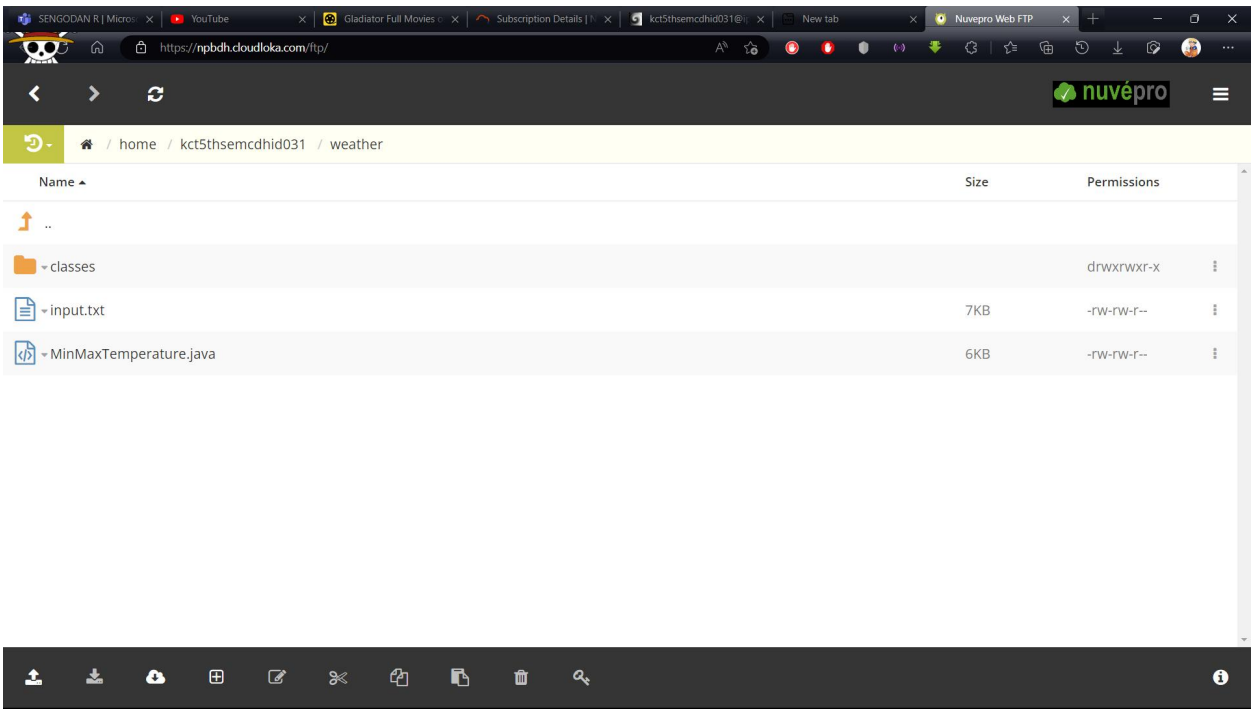
PROCEDURE:

STEP1-LOGIN TO THE WEB SHELL USING THE USERNAME AND PASWORD

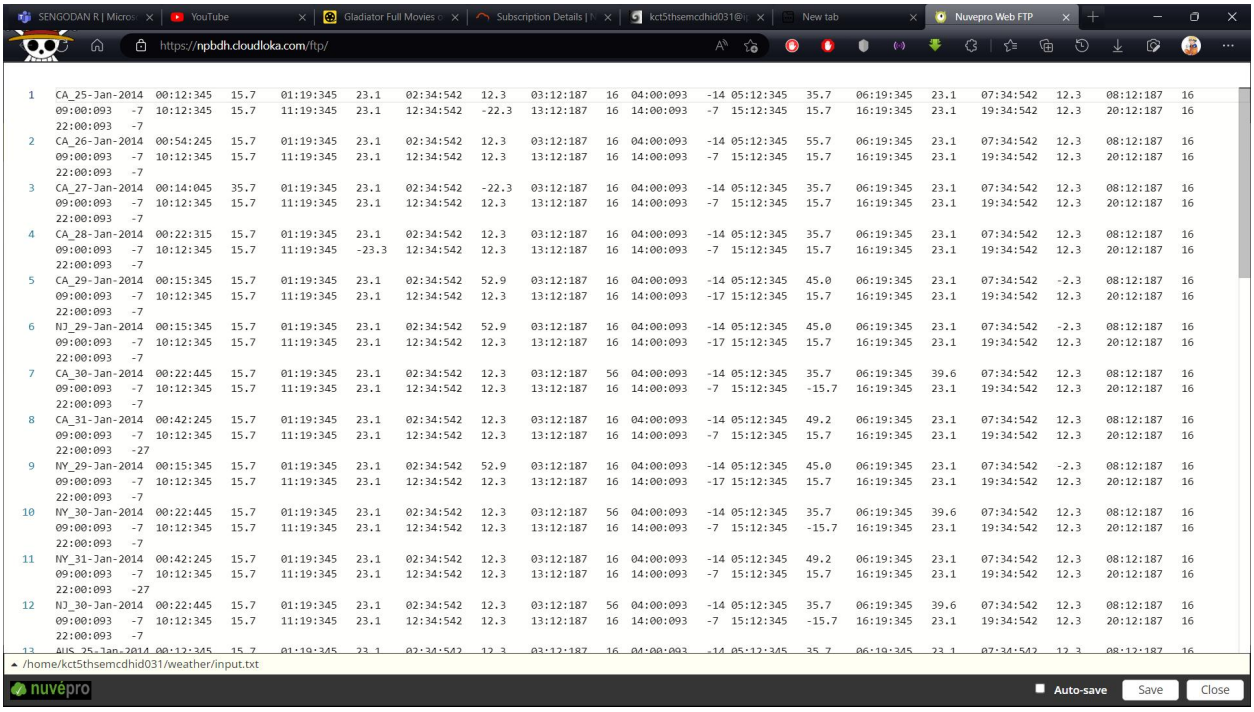
STEP2-CREATE A FOLDER/ DIRECTORY FOR WEATHERIN HOME

STEP3-MOVE TO THAT FOLDER/DIRECTORY

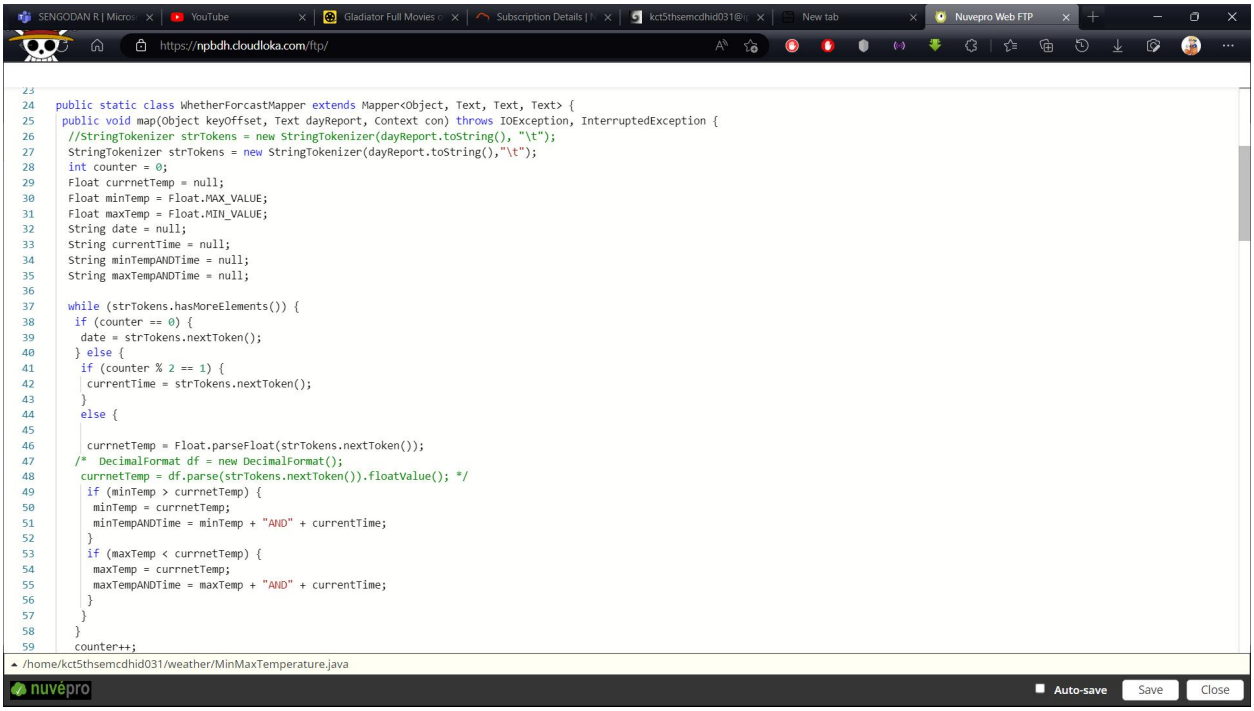
STEP4-CREATE A INPUT FILE AND JAVA FILE INSIDE THE WORDCOUNT FOLDER

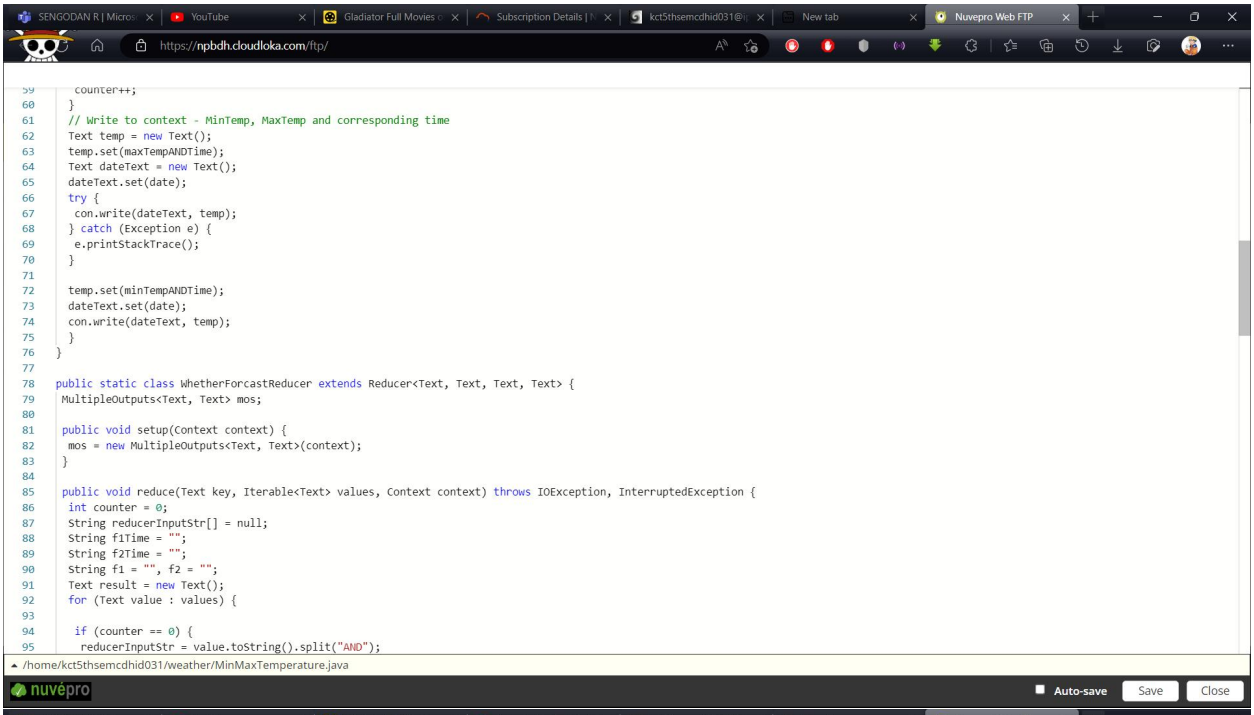


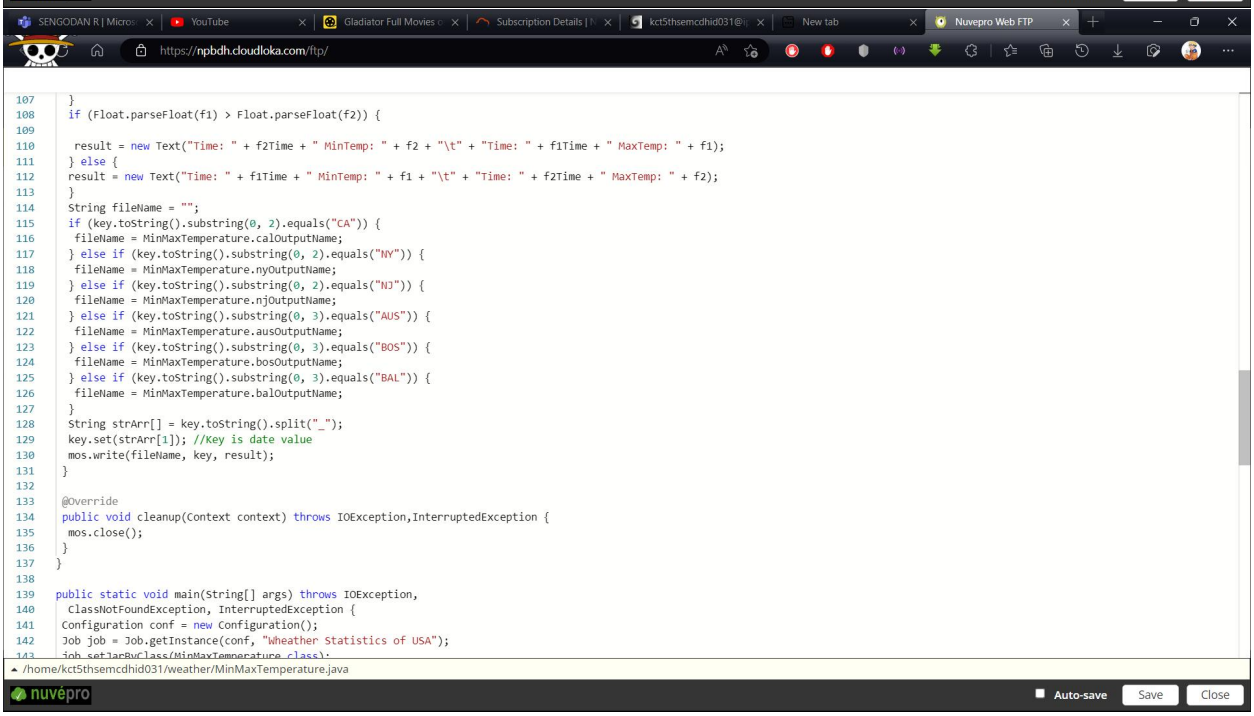
STEP5-PASTE THE INPUT IN THE input.txt

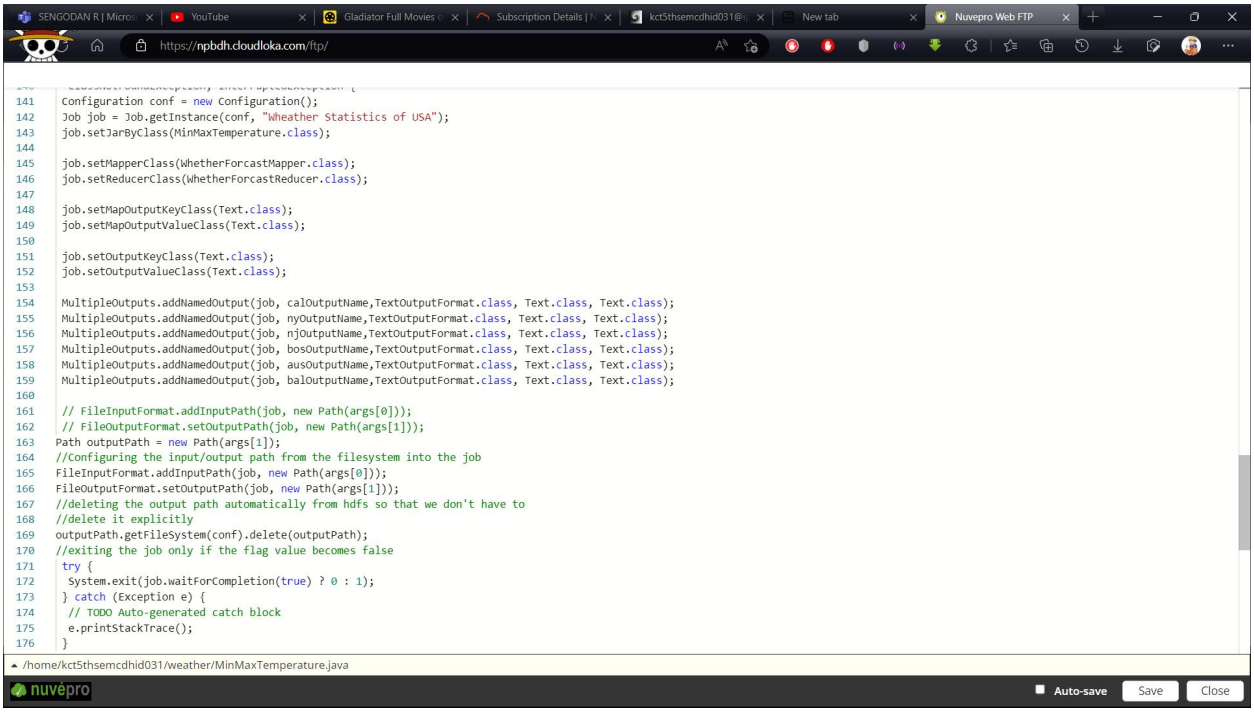


STEP6-PASTE THE SOURCE CODE IN THE FILE




SENGODAN R | Micros
YouTube
Gladiator Full Movies
Subscription Details |
kct5thsemcdhid031@i
New tab
Nuvepro Web FTP
https://npbdh.cloudloka.com/ftp/
counter++;
}
// Write to context - MinTemp, MaxTemp and corresponding time
Text temp = new Text();
temp.set(maxTempANDTime);
Text dateText = new Text();
dateText.set(date);
try {
con.write(dateText, temp);
} catch (Exception e) {
e.printStackTrace();
}
temp.set(minTempANDTime);
dateText.set(date);
con.write(dateText, temp);
}
}
public static class WhetherForcastReducer extends Reducer<Text, Text, Text, Text> {
MultipleOutputs<Text, Text> mos;
public void setup(Context context) {
mos = new MultipleOutputs<Text, Text>(context);
}
public void reduce(Text key, Iterable<Text> values, Context context) throws IOException, InterruptedException {
int counter = 0;
String reducerInputStr[] = null;
String f1Time = "";
String f2Time = "";
String f1 = "", f2 = "";
Text result = new Text();
for (Text value : values) {
if (counter == 0) {
reducerInputStr = value.toString().split("AND");
/home/kct5thsemcdhid031/weather/MinMaxTemperature.java
nuvepro
Auto-save
Save
Close


SENGODAN R | Micros
YouTube
Gladiator Full Movies
Subscription Details |
kct5thsemcdhid031@i
New tab
Nuvepro Web FTP
https://npbdh.cloudloka.com/ftp/
}
if (Float.parseFloat(f1) > Float.parseFloat(f2)) {
result = new Text("Time: " + f2Time + " MinTemp: " + f2 + "\t" + "Time: " + f1Time + " MaxTemp: " + f1);
} else {
result = new Text("Time: " + f1Time + " MinTemp: " + f1 + "\t" + "Time: " + f2Time + " MaxTemp: " + f2);
}
String fileName = "";
if (key.toString().substring(0, 2).equals("CA")) {
fileName = MinMaxTemperature.calOutputName;
} else if (key.toString().substring(0, 2).equals("NY")) {
fileName = MinMaxTemperature.nyOutputName;
} else if (key.toString().substring(0, 2).equals("NJ")) {
fileName = MinMaxTemperature.njOutputName;
} else if (key.toString().substring(0, 3).equals("AUS")) {
fileName = MinMaxTemperature.ausOutputName;
} else if (key.toString().substring(0, 3).equals("BOS")) {
fileName = MinMaxTemperature.bosOutputName;
} else if (key.toString().substring(0, 3).equals("BAL")) {
fileName = MinMaxTemperature.balOutputName;
}
String strArr[] = key.toString().split("_");
key.set(strArr[1]); //Key is date value
mos.write(fileName, key, result);
}
@Override
public void cleanup(Context context) throws IOException,InterruptedException {
mos.close();
}
}
public static void main(String[] args) throws IOException,
ClassNotFoundException, InterruptedException {
Configuration conf = new Configuration();
Job job = Job.getInstance(conf, "Wheather Statistics of USA");
/home/kct5thsemcdhid031/weather/MinMaxTemperature.java
nuvepro
Auto-save
Save
Close



SOURCE CODE:

```
 import java.io.IOException;
import java.util.StringTokenizer;
import java.text.DecimalFormat;

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

public class MinMaxTemperature {
public static String calOutputName = "California";
public static String nyOutputName = "Newyork";
public static String njOutputName = "Newjersy";
public static String ausOutputName = "Austin";
```

```
public static String bosOutputName = "Boston";
public static String balOutputName = "Baltimore";

public static class WhetherForcastMapper extends Mapper<Object, Text, Text, Text> {
public void map(Object keyOffset, Text dayReport, Context con) throws IOException,
InterruptedException {
//StringTokenizer strTokens = new StringTokenizer(dayReport.toString(), "\t");
StringTokenizer strTokens = new StringTokenizer(dayReport.toString(),"\t");
int counter = 0;
Float currnetTemp = null;
Float minTemp = Float.MAX_VALUE;
Float maxTemp = Float.MIN_VALUE;
String date = null;
String currentTime = null;
String minTempANDTime = null;
String maxTempANDTime = null;

while (strTokens.hasMoreElements()) {
if (counter == 0) {
date = strTokens.nextToken();
} else {
if (counter % 2 == 1) {
currentTime = strTokens.nextToken();
}
else {
currnetTemp = Float.parseFloat(strTokens.nextToken());
/* DecimalFormat df = new DecimalFormat();
currnetTemp = df.parse(strTokens.nextToken()).floatValue(); */
if (minTemp > currnetTemp) {
minTemp = currnetTemp;
minTempANDTime = minTemp + "AND" + currentTime;
}
if (maxTemp < currnetTemp) {
maxTemp = currnetTemp;
maxTempANDTime = maxTemp + "AND" + currentTime;
}
}
}
counter++;
}
// Write to context - MinTemp, MaxTemp and corresponding time
Text temp = new Text();
temp.set(maxTempANDTime);
Text dateText = new Text();
```

```
dateText.set(date);
try {
con.write(dateText, temp);
} catch (Exception e) {
e.printStackTrace();
}
temp.set(minTempANDTime);
dateText.set(date);
con.write(dateText, temp);
}
}

public static class WhetherForcastReducer extends Reducer<Text, Text, Text, Text> {
MultipleOutputs<Text, Text> mos;

public void setup(Context context) {
mos = new MultipleOutputs<Text, Text>(context);
}

public void reduce(Text key, Iterable<Text> values, Context context) throws IOException,
InterruptedException {
int counter = 0;
String reducerInputStr[] = null;
String f1Time = "";
String f2Time = "";
String f1 = "", f2 = "";
Text result = new Text();
for (Text value : values) {
if (counter == 0) {
reducerInputStr = value.toString().split("AND");
f1 = reducerInputStr[0];
f1Time = reducerInputStr[1];
}

else {
reducerInputStr = value.toString().split("AND");
f2 = reducerInputStr[0];
f2Time = reducerInputStr[1];
}

counter = counter + 1;
}
if (Float.parseFloat(f1) > Float.parseFloat(f2)) {
```

```
result = new Text("Time: " + f2Time + " MinTemp: " + f2 + "\t" + "Time: " + f1Time + " MaxTemp:
" + f1);
} else {
result = new Text("Time: " + f1Time + " MinTemp: " + f1 + "\t" + "Time: " + f2Time + " MaxTemp:
" + f2);
}
String fileName = "";
if (key.toString().substring(0, 2).equals("CA")) {
fileName = MinMaxTemperature.calOutputName;
} else if (key.toString().substring(0, 2).equals("NY")) {
fileName = MinMaxTemperature.nyOutputName;
} else if (key.toString().substring(0, 2).equals("NJ")) {
fileName = MinMaxTemperature.njOutputName;
} else if (key.toString().substring(0, 3).equals("AUS")) {
fileName = MinMaxTemperature.ausOutputName;
} else if (key.toString().substring(0, 3).equals("BOS")) {
fileName = MinMaxTemperature.bosOutputName;
} else if (key.toString().substring(0, 3).equals("BAL")) {
fileName = MinMaxTemperature.balOutputName;
}
String strArr[] = key.toString().split("_");
key.set(strArr[1]); //Key is date value
mos.write(fileName, key, result);
}

@Override
public void cleanup(Context context) throws IOException,InterruptedException {
mos.close();
}
}

public static void main(String[] args) throws IOException,
ClassNotFoundException, InterruptedException {
Configuration conf = new Configuration();
Job job = Job.getInstance(conf, "Wheather Statistics of USA");
job.setJarByClass(MinMaxTemperature.class);

job.setMapperClass(WhetherForcastMapper.class);
job.setReducerClass(WhetherForcastReducer.class);

job.setMapOutputKeyClass(Text.class);
job.setMapOutputValueClass(Text.class);

job.setOutputKeyClass(Text.class);
```

```
job.setOutputValueClass(Text.class);

MultipleOutputs.addNamedOutput(job, calOutputName,TextOutputFormat.class, Text.class,
Text.class);
MultipleOutputs.addNamedOutput(job, nyOutputName,TextOutputFormat.class, Text.class,
Text.class);
MultipleOutputs.addNamedOutput(job, njOutputName,TextOutputFormat.class, Text.class,
Text.class);
MultipleOutputs.addNamedOutput(job, bosOutputName,TextOutputFormat.class, Text.class,
Text.class);
MultipleOutputs.addNamedOutput(job, ausOutputName,TextOutputFormat.class, Text.class,
Text.class);
MultipleOutputs.addNamedOutput(job, balOutputName,TextOutputFormat.class, Text.class,
Text.class);

// FileInputFormat.addInputPath(job, new Path(args[0]));
// FileOutputFormat.setOutputPath(job, new Path(args[1]));
Path outputPath = new Path(args[1]);
//Configuring the input/output path from the filesystem into the job
FileInputFormat.addInputPath(job, new Path(args[0]));
FileOutputFormat.setOutputPath(job, new Path(args[1]));
//deleting the output path automatically from hdfs so that we don't have to
//delete it explicitly
outputPath.getFileSystem(conf).delete(outputPath);
//exiting the job only if the flag value becomes false
try {
System.exit(job.waitForCompletion(true) ? 0 : 1);
} catch (Exception e) {
// TODO Auto-generated catch block
e.printStackTrace();
}
}

}
```

STEP7-CREATE A DIRECTORY/ FOLDER CLASSES INSIDE THE WEATHER FOLDER

STEP8-SET THE PATH FOR JAVA FILE

COMMAND: export HADOOP_CLASSPATH=$(hadoop classpath)

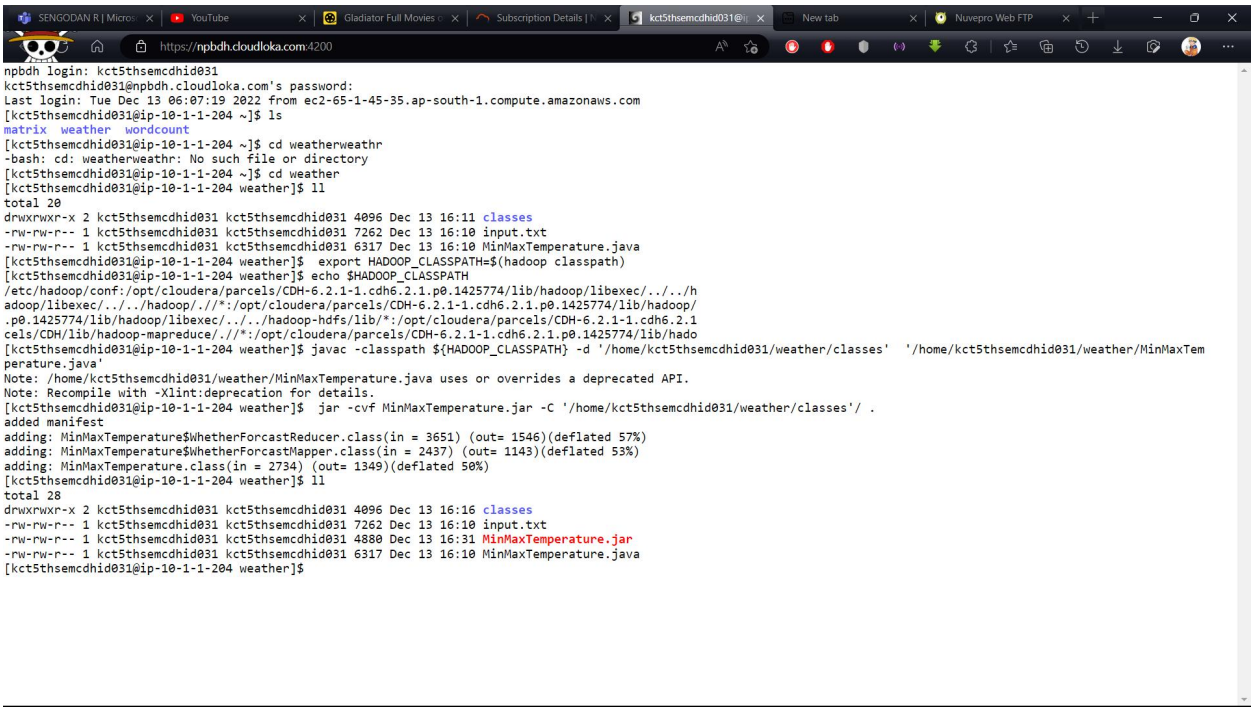
echo $HADOOP_CLASSPATH

STEP9-COMPILE THE JAVAFILE

COMMAND: javac -classpath ${HADOOP_CLASSPATH} -d '/home/<NAME>/wordcount/classes' '/home/<NAME>/wordcount/WordCount.java'

STEP10-CREATE A JAR FILE

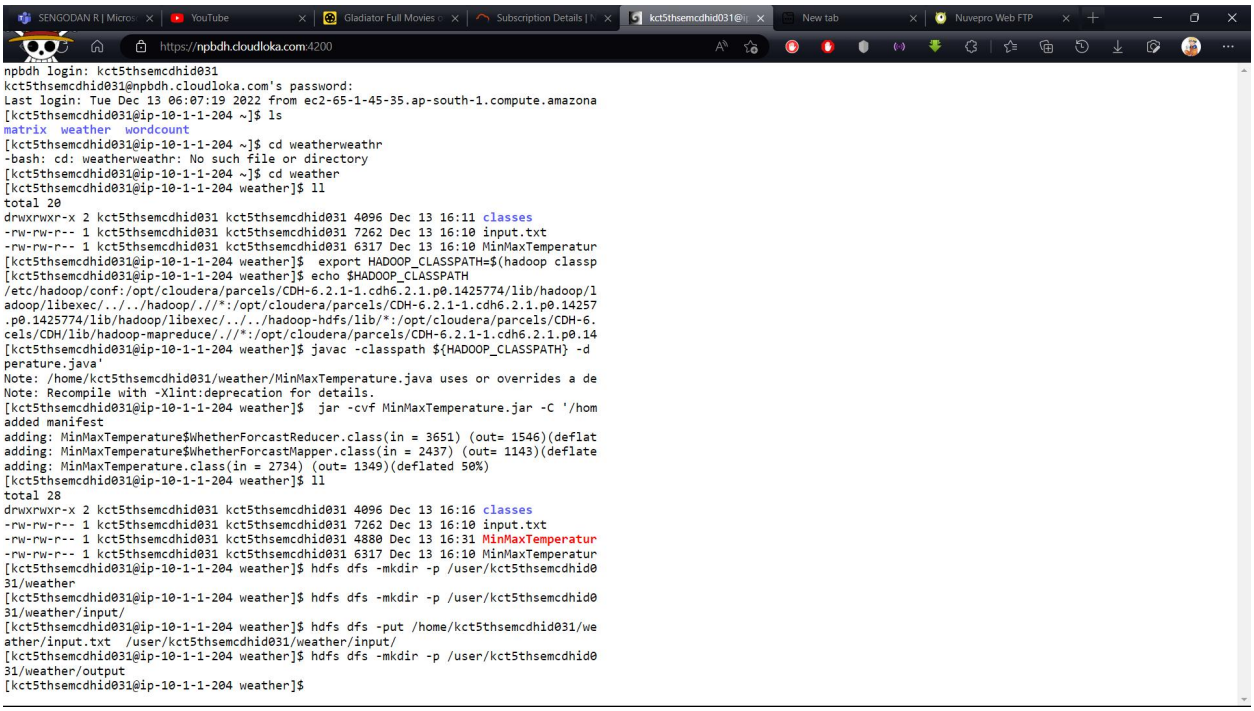
COMMAND: jar -cvf WordCount.jar -C '/home/<NAME>/wordcount/classes'/ .



STEP11-CREATE A DIRECTORY IN HADOOP

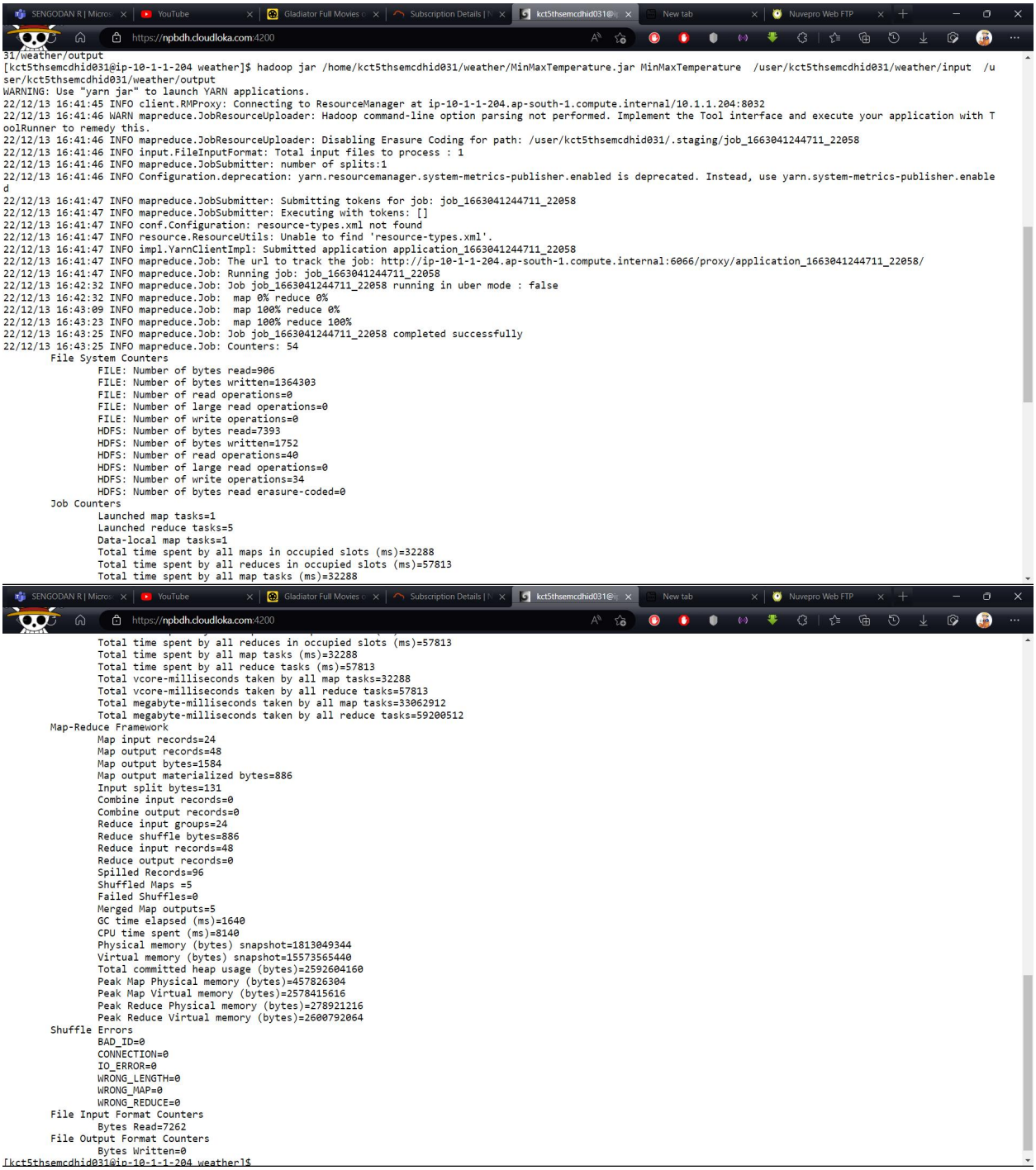
STEP12-CREATE AND PUT THE INPUT FILE IN LOCAL SYSTEM TO HADOOP DIRECTORY

STEP13-CREATE A OUTPUT DIRECTORY IN HDFS INSIDE THE WEATHER



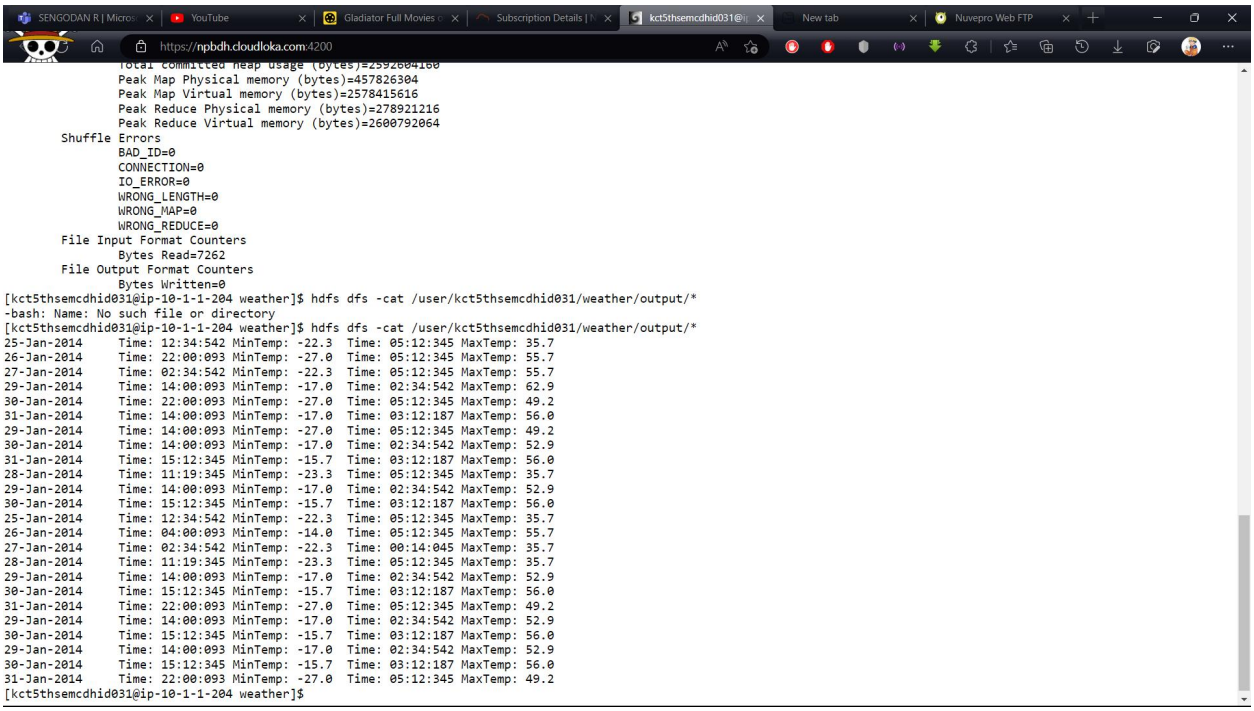
STEP14-RUN THE MAP REDUCE PROGRAM

COMMAND: hadoop jar /home/<NAME>/wordcount/WordCount.jar' WordCount /user/<NAME>/wordcount/input /user/<NAME>/wordcount/output



STEP15-VERIFY THE OUTPUT:

COMMAND: hdfs dfs -cat /user/<Name>/wordcount/output/*


SENGODAN R | Micros
YouTube
Gladiator Full Movies
Subscription Details
kct5thsemcdhid031@ip
New tab
Nuvepro Web FTP
https://npbdh.cloudloka.com:4200
Total committed heap usage (bytes)=2592604160
Peak Map Physical memory (bytes)=457826304
Peak Map Virtual memory (bytes)=2578415616
Peak Reduce Physical memory (bytes)=278921216
Peak Reduce Virtual memory (bytes)=2600792064
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=7262
File Output Format Counters
Bytes Written=0
[kct5thsemcdhid031@ip-10-1-1-204 weather]$ hdfs dfs -cat /user/kct5thsemcdhid031/weather/output/*
-bash: Name: No such file or directory
[kct5thsemcdhid031@ip-10-1-1-204 weather]$ hdfs dfs -cat /user/kct5thsemcdhid031/weather/output/*
25-Jan-2014 Time: 12:34:542 MinTemp: -22.3 Time: 05:12:345 MaxTemp: 35.7
26-Jan-2014 Time: 22:00:093 MinTemp: -27.0 Time: 05:12:345 MaxTemp: 55.7
27-Jan-2014 Time: 02:34:542 MinTemp: -22.3 Time: 05:12:345 MaxTemp: 55.7
29-Jan-2014 Time: 14:00:093 MinTemp: -17.0 Time: 02:34:542 MaxTemp: 62.9
30-Jan-2014 Time: 22:00:093 MinTemp: -27.0 Time: 05:12:345 MaxTemp: 49.2
31-Jan-2014 Time: 14:00:093 MinTemp: -17.0 Time: 03:12:187 MaxTemp: 56.0
29-Jan-2014 Time: 14:00:093 MinTemp: -27.0 Time: 05:12:345 MaxTemp: 49.2
30-Jan-2014 Time: 14:00:093 MinTemp: -17.0 Time: 02:34:542 MaxTemp: 52.9
31-Jan-2014 Time: 15:12:345 MinTemp: -15.7 Time: 03:12:187 MaxTemp: 56.0
28-Jan-2014 Time: 11:19:345 MinTemp: -23.3 Time: 05:12:345 MaxTemp: 35.7
29-Jan-2014 Time: 14:00:093 MinTemp: -17.0 Time: 02:34:542 MaxTemp: 52.9
30-Jan-2014 Time: 15:12:345 MinTemp: -15.7 Time: 03:12:187 MaxTemp: 56.0
25-Jan-2014 Time: 12:34:542 MinTemp: -22.3 Time: 05:12:345 MaxTemp: 35.7
26-Jan-2014 Time: 04:00:093 MinTemp: -14.0 Time: 05:12:345 MaxTemp: 55.7
27-Jan-2014 Time: 02:34:542 MinTemp: -22.3 Time: 00:14:045 MaxTemp: 35.7
28-Jan-2014 Time: 11:19:345 MinTemp: -23.3 Time: 05:12:345 MaxTemp: 35.7
29-Jan-2014 Time: 14:00:093 MinTemp: -17.0 Time: 02:34:542 MaxTemp: 52.9
30-Jan-2014 Time: 15:12:345 MinTemp: -15.7 Time: 03:12:187 MaxTemp: 56.0
31-Jan-2014 Time: 22:00:093 MinTemp: -27.0 Time: 05:12:345 MaxTemp: 49.2
29-Jan-2014 Time: 14:00:093 MinTemp: -17.0 Time: 02:34:542 MaxTemp: 52.9
30-Jan-2014 Time: 15:12:345 MinTemp: -15.7 Time: 03:12:187 MaxTemp: 56.0
29-Jan-2014 Time: 14:00:093 MinTemp: -17.0 Time: 02:34:542 MaxTemp: 52.9
30-Jan-2014 Time: 15:12:345 MinTemp: -15.7 Time: 03:12:187 MaxTemp: 56.0
31-Jan-2014 Time: 22:00:093 MinTemp: -27.0 Time: 05:12:345 MaxTemp: 49.2
[kct5thsemcdhid031@ip-10-1-1-204 weather]$

LAB EXERCISE-8

SPARK

AIM:

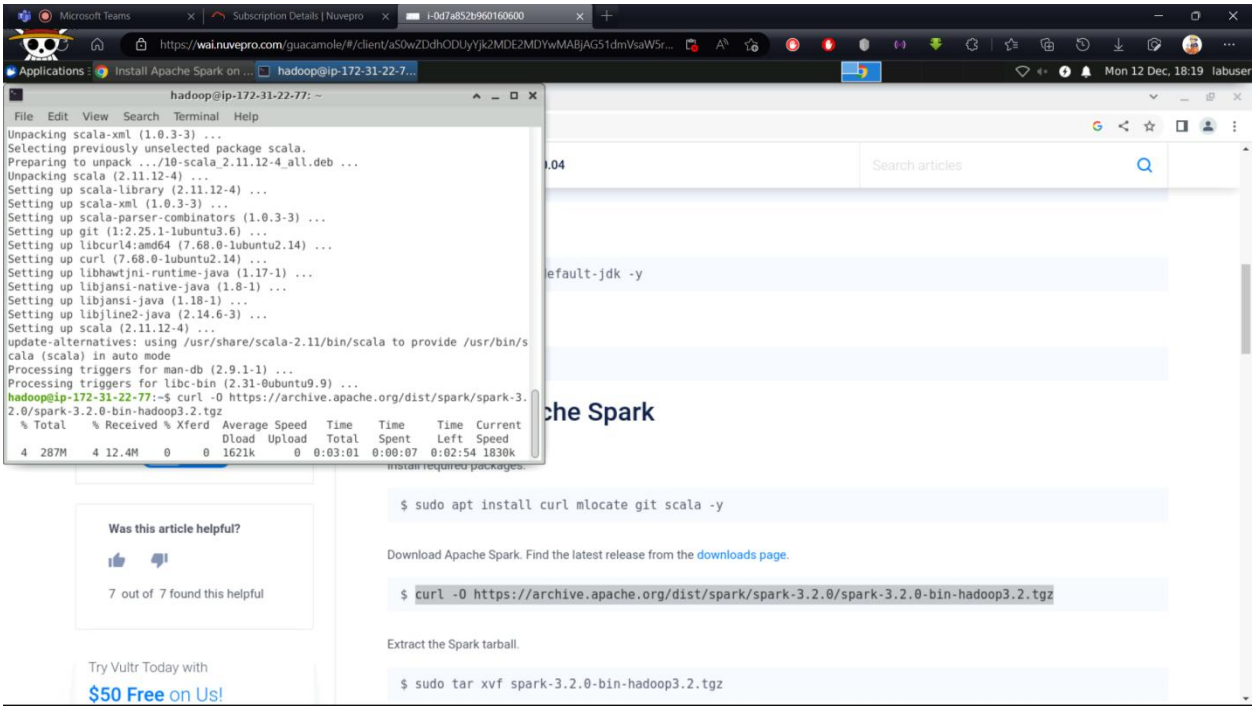
To Perform simple join using Mapper in Spark.

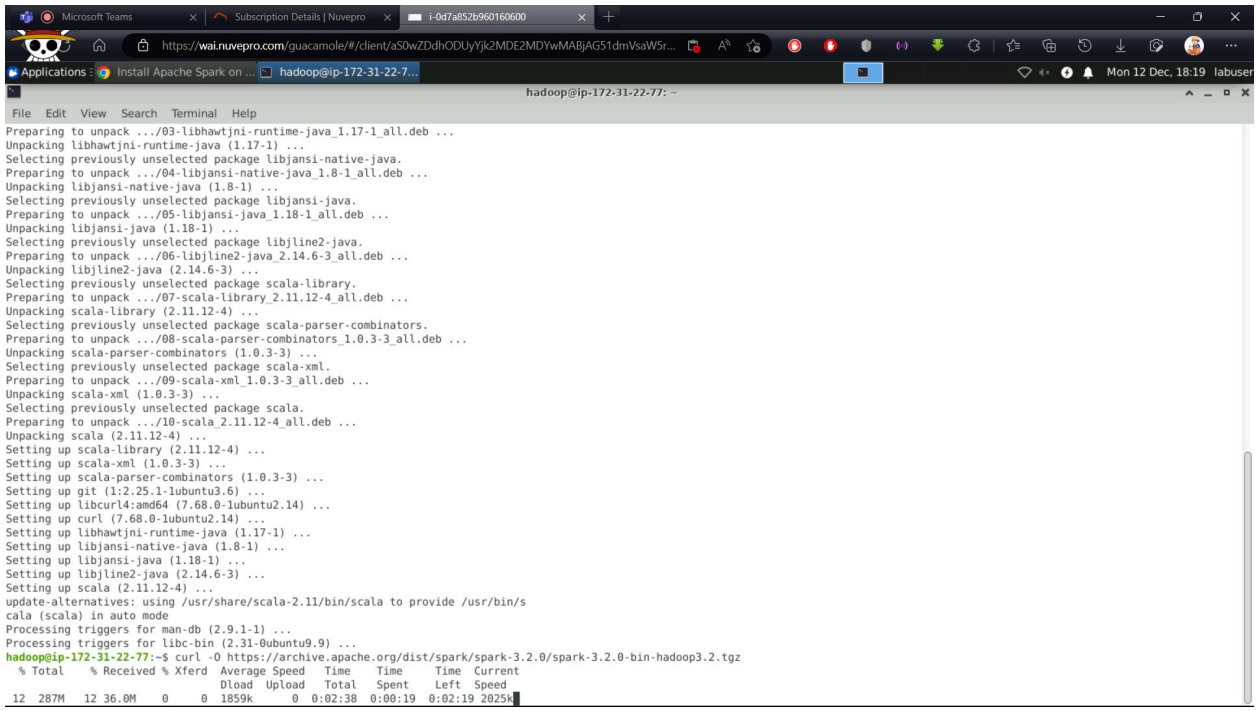
THEORY:

Spark DataFrame supports all basic SQL Join Types like INNER, LEFT OUTER, RIGHT OUTER, LEFT ANTI, LEFT SEMI, CROSS,

SELF JOIN. Spark SQL Joins are wider transformations that result in data shuffling over the network hence they have huge performance issues when not designed with care.

INSTALLATION OF SPARK




Microsoft Teams
Subscription Details | Nuvepro
i-0d7a852b960160600
https://wai.nuvepro.com/guacamole/#/client/aS0wZDdhODUyYjk2MDE2MDYwMABjAG51dmVsaW5r...
Applications
Install Apache Spark on ...
hadoop@ip-172-31-22-7...
Mon 12 Dec, 18:19 labuser
hadoop@ip-172-31-22-77: ~
File Edit View Search Terminal Help
Preparing to unpack .../03-libhawtjni-runtime-java_1.17-1_all.deb ...
Unpacking libhawtjni-runtime-java (1.17-1) ...
Selecting previously unselected package libjansi-native-java.
Preparing to unpack .../04-libjansi-native-java_1.8-1_all.deb ...
Unpacking libjansi-native-java (1.8-1) ...
Selecting previously unselected package libjansi-java.
Preparing to unpack .../05-libjansi-java_1.18-1_all.deb ...
Unpacking libjansi-java (1.18-1) ...
Selecting previously unselected package libjline2-java.
Preparing to unpack .../06-libjline2-java_2.14.6-3_all.deb ...
Unpacking libjline2-java (2.14.6-3) ...
Selecting previously unselected package scala-library.
Preparing to unpack .../07-scala-library_2.11.12-4_all.deb ...
Unpacking scala-library (2.11.12-4) ...
Selecting previously unselected package scala-parser-combinators.
Preparing to unpack .../08-scala-parser-combinators_1.0.3-3_all.deb ...
Unpacking scala-parser-combinators (1.0.3-3) ...
Selecting previously unselected package scala-xml.
Preparing to unpack .../09-scala-xml_1.0.3-3_all.deb ...
Unpacking scala-xml (1.0.3-3) ...
Selecting previously unselected package scala.
Preparing to unpack .../10-scala_2.11.12-4_all.deb ...
Unpacking scala (2.11.12-4) ...
Setting up scala-library (2.11.12-4) ...
Setting up scala-xml (1.0.3-3) ...
Setting up scala-parser-combinators (1.0.3-3) ...
Setting up git (1:2.25.1-1ubuntu3.6) ...
Setting up libcurl4:amd64 (7.68.0-1ubuntu2.14) ...
Setting up curl (7.68.0-1ubuntu2.14) ...
Setting up libhawtjni-runtime-java (1.17-1) ...
Setting up libjansi-native-java (1.8-1) ...
Setting up libjansi-java (1.18-1) ...
Setting up libjline2-java (2.14.6-3) ...
Setting up scala (2.11.12-4) ...
update-alternatives: using /usr/share/scala-2.11/bin/scala to provide /usr/bin/s
cala (scala) in auto mode
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.9) ...
hadoop@ip-172-31-22-77:~$ curl -O https://archive.apache.org/dist/spark/spark-3.2.0/spark-3.2.0-bin-hadoop3.2.tgz
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
12 287M 12 36.0M 0 0 1859k 0 0:02:38 0:00:19 0:02:19 2025k


Microsoft Teams
Subscription Details | Nuvepro
i-0d7a852b960160600
Apache Spark Installation - YouT
Page not found – (BigTechTalk)
https://wai.nuvepro.com/guacamole/#/client/aS0wZDdhODUyYjk2MDE2MDYwMABjAG51dmVsaW5r...
Applications
Install Apache Spark on ...
hadoop@ip-172-31-22-7...
hadoop - File Manager
Mon 12 Dec, 18:33 labuser
hadoop@ip-172-31-22-77: ~/spark-3.2.0-bin-hadoop3.2/bin
File Edit View Search Terminal Help
hadoop@ip-172-31-22-77:~$ ls
hadoop-3.3.1.tar.gz hdfs spark-3.2.0-bin-hadoop3.2.tgz
hadoop-3.3.1.tar.gz.1 spark-3.2.0-bin-hadoop3.2
hadoop@ip-172-31-22-77:~$ cd spark-3.2.0-bin-hadoop3.2
hadoop@ip-172-31-22-77:~/spark-3.2.0-bin-hadoop3.2$ ls
LICENSE R RELEASE conf examples kubernetes python yarn
NOTICE README.md bin data jars licenses sbin
hadoop@ip-172-31-22-77:~/spark-3.2.0-bin-hadoop3.2$ cd bin
hadoop@ip-172-31-22-77:~/spark-3.2.0-bin-hadoop3.2/bin$ ls
beeline pyspark spark-class2.cmd spark-submit
beeline.cmd pyspark.cmd spark-shell spark-submit.cmd
docker-image-tool.sh pyspark2.cmd spark-shell.cmd spark-submit2.cmd
find-spark-home run-example spark-shell2.cmd sparkR
find-spark-home.cmd run-example.cmd spark-sql sparkR.cmd
load-spark-env.cmd spark-class spark-sql.cmd sparkR2.cmd
load-spark-env.sh spark-class.cmd spark-sql2.cmd
hadoop@ip-172-31-22-77:~/spark-3.2.0-bin-hadoop3.2/bin$ ^C
hadoop@ip-172-31-22-77:~/spark-3.2.0-bin-hadoop3.2/bin$ spark-shell
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.spark.unsafe.Platform (file:/opt/spark/jars/spark-unsafe_2.12-3.2.0.jar) to constructor java.nio.DirectByteBuffer(long,int)
WARNING: Please consider reporting this to the maintainers of org.apache.spark.unsafe.Platform
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
22/12/12 18:33:08 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Spark context Web UI available at http://ip-172-31-22-77.ap-south-1.compute.internal:4040
Spark context available as 'sc' (master = local[*], app id = local-1670869990240).
Spark session available as 'spark'.
Welcome to
version 3.2.0
Using Scala version 2.12.15 (OpenJDK 64-Bit Server VM, Java 11.0.17)
Type in expressions to have them evaluated.
Type :help for more information.
scala>

JOIN QUERY:

Queries can access multiple tables at once, or access the same table in such a way that multiple rows of the table are being processed at the same time. A query that accesses multiple rows of the same or different tables at one time is called a join query.

Parameters

Relation:Specifies the relation to be joined. join_type:Specifies the join type.

Syntax:

[INNER] | CROSS | LEFT [OUTER] | [LEFT] SEMI | RIGHT [OUTER] | FULL [OUTER] | [LEFT] ANTI

join_criteria :Specifies how the rows from one relation will be combined with the rows of another relation.

Syntax:

ON boolean_expression | USING (column_name [, ...])

boolean_expression: Specifies an expression with a return type of boolean.

join Operators

join(right: Dataset[_]): DataFrame (1)

join(right: Dataset[_], usingColumn: String): DataFrame (2)

join(right: Dataset[_], usingColumns: Seq[String]): DataFrame (3)

join(right: Dataset[_], usingColumns: Seq[String], joinType: String): DataFrame (4)

join(right: Dataset[_], joinExprs: Column): DataFrame (5)

join(right: Dataset[_], joinExprs: Column, joinType: String): DataFrame (6) Condition-less inner join

Inner join with a single column that exists on both sides Inner join with columns that exist on both sides

Equi-join with explicit join type Inner join

Join with explicit join type. Self-joins are acceptable.

1.INNER JOIN:

The inner join is the default join in Spark SQL. It selects rows that have matching values in both relations.

Syntax:

relation [INNER] JOIN relation [join_criteria]

Commands

1.val left = Seq((0, "zero"), (1, "one")).toDF("id", "left")

val right = Seq((0, "zero"), (2, "two"), (3, "three")).toDF("id", "right") 3.left.join(right, "id").show

left.join(right, "id").explain

2.FULL OUTER JOIN:

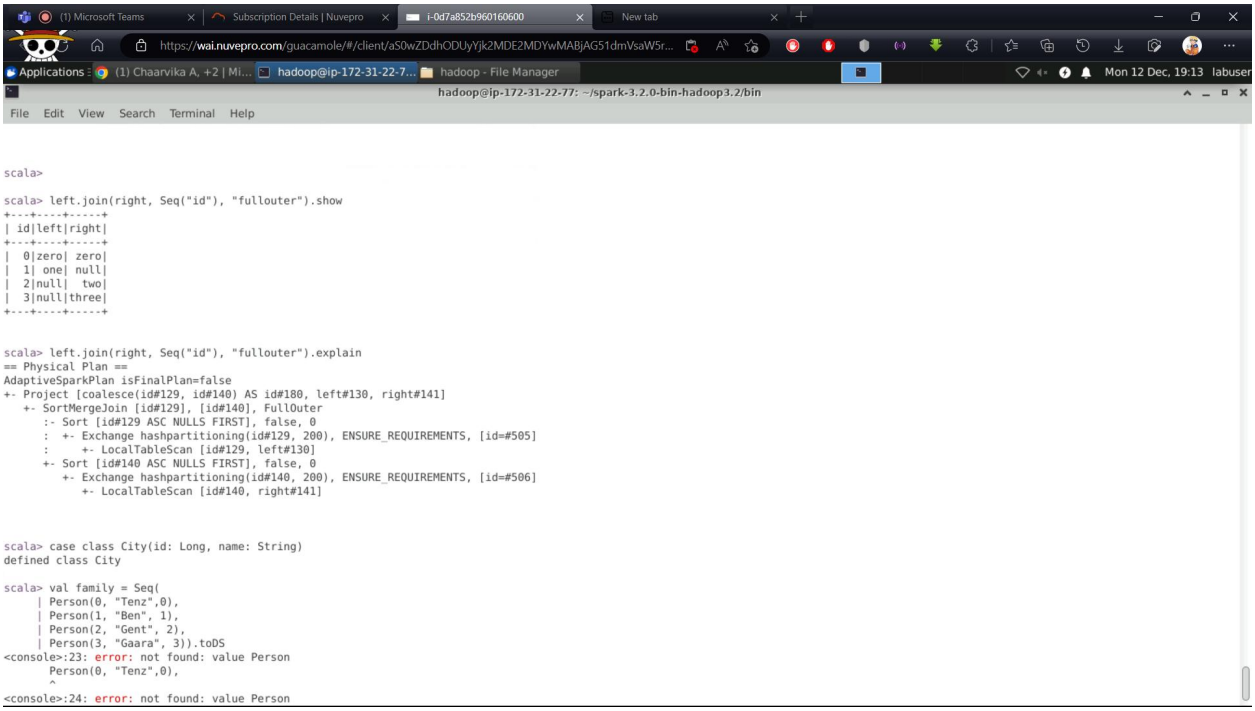
In SQL the FULL OUTER JOIN combines the results of both left and right outer joins and returns all (matched or unmatched) rows from the tables on both sides of the join clause.

The FULL OUTER JOIN keyword returns all records when there is a match in left (table1) or right (table2) table records.

Commands

left.join(right, Seq("id"), "fullouter").show

left.join(right, Seq("id"), "fullouter").explain



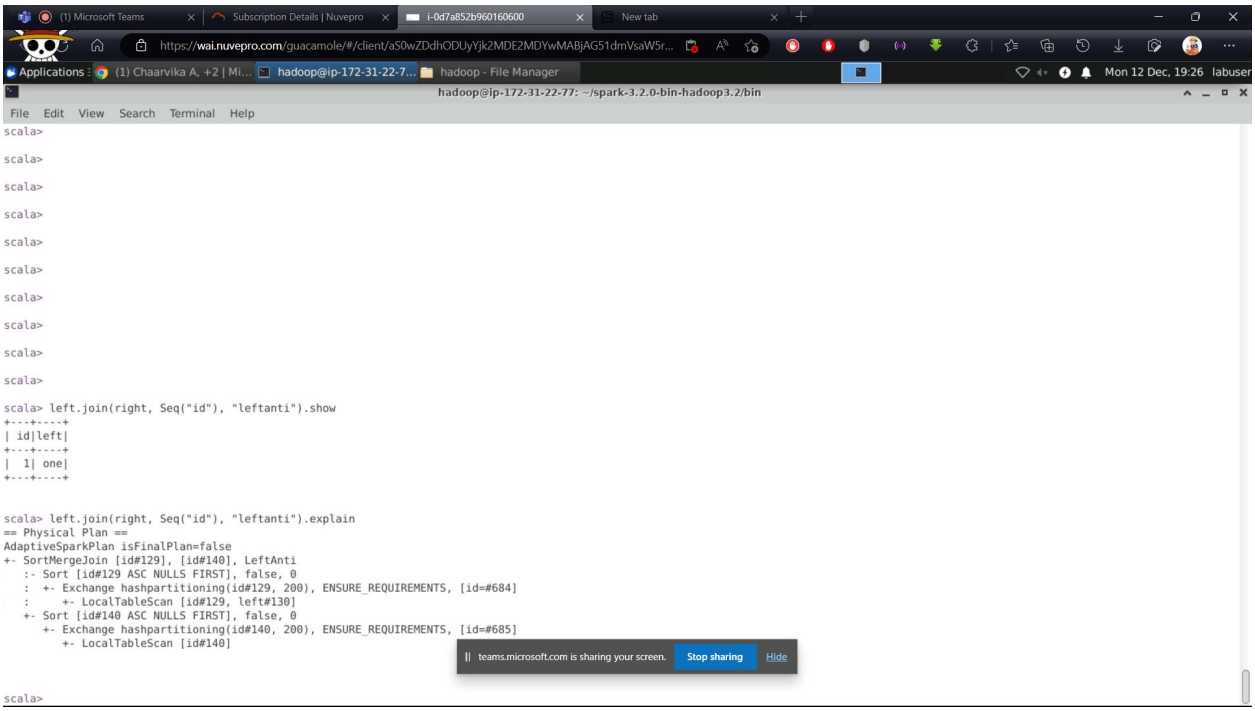
3.LEFT ANTI-JOIN

A left anti join returns that all rows from the first table which do not have a match in the second table.

Commands

left.join(right, Seq("id"), "leftanti").show

left.join(right, Seq("id"), "leftanti").explain



4.BROADCAST JOIN

Spark SQL uses broadcast join (aka broadcast hash join) instead of hash join to optimize join queries when the size of one side data is below spark.sql.autoBroadcastJoinThreshold.

Broadcast join can be very efficient for joins between a large table (fact) with relatively small tables (dimensions) that could then be used to perform a star- schema join. It can avoid sending all data of the large table over the network.

You can use broadcast function or SQL's broadcast hints to mark a dataset to be broadcast when used in a join query.

broadcast join is also called a replicated join (in the distributed system community) or a map-side join (in the Hadoop community).

JoinSelection execution planning strategy uses spark.sql.autoBroadcastJoinThreshold property (default: 10M) to control the size of a dataset before broadcasting it to all worker nodes when performing a join.

Commands

1. val threshold = spark.conf.get("spark.sql.autoBroadcastJoinThreshold").toInt 2.threshold / 1024 / 1024

3.val q = spark.range(100).as("a").join(spark.range(100).as("b")).where($"a.id" ===

$"b.id")

4.println(q.queryExecution.logical.numberedTreeString)

5.q.explain

6.spark.conf.set("spark.sql.autoBroadcastJoinThreshold", -1)
7.spark.conf.get("spark.sql.autoBroadcastJoinThreshold")

1.q.explain

2.val qBroadcast =
spark.range(100).as("a").join(broadcast(spark.range(100)).as("b")).where($"a.id"

=== $"b.id") 3.qBroadcast.explain

4.val qBroadcastLeft = """

SELECT /*+ BROADCAST (lf) */ *

FROM range(100) lf, range(1000) rt WHERE lf.id = rt.id

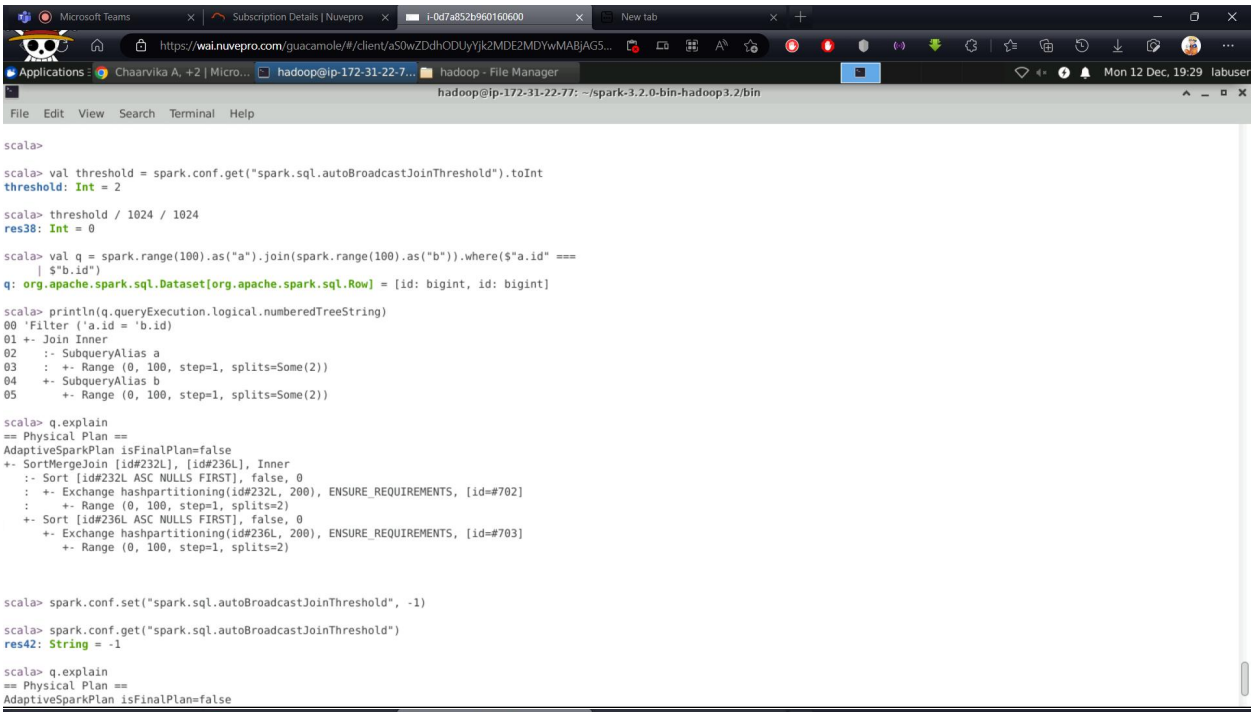
"""

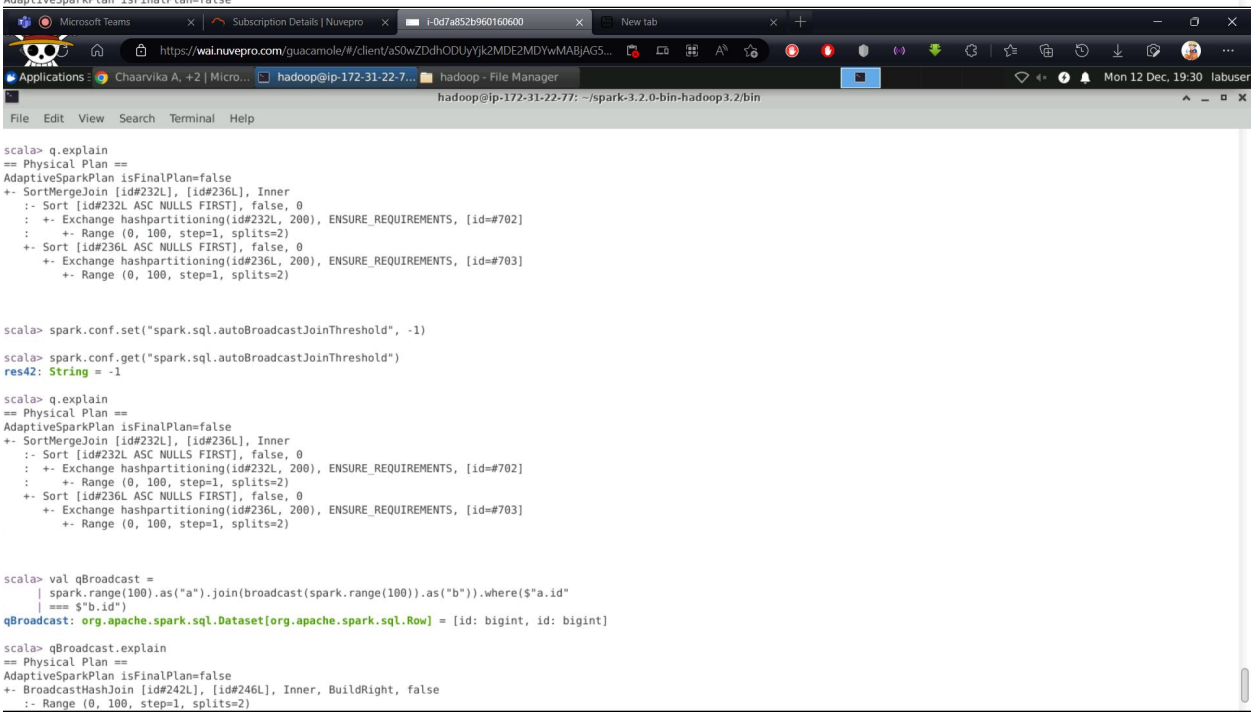
scala> sql(qBroadcastLeft).explain

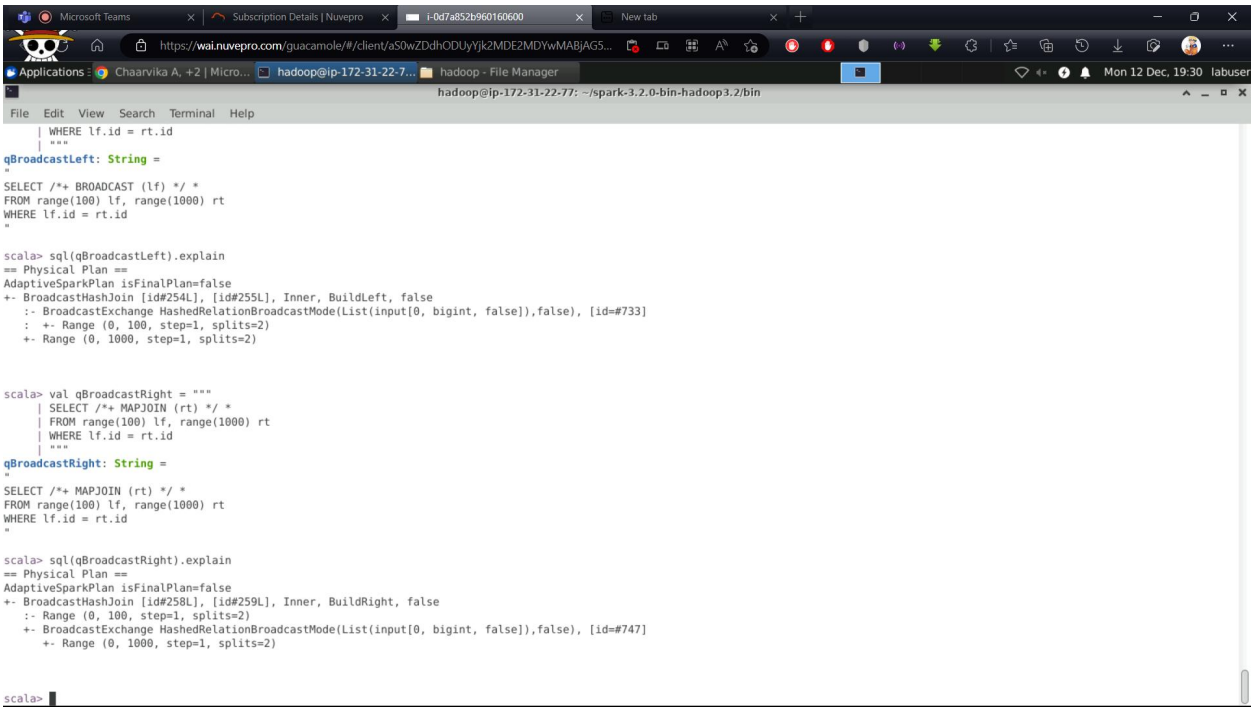
1. val qBroadcastRight = """ SELECT /*+ MAPJOIN (rt) */ * FROM range(100) lf, range(1000) rt
WHERE lf.id = rt.id

"""

scala> sql(qBroadcastRight).explain


Microsoft Teams
Subscription Details | Nuvepro
i-0d7a852b960160600
New tab
https://wai.nuvepro.com/guacamole/#/client/aS0wZDdhODUyYjk2MDE2MDYwMABjAG5...
Applications
Chaarvika A, +2 | Micro...
hadoop@ip-172-31-22-7...
hadoop - File Manager
Mon 12 Dec, 19:29
labuser
hadoop@ip-172-31-22-77: ~/spark-3.2.0-bin-hadoop3.2/bin
File Edit View Search Terminal Help
scala>
scala> val threshold = spark.conf.get("spark.sql.autoBroadcastJoinThreshold").toInt
threshold: Int = 2
scala> threshold / 1024 / 1024
res38: Int = 0
scala> val q = spark.range(100).as("a").join(spark.range(100).as("b")).where($"a.id" ===
| $"b.id")
q: org.apache.spark.sql.Dataset[org.apache.spark.sql.Row] = [id: bigint, id: bigint]
scala> println(q.queryExecution.logical.numberedTreeString)
00 'Filter ('a.id = 'b.id)
01 +- Join Inner
02 :- SubqueryAlias a
03 : +- Range (0, 100, step=1, splits=Some(2))
04 +- SubqueryAlias b
05 +- Range (0, 100, step=1, splits=Some(2))
scala> q.explain
== Physical Plan ==
AdaptiveSparkPlan isFinalPlan=false
+- SortMergeJoin [id#232L], [id#236L], Inner
:- Sort [id#232L ASC NULLS FIRST], false, 0
: +- Exchange hashpartitioning(id#232L, 200), ENSURE_REQUIREMENTS, [id=#702]
: +- Range (0, 100, step=1, splits=2)
+- Sort [id#236L ASC NULLS FIRST], false, 0
+- Exchange hashpartitioning(id#236L, 200), ENSURE_REQUIREMENTS, [id=#703]
+- Range (0, 100, step=1, splits=2)
scala> spark.conf.set("spark.sql.autoBroadcastJoinThreshold", -1)
scala> spark.conf.get("spark.sql.autoBroadcastJoinThreshold")
res42: String = -1
scala> q.explain
== Physical Plan ==
AdaptiveSparkPlan isFinalPlan=false


Microsoft Teams
Subscription Details | Nuvepro
i-0d7a852b960160600
New tab
https://wai.nuvepro.com/guacamole/#/client/aS0wZDdhODUyYjk2MDE2MDYwMABjAG5...
Applications
Chaarvika A, +2 | Micro...
hadoop@ip-172-31-22-7...
hadoop - File Manager
Mon 12 Dec, 19:30
labuser
hadoop@ip-172-31-22-77: ~/spark-3.2.0-bin-hadoop3.2/bin
File Edit View Search Terminal Help
scala> q.explain
== Physical Plan ==
AdaptiveSparkPlan isFinalPlan=false
+- SortMergeJoin [id#232L], [id#236L], Inner
:- Sort [id#232L ASC NULLS FIRST], false, 0
: +- Exchange hashpartitioning(id#232L, 200), ENSURE_REQUIREMENTS, [id=#702]
: +- Range (0, 100, step=1, splits=2)
+- Sort [id#236L ASC NULLS FIRST], false, 0
+- Exchange hashpartitioning(id#236L, 200), ENSURE_REQUIREMENTS, [id=#703]
+- Range (0, 100, step=1, splits=2)
scala> spark.conf.set("spark.sql.autoBroadcastJoinThreshold", -1)
scala> spark.conf.get("spark.sql.autoBroadcastJoinThreshold")
res42: String = -1
scala> q.explain
== Physical Plan ==
AdaptiveSparkPlan isFinalPlan=false
+- SortMergeJoin [id#232L], [id#236L], Inner
:- Sort [id#232L ASC NULLS FIRST], false, 0
: +- Exchange hashpartitioning(id#232L, 200), ENSURE_REQUIREMENTS, [id=#702]
: +- Range (0, 100, step=1, splits=2)
+- Sort [id#236L ASC NULLS FIRST], false, 0
+- Exchange hashpartitioning(id#236L, 200), ENSURE_REQUIREMENTS, [id=#703]
+- Range (0, 100, step=1, splits=2)
scala> val qBroadcast =
| spark.range(100).as("a").join(broadcast(spark.range(100)).as("b")).where($"a.id"
| === $"b.id")
qBroadcast: org.apache.spark.sql.Dataset[org.apache.spark.sql.Row] = [id: bigint, id: bigint]
scala> qBroadcast.explain
== Physical Plan ==
AdaptiveSparkPlan isFinalPlan=false
+- BroadcastHashJoin [id#242L], [id#246L], Inner, BuildRight, false
:- Range (0, 100, step=1, splits=2)


Microsoft Teams
Subscription Details | Nuvepro
i-0d7a852b960160600
New tab
https://wai.nuvepro.com/guacamole/#/client/aS0wZDdhODUyYjk2MDE2MDYwMABjAG5...
Applications
Chaarvika A, +2 | Micro...
hadoop@ip-172-31-22-7...
hadoop - File Manager
Mon 12 Dec, 19:30 labuser
hadoop@ip-172-31-22-77: ~/spark-3.2.0-bin-hadoop3.2/bin
File Edit View Search Terminal Help
| WHERE lf.id = rt.id
| """
qBroadcastLeft: String =
"
SELECT /*+ BROADCAST (lf) */ *
FROM range(100) lf, range(1000) rt
WHERE lf.id = rt.id
"
scala> sql(qBroadcastLeft).explain
== Physical Plan ==
AdaptiveSparkPlan isFinalPlan=false
+- BroadcastHashJoin [id#254L], [id#255L], Inner, BuildLeft, false
:- BroadcastExchange HashedRelationBroadcastMode(List(input[0, bigint, false]),false), [id=#733]
: +- Range (0, 100, step=1, splits=2)
+- Range (0, 1000, step=1, splits=2)
scala> val qBroadcastRight = """
| SELECT /*+ MAPJOIN (rt) */ *
| FROM range(100) lf, range(1000) rt
| WHERE lf.id = rt.id
| """
qBroadcastRight: String =
"
SELECT /*+ MAPJOIN (rt) */ *
FROM range(100) lf, range(1000) rt
WHERE lf.id = rt.id
"
scala> sql(qBroadcastRight).explain
== Physical Plan ==
AdaptiveSparkPlan isFinalPlan=false
+- BroadcastHashJoin [id#258L], [id#259L], Inner, BuildRight, false
:- Range (0, 100, step=1, splits=2)
+- BroadcastExchange HashedRelationBroadcastMode(List(input[0, bigint, false]),false), [id=#747]
+- Range (0, 1000, step=1, splits=2)
scala>

LAB EXCERSICE: 9

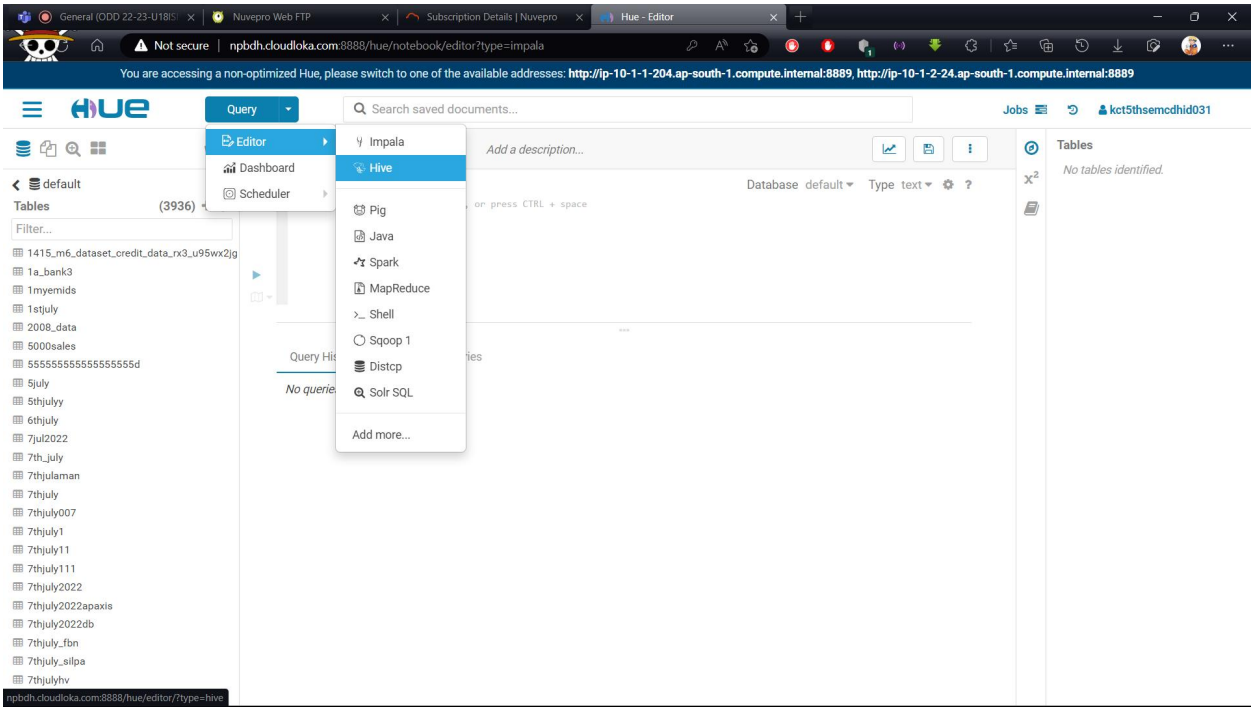
HIVE

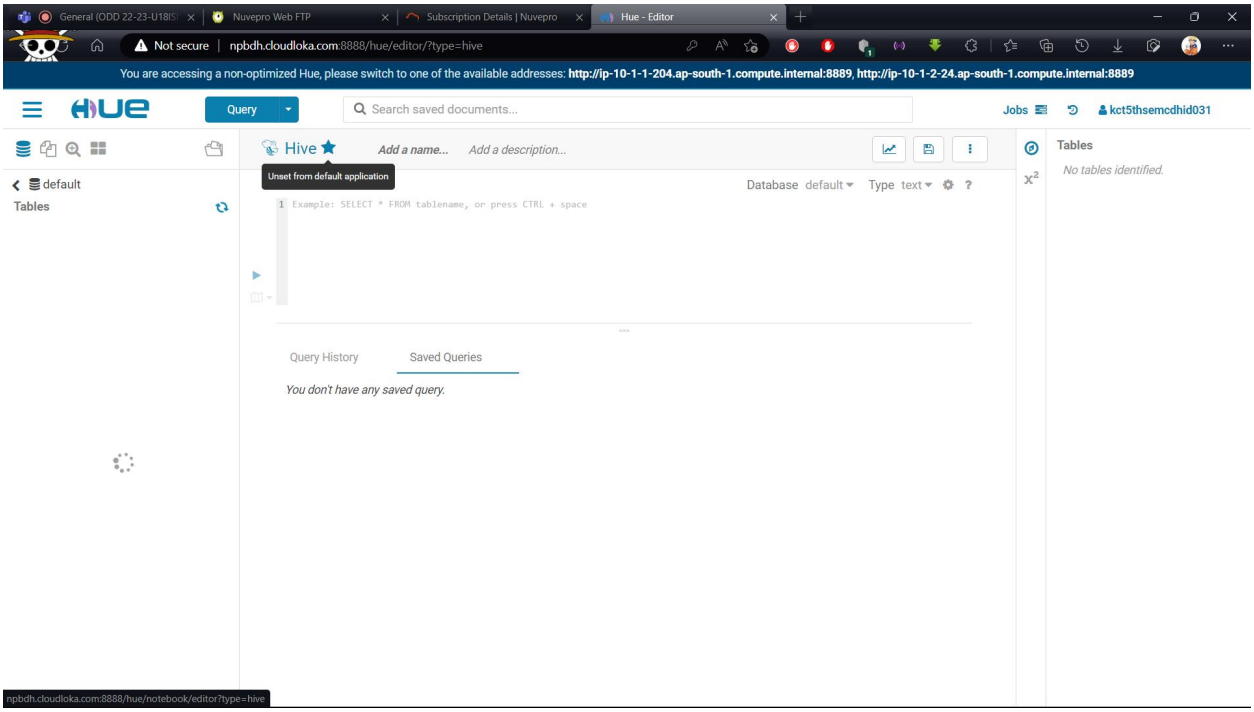
AIM:

Install and Run Hive then use Hive to create, alter, and drop databases, tables, views,functions, and indexes

PROCEDURE:

STEP1:Opening HIVE from HUE portal cloudera

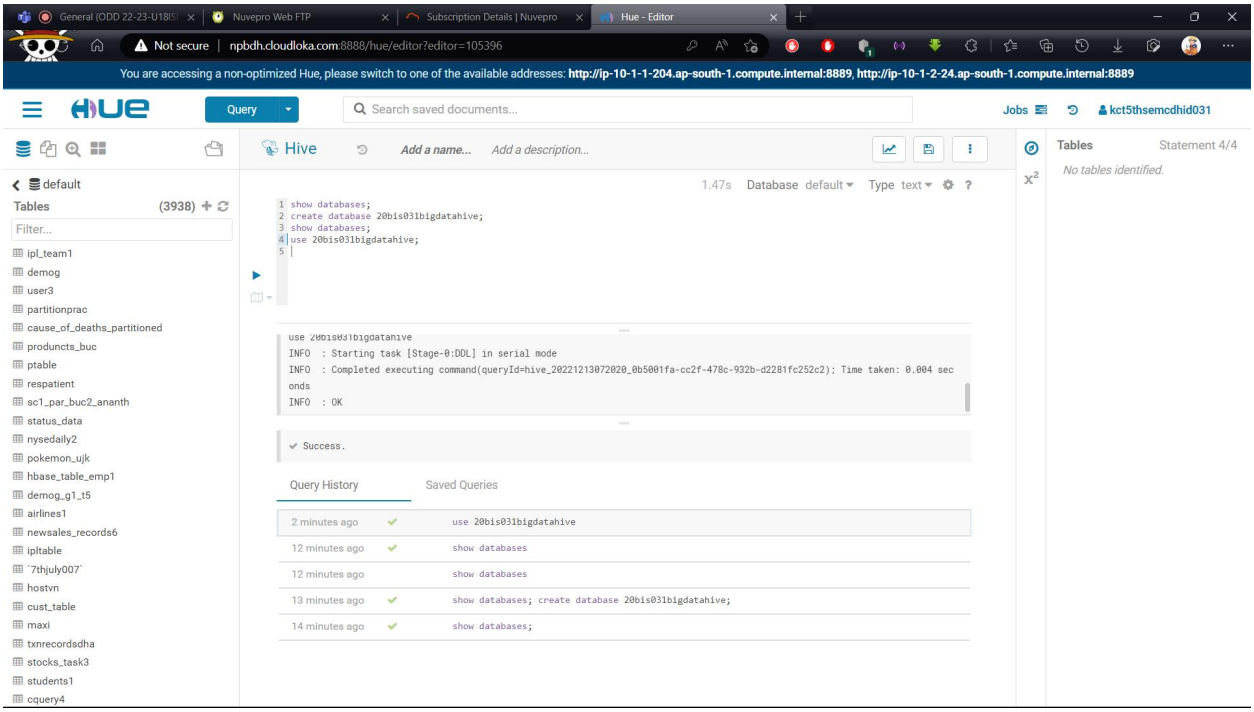


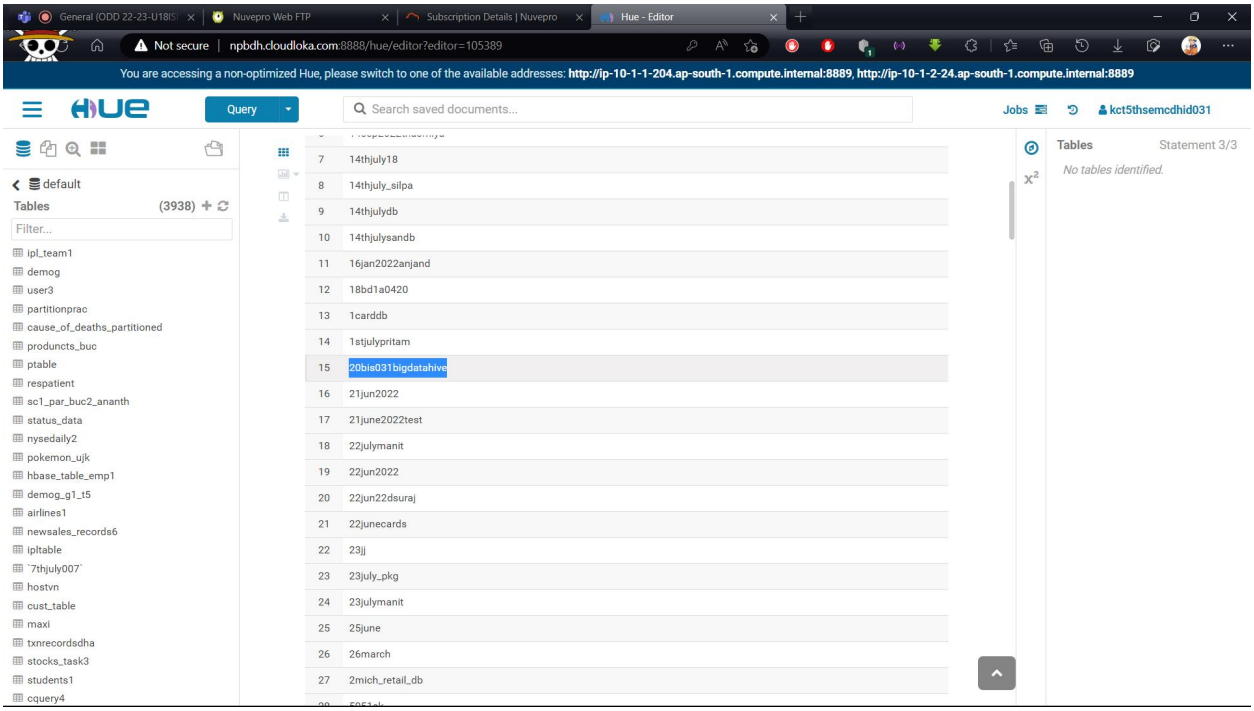


STEP2: Create database

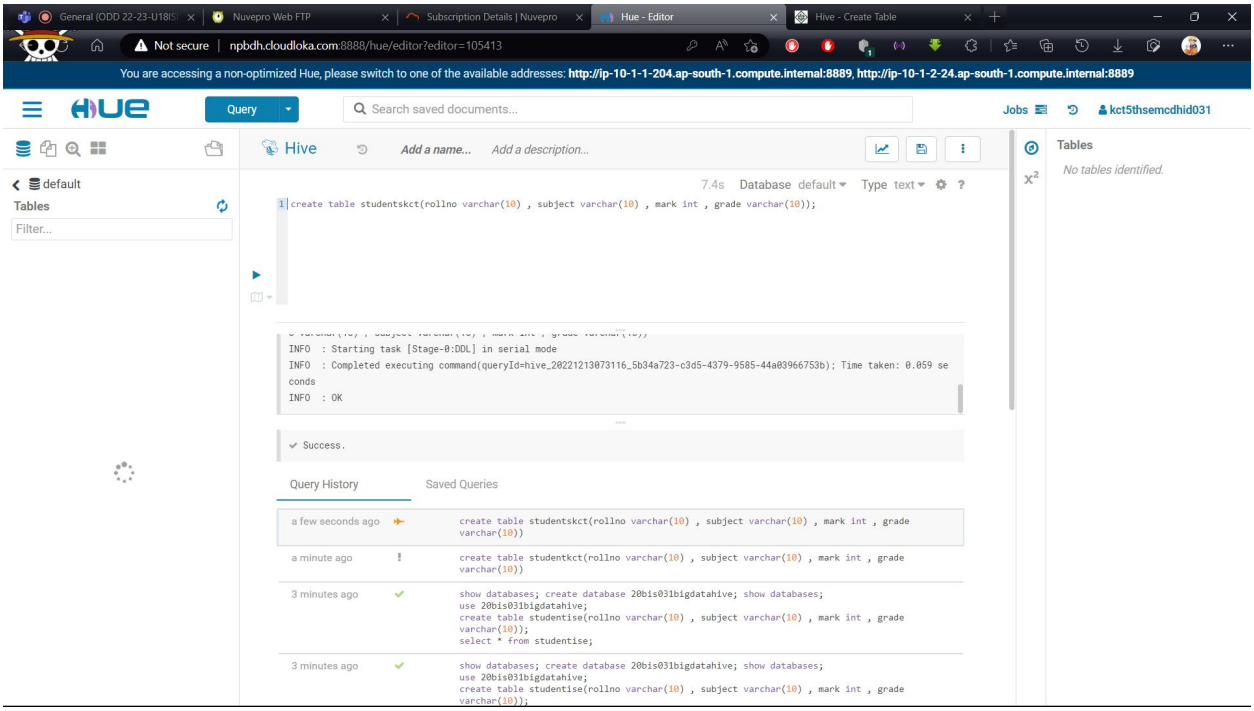
create database 20bis031bigdatahive; show databases;

use 20bis031bigdatahive;



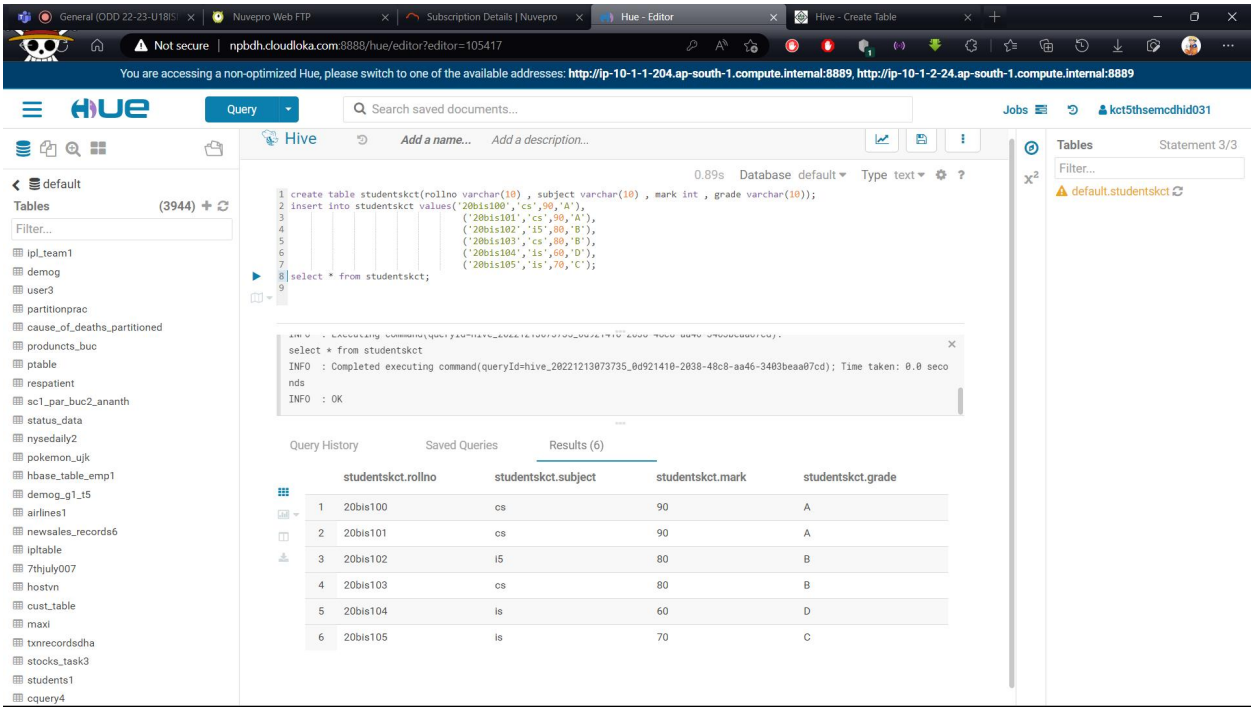


STEP3:Create table



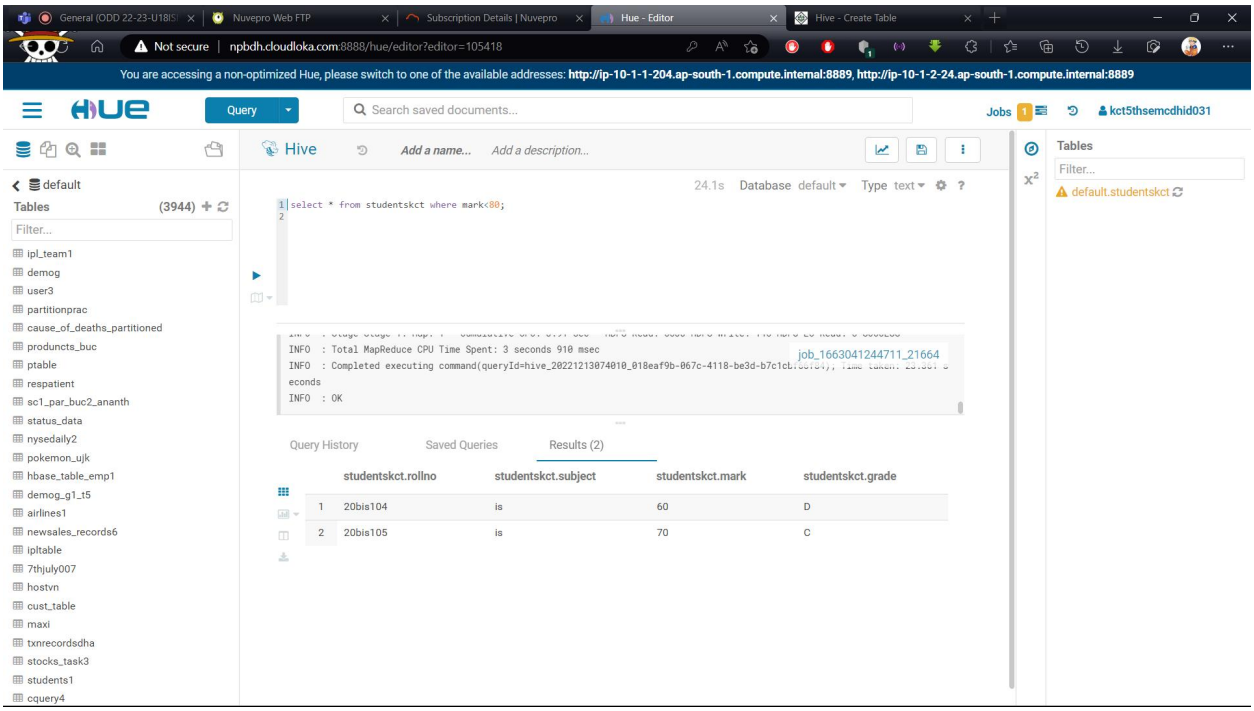
STEP4:Insert values in the table

```
insert into studentskct values('20bis100','cs',90,'A'),
                               ('20bis101','cs',90,'A'),
                               ('20bis102','i5',80,'B'),
                               ('20bis103','cs',80,'B'),
                               ('20bis104','is',60,'D'),
                               ('20bis105','is',70,'C');
select * from studentskct;
```



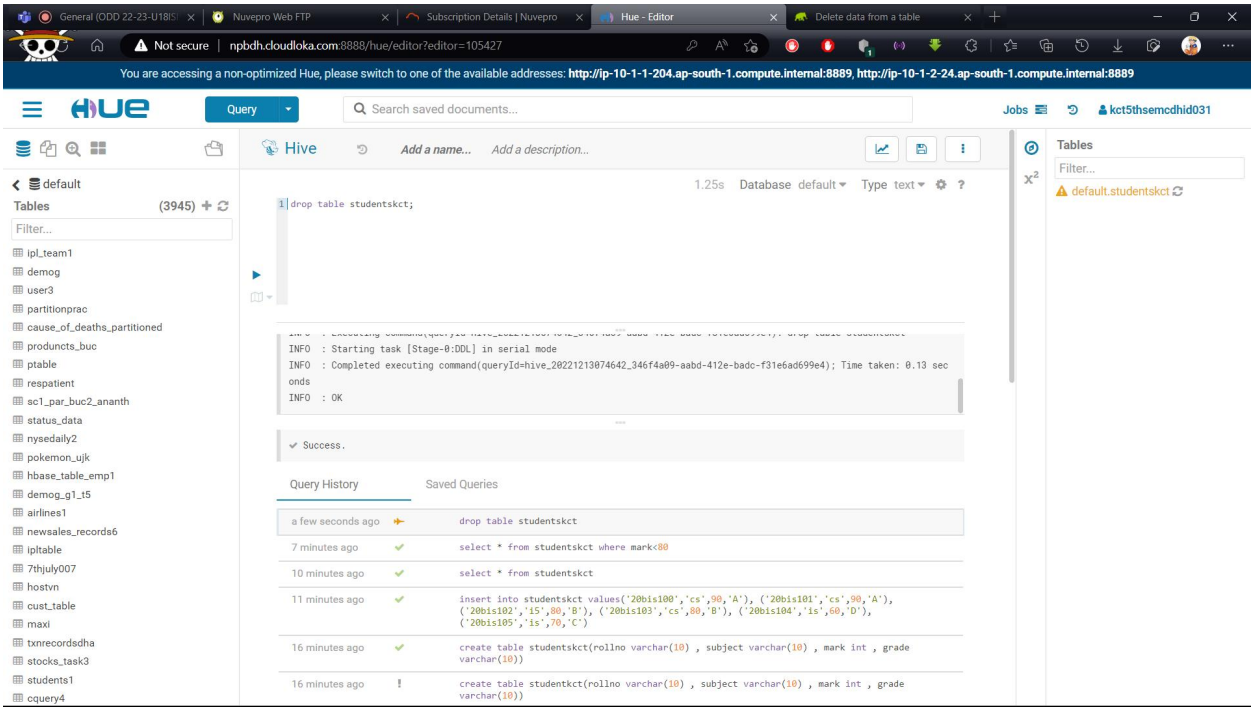
STEP5:Filtering records with condition

```
select * from studentskct where mark<80;
```



STEP6:Drop table

drop table studentskct;



LAB EXERCISE-10

SPARK

AIM:

To Verify, Sparse and perform advance join of data using spark

THEORY:

Join operations in Apache Spark is often a biggest source of performance problems and even full-blown exceptions in Spark. After this talk, you will understand the two most basic methods Spark employs for joining dataframes – to the level of detail of how Spark distributes the data within the cluster. You'll also find out how to work out common errors and even handle the trickiest corner cases we've encountered! After this talk, you should be able to write performance joins in Spark SQL that scale and are zippy fast!

This session will cover different ways of joining tables in Apache Spark.

ShuffleHashJoin

–A ShuffleHashJoin is the most basic way to join tables in Spark – we'll diagram how Spark shuffles the dataset to make this happen.

BroadcastHashJoin

–A BroadcastHashJoin is also a very common way for Spark to join two tables under the special condition that one of your tables is small.

Dealing with Key Skew in a ShuffleHashJoin

–Key Skew is a common source of slowness for a Shuffle Hash Join – we'll describe what this is and how you might work around this.

CartesianJoin

–Cartesian Joins is a hard problem – we'll describe why it's difficult as well as what you need to do to make that work and what to look out for.

One to Many Joins

–When a single row in one table can match to many rows in your other table, the total number of output rows in your joined table can be really high. We'll let you know how to deal with this.

Theta Joins

–If you aren't joining two tables strictly by key, but instead checking on a condition for your tables, you may need to provide some hints to Spark SQL to get this to run well.

1. Broadcast Hash Join

COMMANDS

scala> spark.conf.get("spark.sql.autoBroadcastJoinThreshold") res1: String = 10485760

scala> val data1 = Seq(10, 20, 20, 30, 40, 10, 40, 20, 20, 20, 20, 50)

data1: Seq[Int] = List(10, 20, 20, 30, 40, 10, 40, 20, 20, 20, 20, 50)

scala> val df1 = data1.toDF("id1")

df1: org.apache.spark.sql.DataFrame = [id1: int] scala> val data2 = Seq(30, 20, 40, 50)

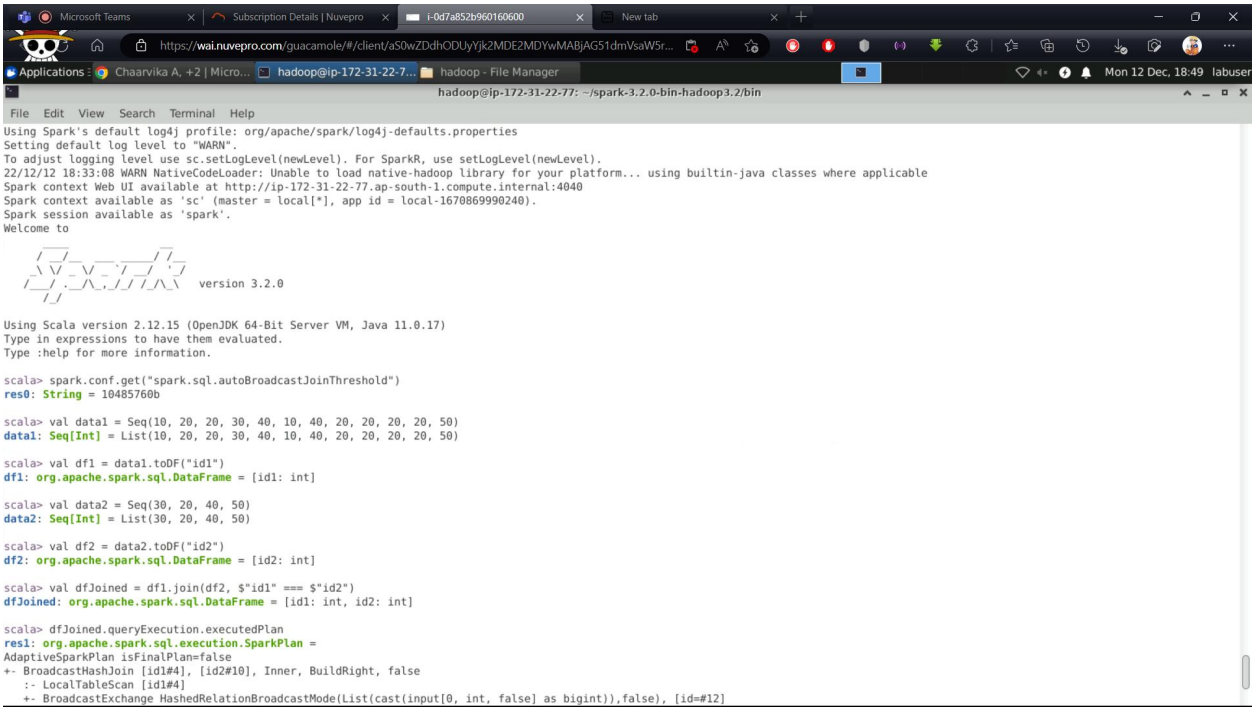
data2: Seq[Int] = List(30, 20, 40, 50) scala> val df2 = data2.toDF("id2")

df2: org.apache.spark.sql.DataFrame = [id2: int]

val dfJoined = df1.join(df2, $"id1" === $"id2")

scala> dfJoined.queryExecution.executedPlan

scala> dfJoined.show



2. Shuffle Hash Join

COMMANDS

scala> spark.conf.set("spark.sql.autoBroadcastJoinThreshold", 2) scala> spark.conf.set("spark.sql.join.preferSortMergeJoin", "false")

scala> spark.conf.get("spark.sql.join.preferSortMergeJoin") res2: String = false

scala> spark.conf.get("spark.sql.autoBroadcastJoinThreshold") res3: String = 2

scala> val data1 = Seq(10, 20, 20, 30, 40, 10, 40, 20, 20, 20, 20, 50)

data1: Seq[Int] = List(10, 20, 20, 30, 40, 10, 40, 20, 20, 20, 20, 50)

scala> val df1 = data1.toDF("id1")

df1: org.apache.spark.sql.DataFrame = [id1: int]

scala> val data2 = Seq(30, 20, 40, 50)

data2: Seq[Int] = List(30, 20, 40, 50)

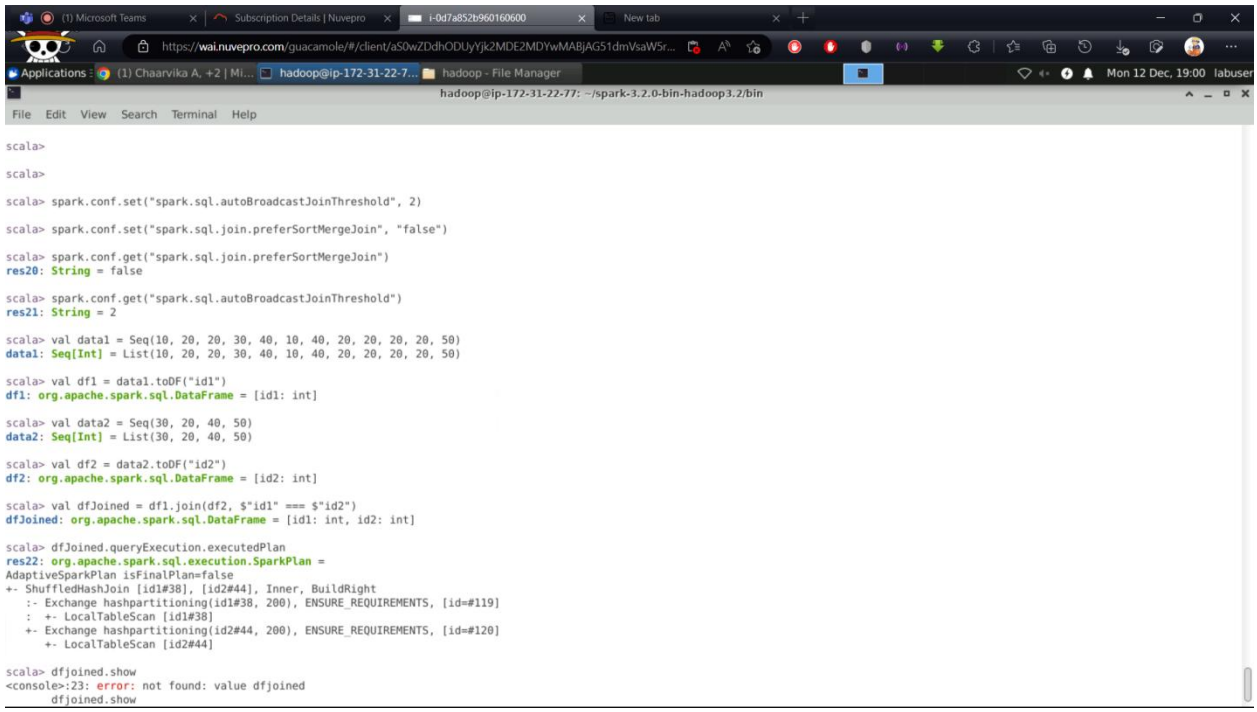
scala> val df2 = data2.toDF("id2")

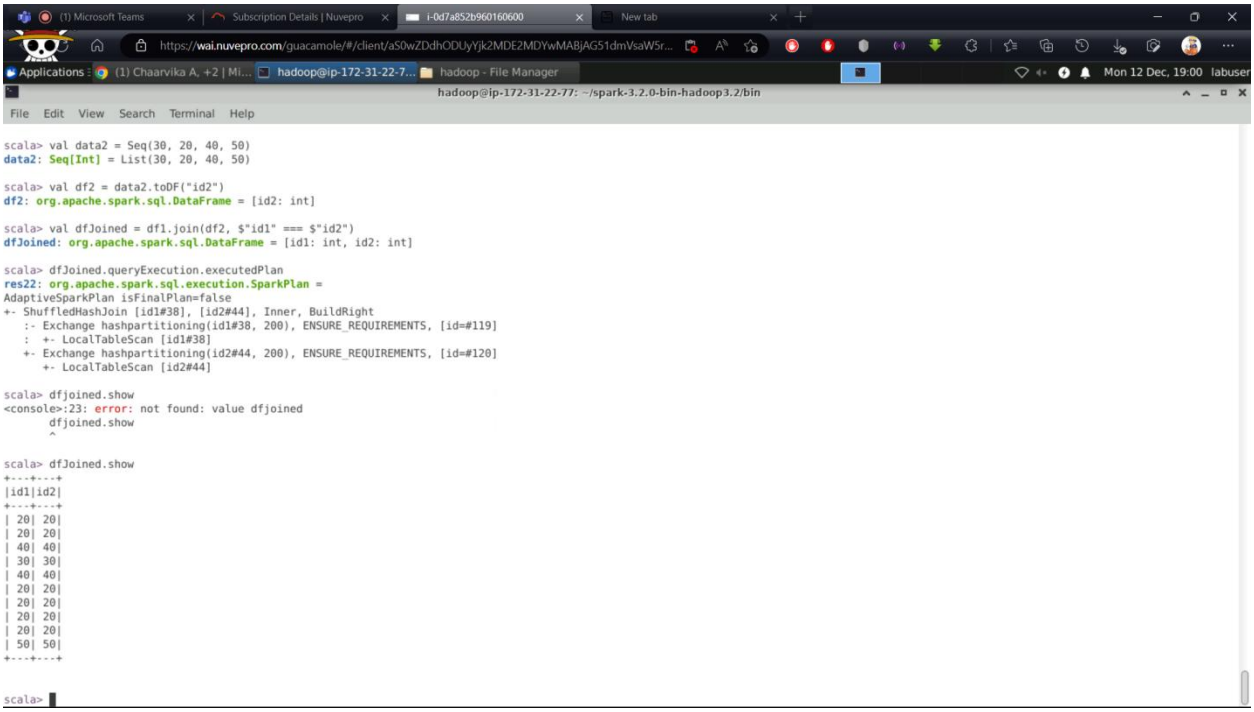
df2: org.apache.spark.sql.DataFrame = [id2: int]

scala> val dfJoined = df1.join(df2, $"id1" === $"id2") dfJoined: org.apache.spark.sql.DataFrame = [id1: int, id2: int]

scala> dfJoined.queryExecution.executedPlan

scala> dfJoined.show





3. CARTESIAN PRODUCT JOIN

COMMANDS

scala> spark.conf.get("spark.sql.join.preferSortMergeJoin") res1: String = true

scala> spark.conf.get("spark.sql.autoBroadcastJoinThreshold") res2: String = -1

scala> val data1 = Seq(10, 20, 20, 30, 40, 10, 40, 20, 20, 20, 20, 50)

data1: Seq[Int] = List(10, 20, 20, 30, 40, 10, 40, 20, 20, 20, 20, 50)

scala> val df1 = data1.toDF("id1")

df1: org.apache.spark.sql.DataFrame = [id1: int] 5.scala> val data2 = Seq(30, 20, 40, 50)

data2: Seq[Int] = List(30, 20, 40, 50) 6.scala> val df2 = data2.toDF("id2")

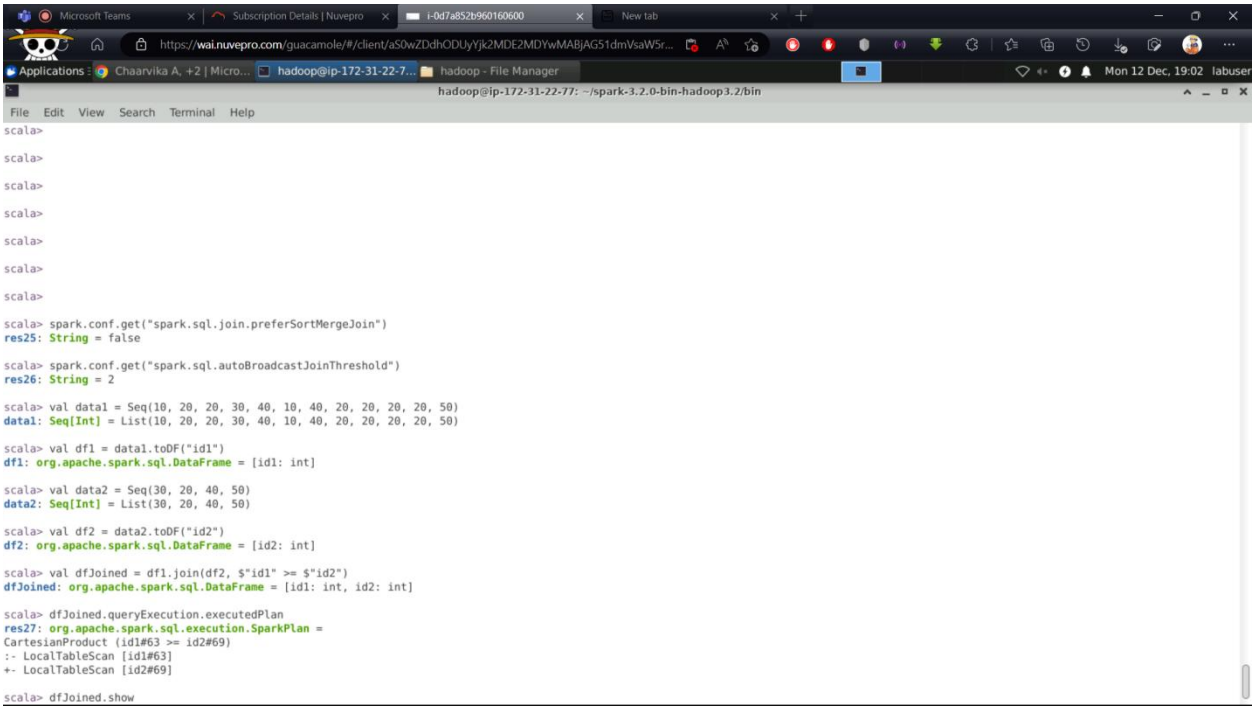
df2: org.apache.spark.sql.DataFrame = [id2: int]

scala> val dfJoined = df1.join(df2, $"id1" >= $"id2")

dfJoined: org.apache.spark.sql.DataFrame = [id1: int, id2: int]

scala> dfJoined.queryExecution.executedPlan

scala> dfJoined.show




Microsoft Teams
Subscription Details | Nuvepro
i-0d7a852b960160600
New tab
https://wai.nuvepro.com/guacamole/#/client/aS0wZDdhODUyYjk2MDE2MDYwMABjAG51dmVsaW5r...
Applications
Chaarvika A, +2 | Micro...
hadoop@ip-172-31-22-7...
hadoop - File Manager
Mon 12 Dec, 19:02 labuser
hadoop@ip-172-31-22-77: ~/spark-3.2.0-bin-hadoop3.2/bin
File Edit View Search Terminal Help
scala> val data2 = Seq(30, 20, 40, 50)
data2: Seq[Int] = List(30, 20, 40, 50)
scala> val df2 = data2.toDF("id2")
df2: org.apache.spark.sql.DataFrame = [id2: int]
scala> val dfJoined = df1.join(df2, $"id1" >= $"id2")
dfJoined: org.apache.spark.sql.DataFrame = [id1: int, id2: int]
scala> dfJoined.queryExecution.executedPlan
res27: org.apache.spark.sql.execution.SparkPlan =
CartesianProduct (id1#63 >= id2#69)
:- LocalTableScan [id1#63]
+- LocalTableScan [id2#69]
scala> dfJoined.show
+---+---+
|id1|id2|
+---+---+
20	20
20	20
30	30
30	20
40	30
40	20
40	40
40	30
40	20
20	20
20	20
20	20
50	30
50	20
40	40
50	40
50	50
+---+---+
scala>