

# Pseudo Node Hadoop Setup

Login 1<sup>st</sup> with u r id & password on VMWare.

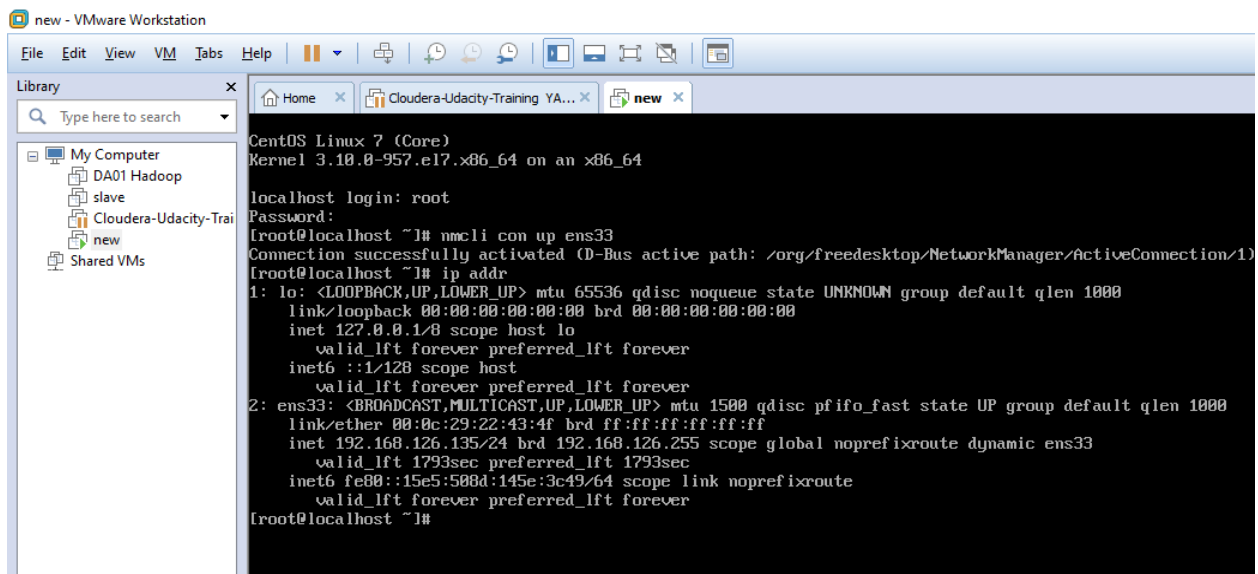
Step 1: check your ip address

```
[root@localhost ~]# nmcli con up ens 33
```

Connenction activated successfully.

```
[root@localhost ~]# ip addr
```

Activate your ip address



```
new - VMware Workstation
File Edit View VM Tabs Help
Library
Type here to search
My Computer
  DA01 Hadoop
  slave
  Cloudera-Udacity-Trai
  new
  Shared VMs
CentOS Linux 7 (Core)
Kernel 3.10.0-957.el7.x86_64 on an x86_64
localhost login: root
Password:
[root@localhost ~]# nmcli con up ens33
Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/1)
[root@localhost ~]# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:22:43:4f brd ff:ff:ff:ff:ff:ff
    inet 192.168.126.135/24 brd 192.168.126.255 scope global noprefixroute dynamic ens33
        valid_lft 1793sec preferred_lft 1793sec
    inet6 fe80::15e5:508d:145e:3c49/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
[root@localhost ~]#
```

Step 1: Download latest jdk from Hadoop setup file

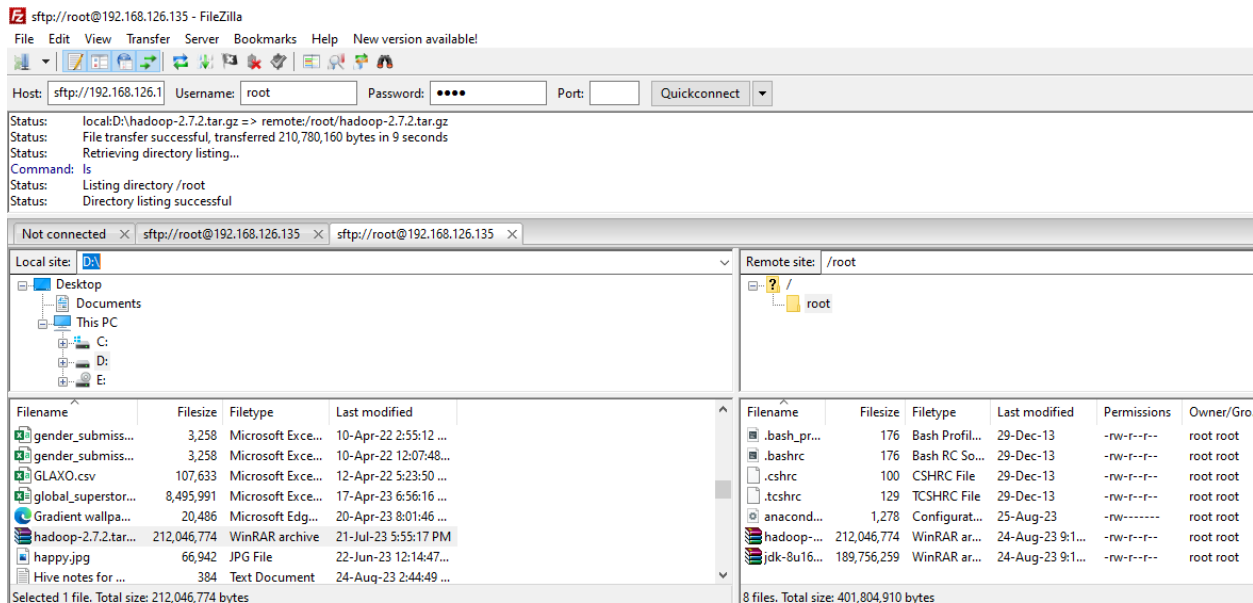
Files name :

jdk-8u161-linux-x64.tar

hadoop-2.7.2.tar

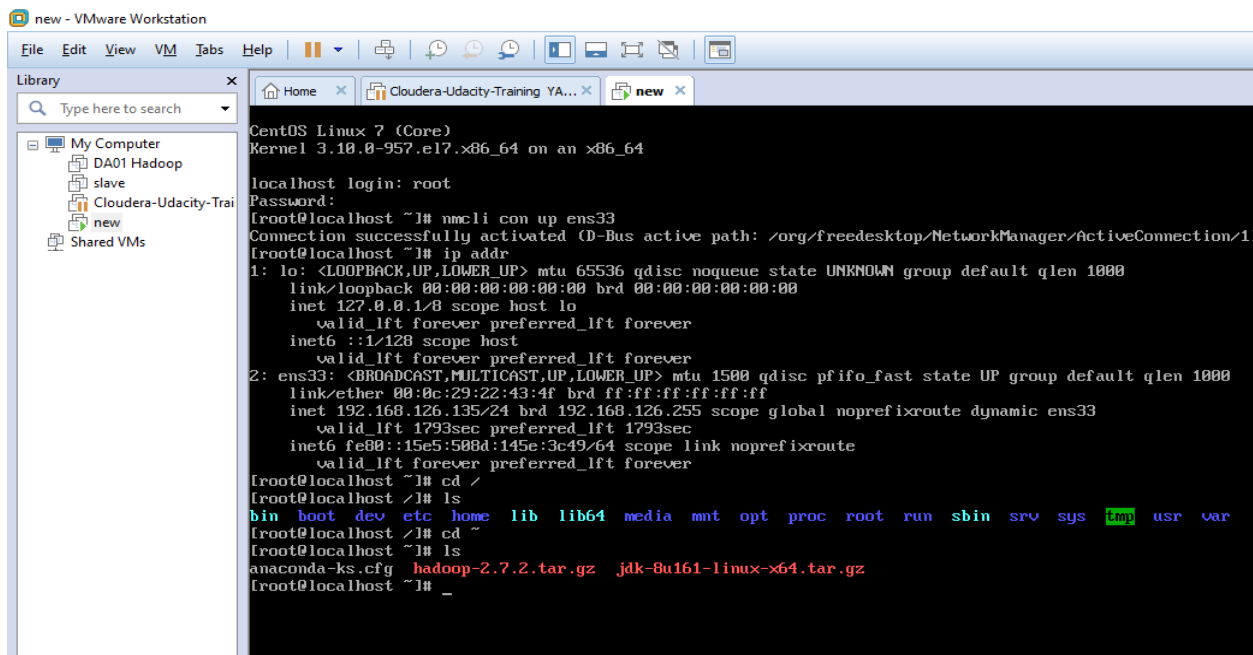
Using filezilla transfer the file from your pc to vmcare workstation

Just drag & drop file from local size pc to root.



[root@localhost~]# ls

Shows all files



# Installation of JDK

## 2: Extract the tar file

Extract JDK file: `tar -xvf jdk..tar.gz`

```
jdk1.8.0_161/jre/lib/jce.jar
jdk1.8.0_161/jre/lib/flashplayer.properties
jdk1.8.0_161/jre/lib/jfxswt.jar
jdk1.8.0_161/jre/lib/fontconfig.SuSE.10.properties.src
jdk1.8.0_161/jre/lib/fontconfig.SuSE.11.bfc
jdk1.8.0_161/jre/COPYRIGHT
jdk1.8.0_161/jre/THIRDPARTYLICENSEREADME-JAVAFX.txt
jdk1.8.0_161/jre/Welcome.html
jdk1.8.0_161/jre/README
jdk1.8.0_161/README.html
[root@localhost ~]# ls
anaconda-ks.cfg  hadoop-2.7.2.tar.gz  jdk1.8.0_161  jdk-8u161-linux-x64.tar.gz
[root@localhost ~]#
```

After Extracted file showing in blue colour & zip file showing in red colour.

### Step 3: Move jdk extracted file dir to /usr/local/java

`mv jdk1.6... /usr/local/java`

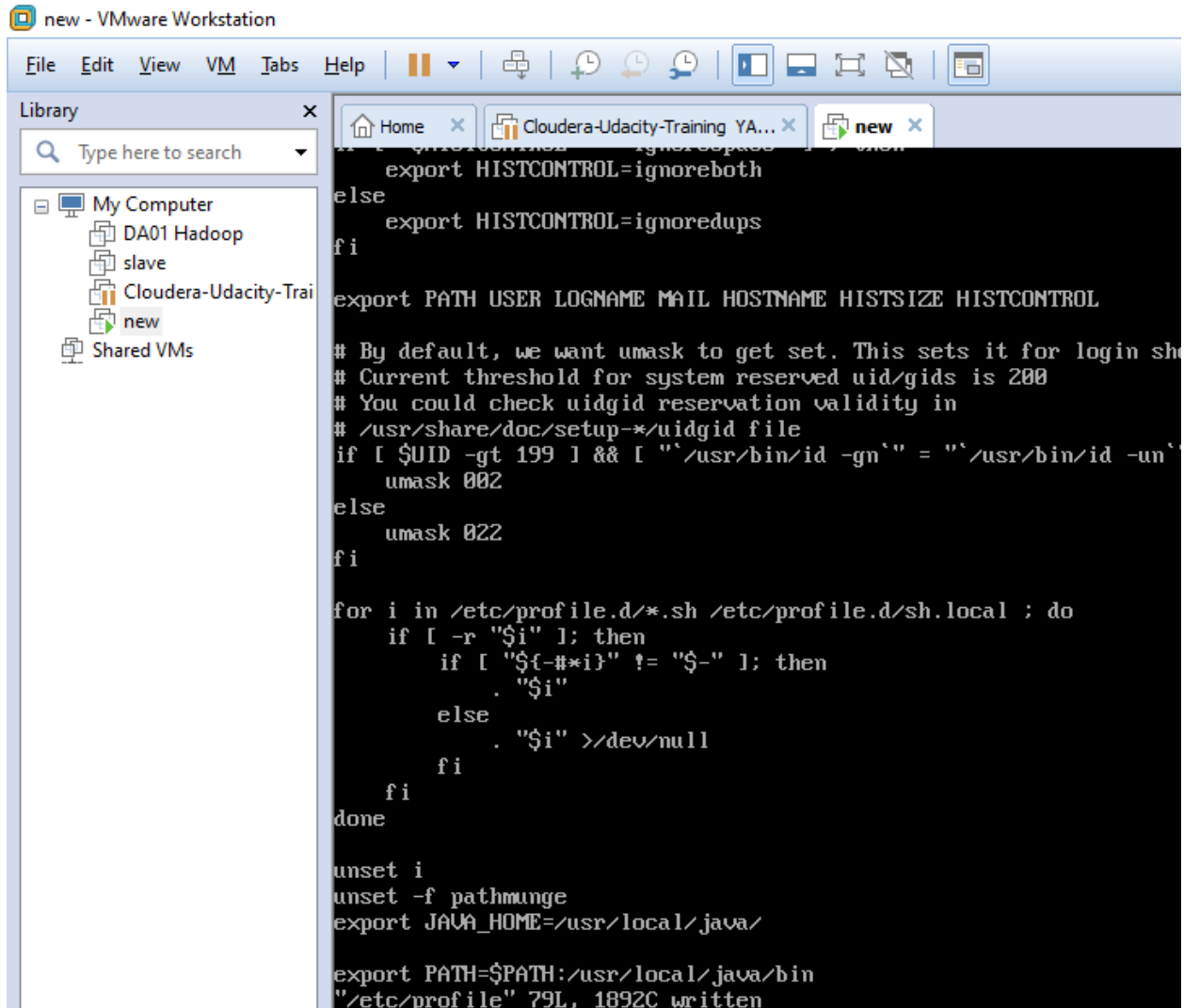
```
jdk1.8.0_161/jre/COPYRIGHT
jdk1.8.0_161/jre/THIRDPARTYLICENSEREADME-JAVAFX.txt
jdk1.8.0_161/jre/Welcome.html
jdk1.8.0_161/jre/README
jdk1.8.0_161/README.html
[root@localhost ~]# ls
anaconda-ks.cfg  hadoop-2.7.2.tar.gz  jdk1.8.0_161  jdk-8u161-linux-x64.tar.gz
[root@localhost ~]# mv jdk1.8.0_161/ /usr/local/java
[root@localhost ~]# ls
anaconda-ks.cfg  hadoop-2.7.2.tar.gz  jdk-8u161-linux-x64.tar.gz
[root@localhost ~]#
```

### Step 4: Set the environment variables

`vi /etc/profile`

inserts this path in the file:

```
export JAVA_HOME=/usr/local/java/
export PATH=$PATH:/usr/local/java/bin
```



### Step 5: Check if jps (java process) is running

source

*[Source is like refresh command in windows]*

Jps

```

unset -f pathmunge
export JAVA_HOME=/usr/local/java/

export PATH=$PATH:/usr/local/java/bin
"/etc/profile" 79L, 1892C written
[root@localhost ~]# source /etc/profile
[root@localhost ~]# jps
7567 Jps

```

# Install Apache Hadoop

## Step 1: Unpack the Apache Hadoop

tar -xvf hadoop-2.6.0.tar.gz

```
umask 002
else
umask 022
fi

for i in /etc/profile.d/*.sh /etc/profile.d/sh.local
do
if [ -r "$i" ]; then
if [ "${-#*i}" != "$-" ]; then
. "$i"
else
. "$i" >/dev/null
fi
fi
done

unset i
unset -f pathmunge
export JAVA_HOME=/usr/local/java/

export PATH=$PATH:/usr/local/java/bin
"/etc/profile" 79L, 1892C written
[root@localhost ~]# source /etc/profile
[root@localhost ~]# jps
7567 Jps
[root@localhost ~]# tar -xvf hadoop-2.7.2.tar.gz
```

## Step 2: Move hadoop extracted file dir to /usr/local/hadoop

```
Home x Cloudera-Udacity-Training YA... x new x
hadoop-2.7.2/share/hadoop/common/lib/avro-1.7.4.jar
hadoop-2.7.2/share/hadoop/common/lib/commons-beanutils-core-1.8.0.jar
hadoop-2.7.2/share/hadoop/common/lib/servlet-api-2.5.jar
hadoop-2.7.2/share/hadoop/common/lib/api-asn1-api-1.0.0-M20.jar
hadoop-2.7.2/share/hadoop/common/lib/gson-2.2.4.jar
hadoop-2.7.2/share/hadoop/common/lib/commons-cli-1.2.jar
hadoop-2.7.2/share/hadoop/common/lib/junit-4.11.jar
hadoop-2.7.2/share/hadoop/common/lib/jettison-1.1.jar
hadoop-2.7.2/share/hadoop/common/lib/jsr305-3.0.0.jar
hadoop-2.7.2/share/hadoop/common/lib/commons-logging-1.1.3.jar
hadoop-2.7.2/share/hadoop/common/lib/slf4j-log4j12-1.7.10.jar
hadoop-2.7.2/share/hadoop/common/lib/hamcrest-core-1.3.jar
hadoop-2.7.2/share/hadoop/common/lib/slf4j-api-1.7.10.jar
hadoop-2.7.2/share/hadoop/common/lib/commons-httpclient-3.1.jar
hadoop-2.7.2/share/hadoop/common/lib/commons-beanutils-1.7.0.jar
hadoop-2.7.2/share/hadoop/common/lib/paranamer-2.3.jar
hadoop-2.7.2/share/hadoop/common/hadoop-common-2.7.2-tests.jar
hadoop-2.7.2/share/hadoop/common/sources/
hadoop-2.7.2/share/hadoop/common/sources/hadoop-common-2.7.2-sources.jar
hadoop-2.7.2/share/hadoop/common/sources/hadoop-common-2.7.2-test-sources.jar
hadoop-2.7.2/lib/
hadoop-2.7.2/lib/native/
hadoop-2.7.2/lib/native/libhdfs.so
hadoop-2.7.2/lib/native/libhadooputils.a
hadoop-2.7.2/lib/native/libhdfs.so.0.0.0
hadoop-2.7.2/lib/native/libhadoop.so.1.0.0
hadoop-2.7.2/lib/native/libhadoop.a
hadoop-2.7.2/lib/native/libhdfs.a
hadoop-2.7.2/lib/native/libhadoop.so
hadoop-2.7.2/lib/native/libhadooppipes.a
hadoop-2.7.2/LICENSE.txt
[root@localhost ~]# ls
anaconda-ks.cfg  hadoop-2.7.2  hadoop-2.7.2.tar.gz  jdk-8u161-linux-x64.tar.gz
[root@localhost ~]# mv hadoop-2.7.2/ /usr/local/hadoop
[root@localhost ~]# ls
anaconda-ks.cfg  hadoop-2.7.2.tar.gz  jdk-8u161-linux-x64.tar.gz
[root@localhost ~]#
```

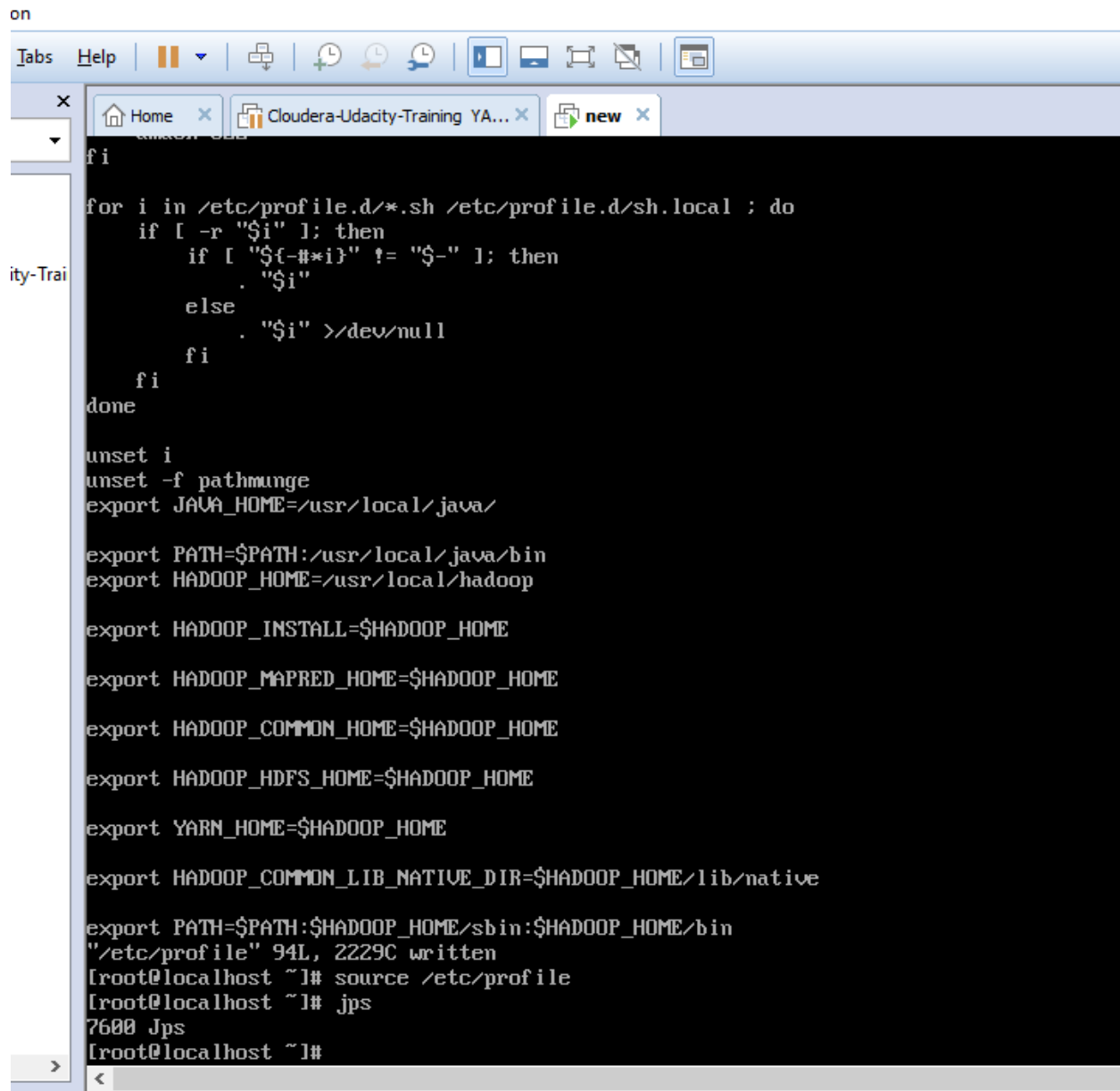
#### Step 4: Set the environment Variables

- vi /etc/profile [we are using this file]

Inserts this path in 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
```

on



```
fi
for i in /etc/profile.d/*.sh /etc/profile.d/sh.local ; do
    if [ -r "$i" ]; then
        if [ "${-#*i}" != "$-" ]; then
            . "$i"
        else
            . "$i" >>/dev/null
        fi
    fi
done

unset i
unset -f pathmunge
export JAVA_HOME=/usr/local/java/

export PATH=$PATH:/usr/local/java/bin
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
"/etc/profile" 94L, 2229C written
[root@localhost ~]# source /etc/profile
[root@localhost ~]# jps
7600 Jps
[root@localhost ~]#
```

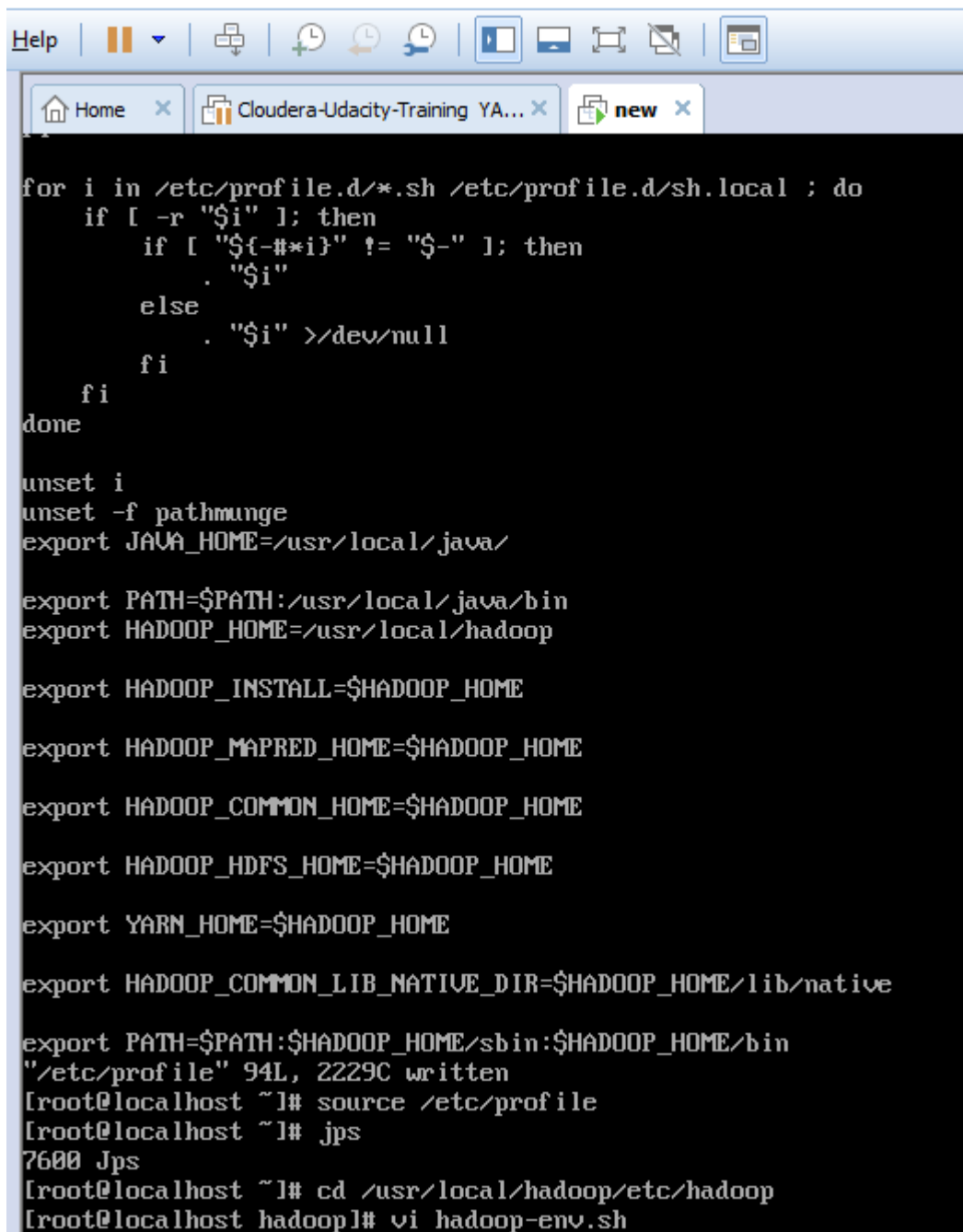
### Step 5: Set JAVA path in `hadoop-env.sh`

`cd /usr/local/hadoop/etc/Hadoop`

`vi hadoop-env.sh`

insert the path in the file:

`export JAVA_HOME=/usr/local/java`



```
for i in /etc/profile.d/*.sh /etc/profile.d/sh.local ; do
    if [ -r "$i" ]; then
        if [ "${-#*i}" != "$-" ]; then
            . "$i"
        else
            . "$i" >/dev/null
        fi
    fi
done

unset i
unset -f pathmunge
export JAVA_HOME=/usr/local/java/

export PATH=$PATH:/usr/local/java/bin
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
"/etc/profile" 94L, 2229C written
[root@localhost ~]# source /etc/profile
[root@localhost ~]# jps
7600 Jps
[root@localhost ~]# cd /usr/local/hadoop/etc/hadoop
[root@localhost hadoop]# vi hadoop-env.sh
```

## Step 6: Configure the xml files –

The properties need to be copied between <configuration> and </configuration> tag.

**core-site.xml**





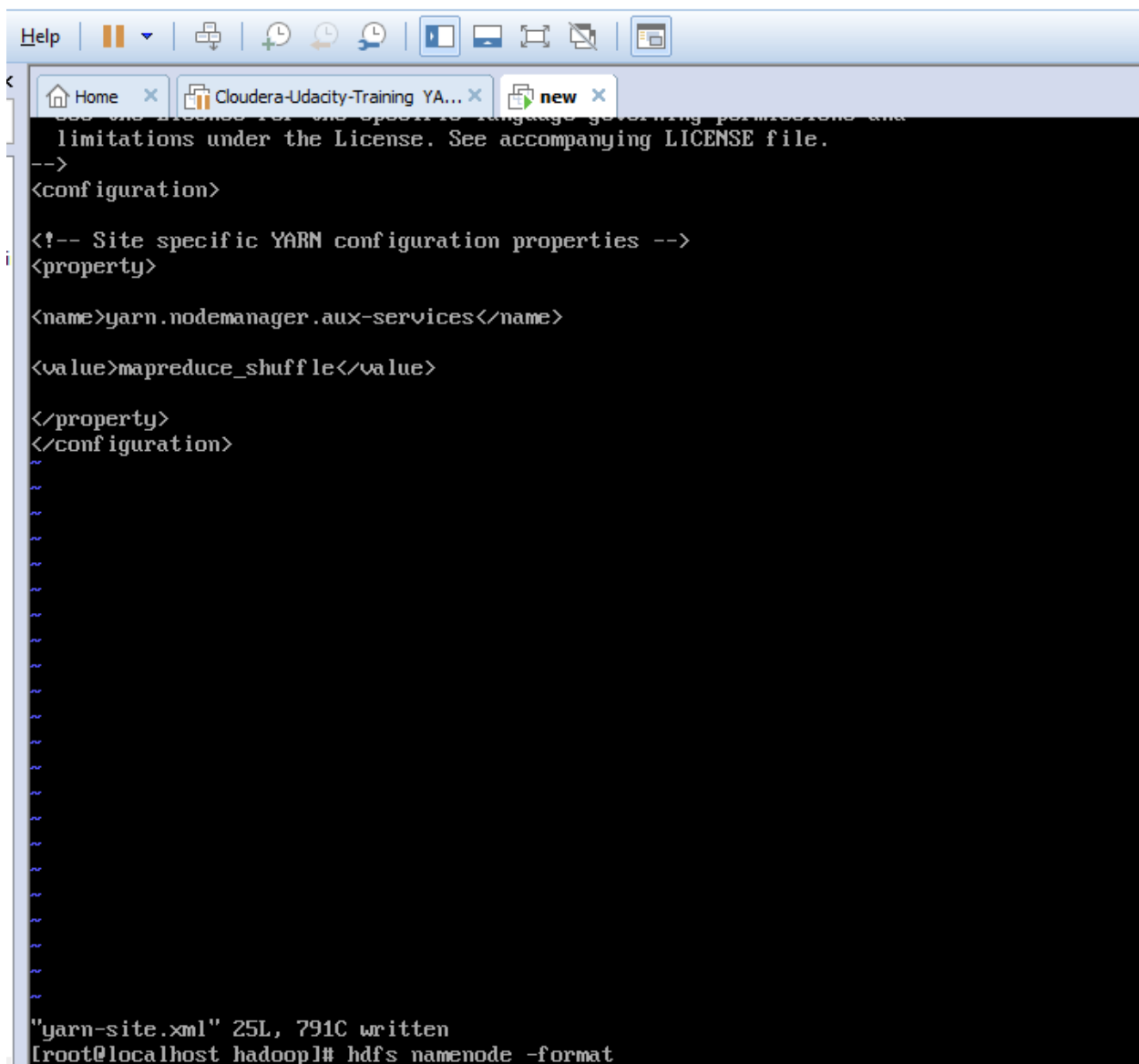


```
"hdfs-site.xml" 43L, 1024C written
[root@localhost hadoop]# mv mapred-site.xml.template mapred-site.xml
[root@localhost hadoop]# ls
capacity-scheduler.xml  hadoop-env.sh          https-env.sh           kms-env.sh             mapred-env.sh          ssl-server.xml
configuration.xml       hadoop-metrics2.properties  https-log4j.properties  kms-log4j.properties  mapred-queues.xml.template  yarn-env.cmd
container-executor.cfg  hadoop-metrics.properties  https-signature.secret  kms-site.xml           mapred-site.xml        yarn-env.sh
core-site.xml           hadoop-policy.xml         https-site.xml          log4j.properties      slaves                  yarn-site.xml
hadoop-env.cmd          hdfs-site.xml            kms-acls.xml            mapred-env.cmd         ssl-client.xml.example
[root@localhost hadoop]#
```

le or press Ctrl+G.







The screenshot shows a terminal window with a blue title bar. The window has three tabs: 'Home', 'Cloudera-Udacity-Training YA...', and 'new'. The terminal content is as follows:

```
See the license for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->
<configuration>

<!-- Site specific YARN configuration properties -->
<property>

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

<value>mapreduce_shuffle</value>

</property>
</configuration>

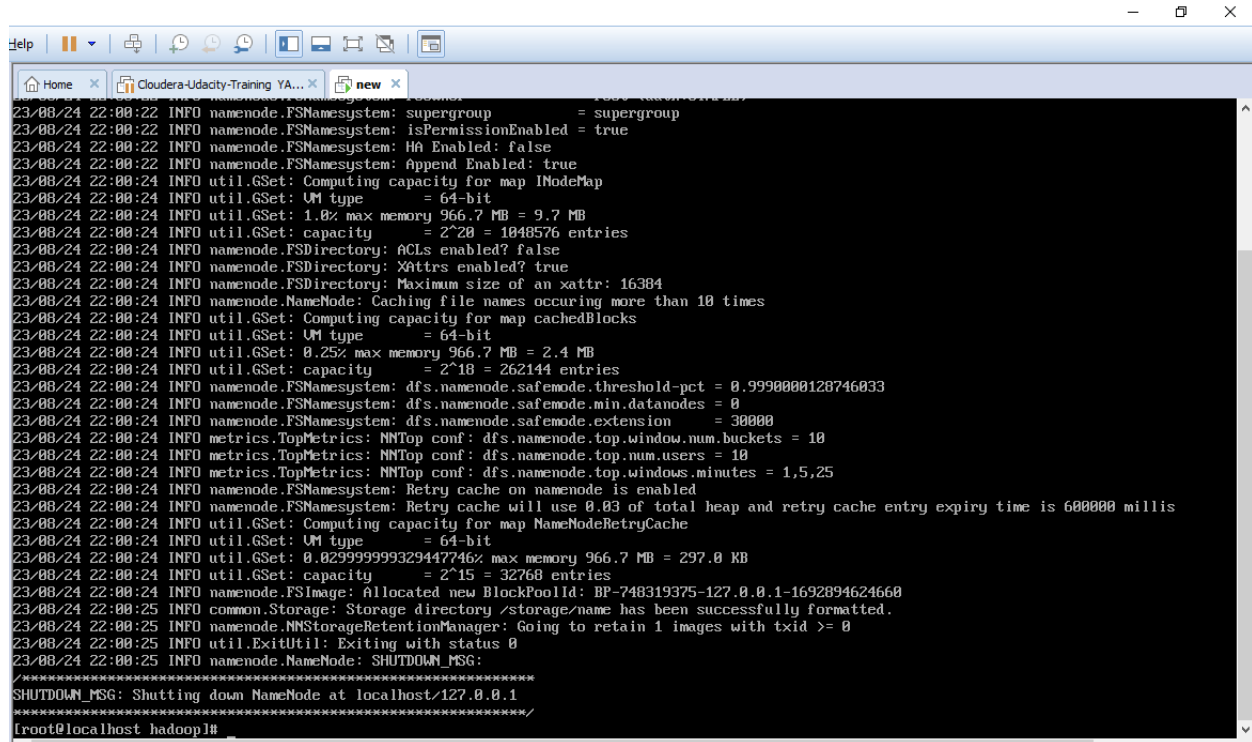
```

Below the XML content, there are several lines of output:

```
"yarn-site.xml" 25L, 791C written
[root@localhost hadoop]# hdfs namenode -format
```

## Step 7: Format the Namenode

hdfs namenode -format

A screenshot of a terminal window with a blue title bar. The window contains a series of log messages from the Hadoop NameNode. The logs show various configuration parameters and status updates. At the bottom, there is a 'SHUTDOWN\_MSG' indicating the NameNode is shutting down at localhost/127.0.0.1. The terminal prompt is '[root@localhost hadoop]#'.

```
23/08/24 22:00:22 INFO namenode.FSNamesystem: supergroup = supergroup
23/08/24 22:00:22 INFO namenode.FSNamesystem: isPermissionEnabled = true
23/08/24 22:00:22 INFO namenode.FSNamesystem: HA Enabled: false
23/08/24 22:00:22 INFO namenode.FSNamesystem: Append Enabled: true
23/08/24 22:00:24 INFO util.GSet: Computing capacity for map INodeMap
23/08/24 22:00:24 INFO util.GSet: UM type = 64-bit
23/08/24 22:00:24 INFO util.GSet: 1.0% max memory 966.7 MB = 9.7 MB
23/08/24 22:00:24 INFO util.GSet: capacity = 2^20 = 1048576 entries
23/08/24 22:00:24 INFO namenode.FSDirectory: ACLs enabled? false
23/08/24 22:00:24 INFO namenode.FSDirectory: XAttrs enabled? true
23/08/24 22:00:24 INFO namenode.FSDirectory: Maximum size of an xattr: 16384
23/08/24 22:00:24 INFO namenode.NameNode: Caching file names occurring more than 10 times
23/08/24 22:00:24 INFO util.GSet: Computing capacity for map cachedBlocks
23/08/24 22:00:24 INFO util.GSet: UM type = 64-bit
23/08/24 22:00:24 INFO util.GSet: 0.25% max memory 966.7 MB = 2.4 MB
23/08/24 22:00:24 INFO util.GSet: capacity = 2^18 = 262144 entries
23/08/24 22:00:24 INFO namenode.FSNamesystem: dfs.namenode.safemode.threshold-pct = 0.9990000128746033
23/08/24 22:00:24 INFO namenode.FSNamesystem: dfs.namenode.safemode.min.datanodes = 0
23/08/24 22:00:24 INFO namenode.FSNamesystem: dfs.namenode.safemode.extension = 30000
23/08/24 22:00:24 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.window.num.buckets = 10
23/08/24 22:00:24 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.num.users = 10
23/08/24 22:00:24 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.windows.minutes = 1,5,25
23/08/24 22:00:24 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
23/08/24 22:00:24 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total heap and retry cache entry expiry time is 600000 millis
23/08/24 22:00:24 INFO util.GSet: Computing capacity for map NameNodeRetryCache
23/08/24 22:00:24 INFO util.GSet: UM type = 64-bit
23/08/24 22:00:24 INFO util.GSet: 0.029999999329447746% max memory 966.7 MB = 297.0 KB
23/08/24 22:00:24 INFO util.GSet: capacity = 2^15 = 32768 entries
23/08/24 22:00:24 INFO namenode.FSImage: Allocated new BlockPoolId: BP-748319375-127.0.0.1-1692894624660
23/08/24 22:00:25 INFO common.Storage: Storage directory /storage/name has been successfully formatted.
23/08/24 22:00:25 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
23/08/24 22:00:25 INFO util.ExitUtil: Exiting with status 0
23/08/24 22:00:25 INFO namenode.NameNode: SHUTDOWN_MSG:
/*****
SHUTDOWN_MSG: Shutting down NameNode at localhost/127.0.0.1
*****/
[root@localhost hadoop]#
```

## Step 8: Go to sbin

We need to start the services now

```
cd /usr/local/hadoop/sbin
```

## Step 9: Start dfs components

```
sh start-dfs.sh
```

HDFS Services/Processes:

**Name node**

**Secondary name node**

**Data node**

```

*****
[root@localhost hadoop]# cd /usr/local/hadoop/sbin
[root@localhost sbin]# sh start-dfs.sh
23/08/24 22:05:44 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applic
Starting namenodes on [localhost]
The authenticity of host 'localhost (::1)' can't be established.
ECDSA key fingerprint is SHA256:J175ha6A23teAB/S00q2M/xdLrGBJ1o9DUTZECHnQ+0.
ECDSA key fingerprint is MD5:c4:12:3b:a4:26:e4:4f:60:5e:e1:07:8e:45:e6:09:5e.
Are you sure you want to continue connecting (yes/no)? yes
localhost: Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
root@localhost's password:
localhost: starting namenode, logging to /usr/local/hadoop/logs/hadoop-root-namenode-localhost.localdomain.out
root@localhost's password:
localhost: starting datanode, logging to /usr/local/hadoop/logs/hadoop-root-datanode-localhost.localdomain.out
Starting secondary namenodes [0.0.0.0]
The authenticity of host '0.0.0.0 (0.0.0.0)' can't be established.
ECDSA key fingerprint is SHA256:J175ha6A23teAB/S00q2M/xdLrGBJ1o9DUTZECHnQ+0.
ECDSA key fingerprint is MD5:c4:12:3b:a4:26:e4:4f:60:5e:e1:07:8e:45:e6:09:5e.
Are you sure you want to continue connecting (yes/no)? yes
0.0.0.0: Warning: Permanently added '0.0.0.0' (ECDSA) to the list of known hosts.
root@0.0.0.0's password:
0.0.0.0: starting secondarynamenode, logging to /usr/local/hadoop/logs/hadoop-root-secondarynamenode-localhost.localdomain.out
23/08/24 22:07:00 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applic
[root@localhost sbin]# jps
7860 DataNode
8140 Jps
8039 SecondaryNameNode
7770 NameNode
[root@localhost sbin]#

```

## Step 10: Start YARN components

**sh start-yarn.sh**

YARN Services/Processes:

**resource manager**

**node manager**

```

[root@localhost sbin]# sh start-yarn.sh
starting yarn daemons
starting resourcemanager, logging to /usr/local/hadoop/logs/yarn-root-resourcemanager-localhost.localdomain.out
root@localhost's password:
localhost: starting nodemanager, logging to /usr/local/hadoop/logs/yarn-root-nodemanager-localhost.localdomain.out
[root@localhost sbin]# jps
7860 DataNode
8500 Jps
8197 ResourceManager
8293 NodeManager
8039 SecondaryNameNode
7770 NameNode
[root@localhost sbin]#

```