

Tutorial 2

Hadoop Distributed File System (HDFS) on Ubuntu

All commands are case sensitive on Ubuntu operating system

- 1) Start the terminal by writing on the search box or Press **Ctrl + Alt + t** together to open a terminal as shown below
 - **Note: \$ sign shows the cursor on the ubuntu shell, do not write with commands**
- 2) Update the repository in Ubuntu by using the following command
\$sudo apt update
- 3) Now purge the java installations by using the following command
\$sudo apt purge openjdk*
If the JAVA is not installed, you will get a message that Java is not installed. First install java on Ubuntu OS.
- 4) Install JAVA (version jdk 8) by using this command, If you get the option for **yes/ no**, type **yes**
\$sudo apt install openjdk-8-jdk
\$sudo apt install rsync
- 5) The above process takes some time to install java depending on the system architecture. You can check installation of java by using this command, **java -version**
- 6) Update the Operating system after JAVA installation as
\$sudo apt update
- 7) Run the following to check where Java is installed or not properly
\$sudo update-alternatives --config java
It will ask you two options and press the <Enter> key for default option as mentioned below
/usr/lib/jvm/java-8-openjdkamd64/jre/bin/java
- 8) Now we set the path that Hadoop finds the java on Ubuntu by using Linux '**nano**' editor [Check how **nano** editor working on <https://www.hostinger.com/tutorials/how-to-install-and-use-nano-text-editor>]
\$sudo nano /etc/profile
- 9) A file will be opened in the **nano editor** (All users who log in to the bash or sh shells use it. The PATH variable, user restrictions, and other user settings are typically defined in **profile** file. This file is only run for login shell and therefore does not run when a script is executed.) and set the following path at the end of this file
export JAVA_HOME=/usr
After writing the above path in the **profile** file, press **ctrl + x** to exit from the editor, write **y** to save all the updates in the file, then press the **Enter key**
[**export**- command is one of the bash shell BUILTINS commands, which means it is part of your shell.]
- 10) Set this file as the source as
\$source /etc/profile
While **source** is a shell built-in command which is used to read and execute the content of a file in a current session after update.
- 11) Disable ipv6 because Hadoop supports only ipv4 generally. Open the file (**sysctl.conf**) by using the command
\$sudo nano /etc/sysctl.conf
- 12) Move the cursor down to the end of the file and append the following three lines (careful about spaces)
net.ipv6.conf.all.disable_ipv6 = 1
net.ipv6.conf.default.disable_ipv6 = 1
net.ipv6.conf.lo.disable_ipv6 = 1

- 13) Reboot the system by using the following command

```
$sudo reboot
```

- 14) Now we need to configure **SSH keys (secure shell)** to run the Hadoop. For this, we will create another user named as "**hduser**" in **hadoopgroup** group. First create the Hadoop group

```
$sudo addgroup hadoopgroup
```

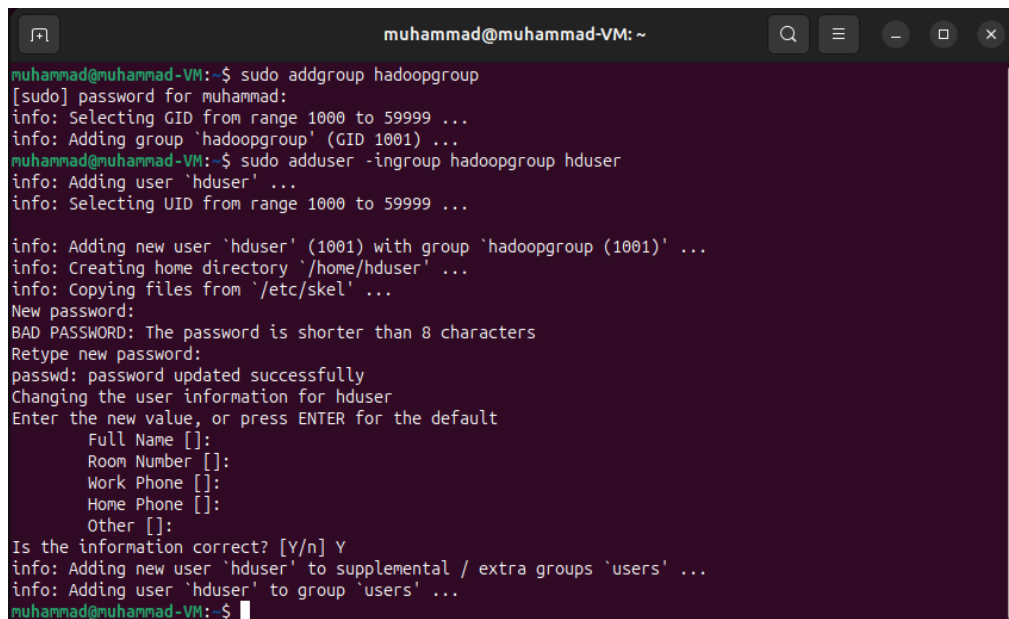
[The command in step (14) adds a new user group to your system, called as **hadoopgroup**]

Now add another user named as **hduser** to the **hadoopgroup**.

```
$sudo adduser -ingroup hadoopgroup hduser
```

It will ask you to follow information as mentioned on the screen. You can use password of your choice (**caution:** use three- or four-letter password).

- Also add hduser user as a super user with admin privileges using the command
- **\$sudo adduser hduser sudo**



```
muhammad@muhammad-VM: ~
muhammad@muhammad-VM:~$ sudo addgroup hadoopgroup
[sudo] password for muhammad:
info: Selecting GID from range 1000 to 59999 ...
info: Adding group 'hadoopgroup' (GID 1001) ...
muhammad@muhammad-VM:~$ sudo adduser -ingroup hadoopgroup hduser
info: Adding user 'hduser' ...
info: Selecting UID from range 1000 to 59999 ...

info: Adding new user 'hduser' (1001) with group 'hadoopgroup (1001)' ...
info: Creating home directory '/home/hduser' ...
info: Copying files from '/etc/skel' ...
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: password updated successfully
Changing the user information for hduser
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] Y
info: Adding new user 'hduser' to supplemental / extra groups 'users' ...
info: Adding user 'hduser' to group 'users' ...
muhammad@muhammad-VM:~$
```

You can leave the Full Name, Room Number, Work Phone, Home Phone and Other or add the details of your choice.

- 15) Install **ssh** by using the following command

```
$sudo apt install ssh
```

[Secure Shell (SSH) is a cryptographic network protocol for operating network services securely over an unsecured network. There are master nodes and slave nodes when a Hadoop cluster is constructed. The slave nodes' tasks are managed by the master node. SSH is used to maintain a connection between these nodes, each of which is a unique system. SSH is primarily used to maintain communication between the master and slave nodes.]

- 16) Enable **ssh** by using

```
$sudo systemctl enable ssh
```

- 17) Start **ssh** by using

```
$sudo systemctl start ssh
```

- 18) Switch to the already created new user, **hduser** by using

```
$su - hduser
```

and use the password as you set during the creation of this user.

[The Unix command **su**, described as substitute user, super user, switch user, or set user, is used by a computer user to execute commands with the privileges of another user account.]

- 19) Generate the key by using

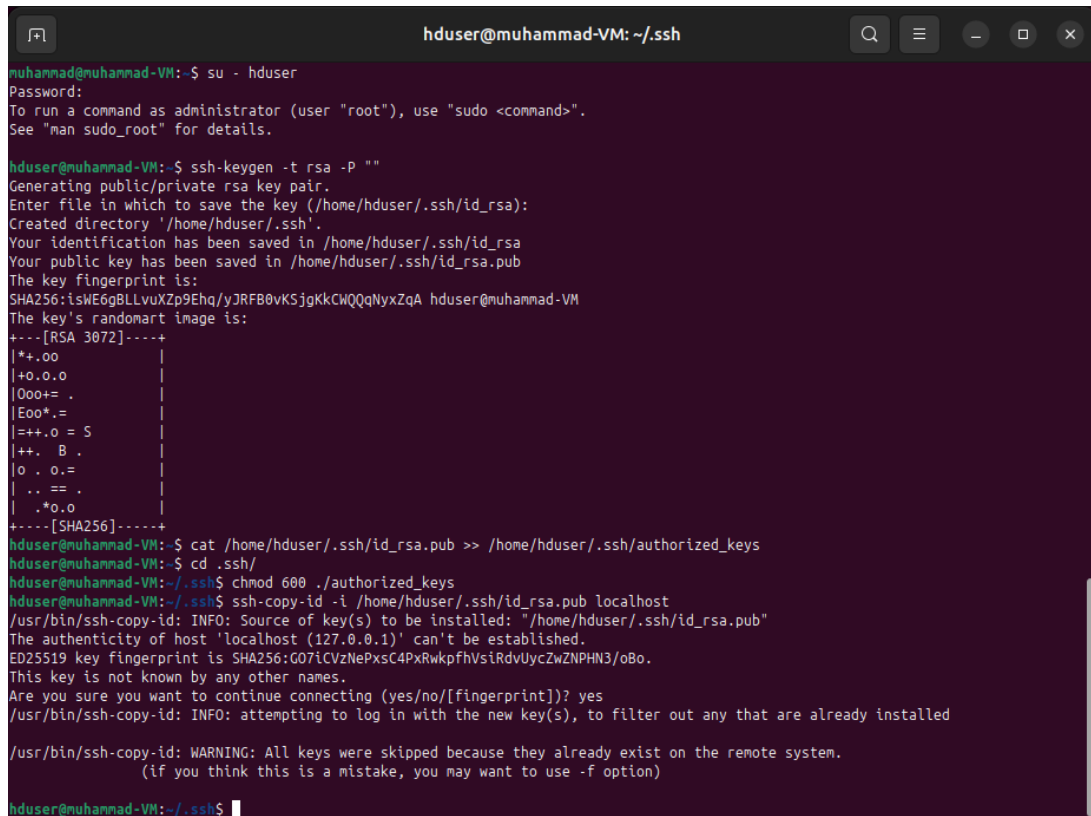
```
$ssh-keygen -t rsa -P ""
```

Hit Enter Key two times

The generated key will be generated in the same folder in the file 'id_rsa.pub'. copy the key into file named as 'authorized_keys' for safe purpose. *Please type the following command rather than copy and paste.*

```
$cat /home/hduser/.ssh/id_rsa.pub >> /home/hduser/.ssh/authorized_keys
```

[The cat command allows you to create single or multiple files, view contain of file, concatenate files and redirect output in terminal or files.]



```

hduser@muhammad-VM: ~/.ssh
muhammad@muhammad-VM:~$ su - hduser
Password:
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

hduser@muhammad-VM:~$ ssh-keygen -t rsa -P ""
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hduser/.ssh/id_rsa):
Created directory '/home/hduser/.ssh'.
Your identification has been saved in /home/hduser/.ssh/id_rsa
Your public key has been saved in /home/hduser/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:1sWE6gBLVvuXZp9Ehq/yJRFB0vKSjgKkCWQQqNyxZqA hduser@muhammad-VM
The key's randomart image is:
+---[RSA 3072]-----+
|*+.00|
|+0.0.0|
|000+=.|
|Eoo*.=|
|==+.0 = S|
|++ . B .|
|o . 0.=|
|.. ==.|
| .*o.o|
+---[SHA256]-----+
hduser@muhammad-VM:~$ cat /home/hduser/.ssh/id_rsa.pub >> /home/hduser/.ssh/authorized_keys
hduser@muhammad-VM:~$ cd .ssh/
hduser@muhammad-VM:~/.ssh$ chmod 600 ./authorized_keys
hduser@muhammad-VM:~/.ssh$ ssh-copy-id -i /home/hduser/.ssh/id_rsa.pub localhost
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/hduser/.ssh/id_rsa.pub"
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ED25519 key fingerprint is SHA256:G07iCVzNePxsC4PxRwkpFhVsiRdvUycZwZNP3/oBo.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: WARNING: All keys were skipped because they already exist on the remote system.
(if you think this is a mistake, you may want to use -f option)
hduser@muhammad-VM:~/.ssh$

```

- 20) **ssh-keygen** generates, manages and converts authentication keys for ssh(1). **ssh-keygen** can create RSA keys for use by SSH protocol version 1 and RSA (Rivest-Shamir-Adleman encryption) or DSA (Digital Signature Algorithm) keys for use by SSH protocol version 2.
- 21) You can check all authorized keys in the directory by using the following commands


```

$cd .ssh/
$chmod 600 ./authorized_keys
$ssh-copy-id -i /home/hduser/.ssh/id_rsa.pub localhost

```

 and if some option asks by the Ubuntu OS, then press **yes**.
 [chmod permissions of 600 mean that the owner has full read and write access to the file, while no other user can access the file.]
- 22) For testing purpose


```

$cd ..
$ssh localhost, after successful execution of this command and then write
$exit command

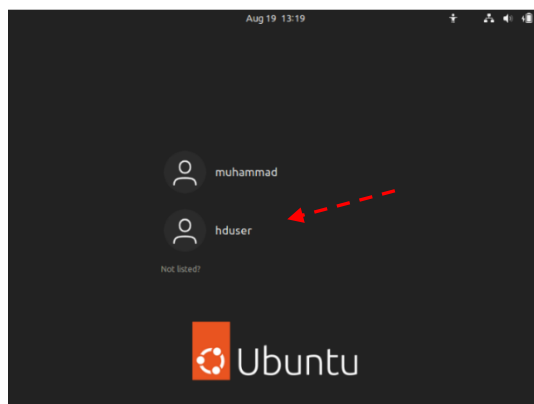
```

```
hduser@muhammad-VM: ~  
hduser@muhammad-VM:~/.ssh$ cd ..  
hduser@muhammad-VM:~$ ssh localhost  
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-40-generic x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:        https://ubuntu.com/pro  
  
Expanded Security Maintenance for Applications is not enabled.  
  
91 updates can be applied immediately.  
To see these additional updates run: apt list --upgradable  
  
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
hduser@muhammad-VM:~$ exit  
logout  
Connection to localhost closed.  
hduser@muhammad-VM:~$
```

The environment in Ubuntu OS is ready for Hadoop distributed file system (hdfs) and now we start Hadoop installation after completion of this setup. Move to the main directory by using the following command as

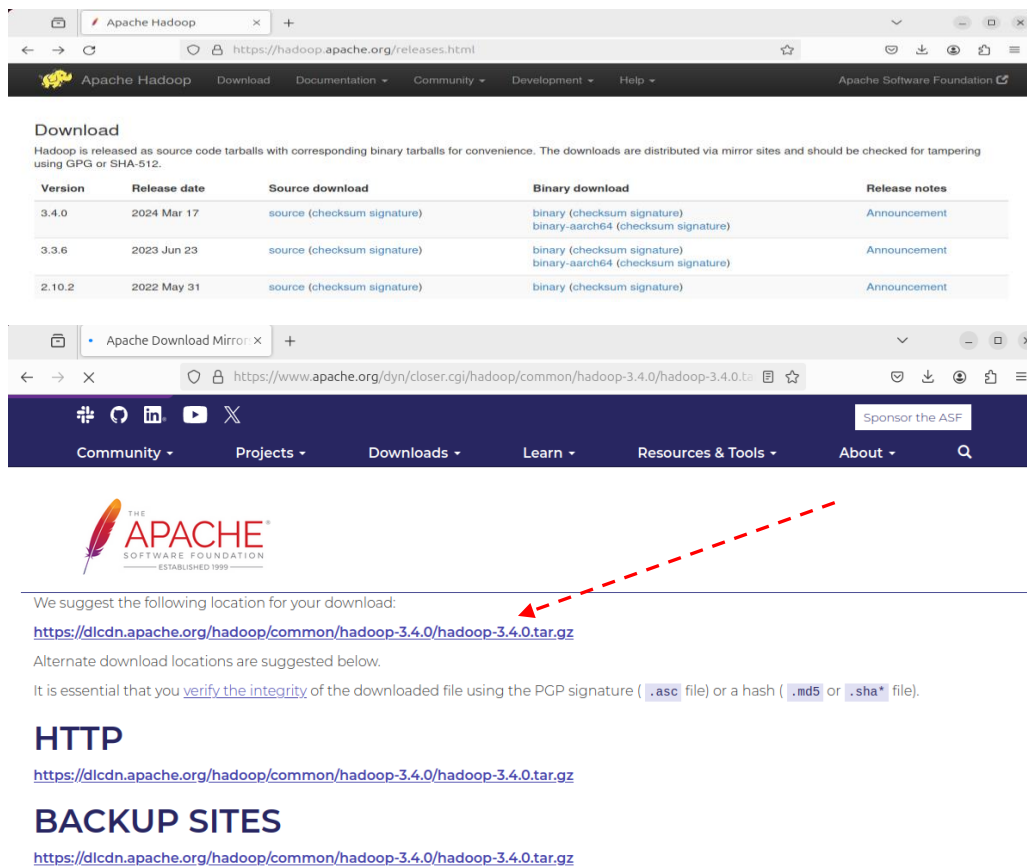
\$cd Hit the Enter key after writing **cd** command
\$sudo reboot

And login in **hduser** using your **password**.



It is new username “**hduser**” and see the resolution of the screen for this user again as you did for your username, like “muhammad”.

- 23) First, we download the Hadoop package by opening Mozilla Firefox browser in the ubuntu OS running in Oracle virtual box.
- 24) Open the website address, **hadoop.apache.org** and click on download link. Then download the binary package, 3.4.0 (available on 17th March 2024).
- 25) Copy the link location for binary file as mentioned below



Download

Hadoop is released as source code tarballs with corresponding binary tarballs for convenience. The downloads are distributed via mirror sites and should be checked for tampering using GPG or SHA-512.

Version	Release date	Source download	Binary download	Release notes
3.4.0	2024 Mar 17	source (checksum signature)	binary (checksum signature) binary-aarch64 (checksum signature)	Announcement
3.3.6	2023 Jun 23	source (checksum signature)	binary (checksum signature) binary-aarch64 (checksum signature)	Announcement
2.10.2	2022 May 31	source (checksum signature)	binary (checksum signature)	Announcement

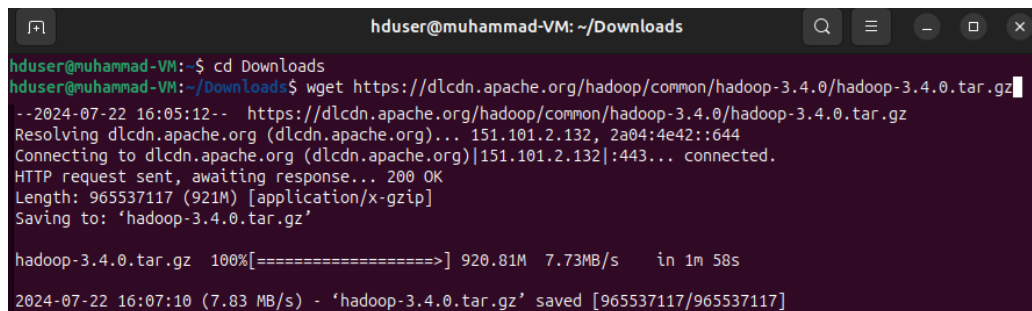
We suggest the following location for your download:
<https://d1cdn.apache.org/hadoop/common/hadoop-3.4.0/hadoop-3.4.0.tar.gz>
 Alternate download locations are suggested below.
 It is essential that you [verify the integrity](#) of the downloaded file using the PGP signature (`.asc` file) or a hash (`.md5` or `.sha*` file).

HTTP
<https://d1cdn.apache.org/hadoop/common/hadoop-3.4.0/hadoop-3.4.0.tar.gz>

BACKUP SITES
<https://d1cdn.apache.org/hadoop/common/hadoop-3.4.0/hadoop-3.4.0.tar.gz>

- 26) If you face difficulty in using Mozilla Firefox browser in download, then you can also use the command **wget** on the terminal as mentioned below in the screenshot.

```
$cd Downloads
$wget https://d1cdn.apache.org/hadoop/common/hadoop-3.4.0/hadoop-3.4.0.tar.gz
```

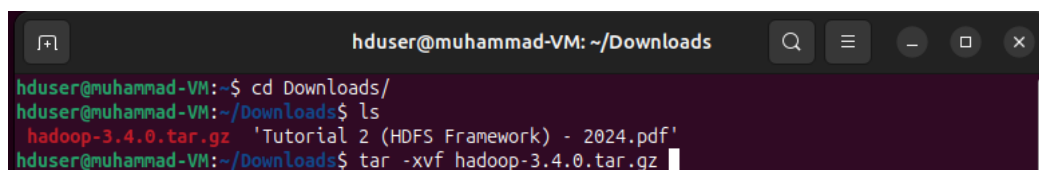


```
hduser@muhammad-VM: ~/Downloads
hduser@muhammad-VM:~$ cd Downloads
hduser@muhammad-VM:~/Downloads$ wget https://d1cdn.apache.org/hadoop/common/hadoop-3.4.0/hadoop-3.4.0.tar.gz
--2024-07-22 16:05:12-- https://d1cdn.apache.org/hadoop/common/hadoop-3.4.0/hadoop-3.4.0.tar.gz
Resolving d1cdn.apache.org (d1cdn.apache.org)... 151.101.2.132, 2a04:4e42::644
Connecting to d1cdn.apache.org (d1cdn.apache.org)|151.101.2.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 965537117 (921M) [application/x-gzip]
Saving to: 'hadoop-3.4.0.tar.gz'

hadoop-3.4.0.tar.gz 100%[=====] 920.81M 7.73MB/s in 1m 58s
2024-07-22 16:07:10 (7.83 MB/s) - 'hadoop-3.4.0.tar.gz' saved [965537117/965537117]
```

This command can be used on Google Cloud Platform or AWS cloud.

- 27) After completion of download, use **ls** command and check the downloaded package is available or not on your local drive in Ubuntu. Unzip this package using the command



```
hduser@muhammad-VM: ~/Downloads
hduser@muhammad-VM:~$ cd Downloads/
hduser@muhammad-VM:~/Downloads$ ls
hadoop-3.4.0.tar.gz 'Tutorial 2 (HDFS Framework) - 2024.pdf'
hduser@muhammad-VM:~/Downloads$ tar -xvf hadoop-3.4.0.tar.gz
```

```
$tar -xvf hadoop-3.4.0.tar.gz
```

[The Linux 'tar' stands for tape archive, is used to create Archive and extract the Archive files.]

```

hduser@muhammad-VM: ~/Downloads
hadoop-3.4.0/sbin/hadoop-daemons.sh
hadoop-3.4.0/sbin/refresh-namenodes.sh
hadoop-3.4.0/sbin/start-balancer.sh
hadoop-3.4.0/sbin/start-all.sh
hduser@muhammad-VM:~/Downloads$ ls
hadoop-3.4.0  hadoop-3.4.0.tar.gz  'Tutorial 2 (HDFS Framework) - 2024.pdf'
hduser@muhammad-VM:~/Downloads$ sudo mv ./hadoop-3.4.0 /usr/local
[sudo] password for hduser:
hduser@muhammad-VM:~/Downloads$ ls
hadoop-3.4.0.tar.gz  'Tutorial 2 (HDFS Framework) - 2024.pdf'
hduser@muhammad-VM:~/Downloads$

```

To move the unzipped Hadoop folder (Hadoop-3.4.0) to **usr** directory, use the command below

```
$sudo mv ./hadoop-3.4.0 /usr/local/
```

[**mv** command moves the directory from current to **local** directory, **./** means from the current directory]

```
$cd /usr/local
```

```

hduser@muhammad-VM: /usr/local
hduser@muhammad-VM:~/Downloads$ cd /usr/local
hduser@muhammad-VM: /usr/local$

```

- 28) Create a short link for the Hadoop folder (**Hadoop-3.4.0**) by using the command below

```
$sudo ln -sf /usr/local/hadoop-3.4.0/ /usr/local/hadoop
```

[The **ln** command is a standard Unix command utility used to create a hard link or a symbolic link (symlink) to an existing file. The use of a hard link allows multiple filenames to be associated with the same file since a hard link points to the inode (index node) of a given file, the data of which is stored on disk.]

```

hduser@muhammad-VM:~/Downloads$ cd /usr/local
hduser@muhammad-VM: /usr/local$ ls
bin  etc  games  hadoop-3.4.0  include  lib  man  sbin  share  src
hduser@muhammad-VM: /usr/local$ sudo ln -sf /usr/local/hadoop-3.4.0/ /usr/local/hadoop
hduser@muhammad-VM: /usr/local$ ls
bin  etc  games  hadoop  hadoop-3.4.0  include  lib  man  sbin  share  src
hduser@muhammad-VM: /usr/local$

```

- 29) Change the permission by using the command

```
$sudo chown -R hduser:hadoopgroup /usr/local/hadoop*
```

[The command **chown**, an abbreviation of change owner, is used on Unix-like systems to change the owner of file system files, directories. Unprivileged users who wish to change the group membership of a file that they own may use **chgrp**.]

```

hduser@muhammad-VM: ~/Downloads$ cd /usr/local
hduser@muhammad-VM: /usr/local$ ls
bin  etc  games  hadoop-3.4.0  include  lib  man  sbin  share  src
hduser@muhammad-VM: /usr/local$ sudo ln -sf /usr/local/hadoop-3.4.0/ /usr/local/hadoop
hduser@muhammad-VM: /usr/local$ ls
bin  etc  games  hadoop  hadoop-3.4.0  include  lib  man  sbin  share  src
hduser@muhammad-VM: /usr/local$ sudo chown -R hduser:hadoopgroup /usr/local/hadoop*
hduser@muhammad-VM: /usr/local$ ls -l
total 36
drwxr-xr-x  2 root  root    4096 Apr 24 11:47 bin
drwxr-xr-x  2 root  root    4096 Apr 24 11:47 etc
drwxr-xr-x  2 root  root    4096 Apr 24 11:47 games
lrwxrwxrwx  1 hduser  hadoopgroup  24 Jul 22 18:36 hadoop -> /usr/local/hadoop-3.4.0/
drwxr-xr-x 10 hduser  hadoopgroup 4096 Mar  4 08:05 hadoop-3.4.0
drwxr-xr-x  2 root  root    4096 Apr 24 11:47 include
drwxr-xr-x  3 root  root    4096 Apr 24 11:47 lib
lrwxrwxrwx  1 root  root      9 Apr 24 11:47 man -> share/man
drwxr-xr-x  2 root  root    4096 Apr 24 11:47 sbin
drwxr-xr-x  7 root  root    4096 Apr 24 11:49 share
drwxr-xr-x  2 root  root    4096 Apr 24 11:47 src
hduser@muhammad-VM: /usr/local$ cd
hduser@muhammad-VM: ~$ pwd
/home/hduser
hduser@muhammad-VM: ~$ nano ~/.bashrc

```

- Restart your system and login as **hduser** along with password.

30) Use the following command to update **bashrc** file

\$cd Hit the Enter Key to shift to /home/hduser

\$nano ~/.bashrc

[BASH is a Linux shell and BASH stands for Bourne Again Shell. The 'rc' suffix goes back to Unix's grandparent, CTSS. It had a command-script feature called "runcom". Early Unixes used 'rc' for the name of the operating system's boot script, as a tribute to CTSS runcom.]

- Move down to the end of file (./bashrc) and add the following lines of code

```

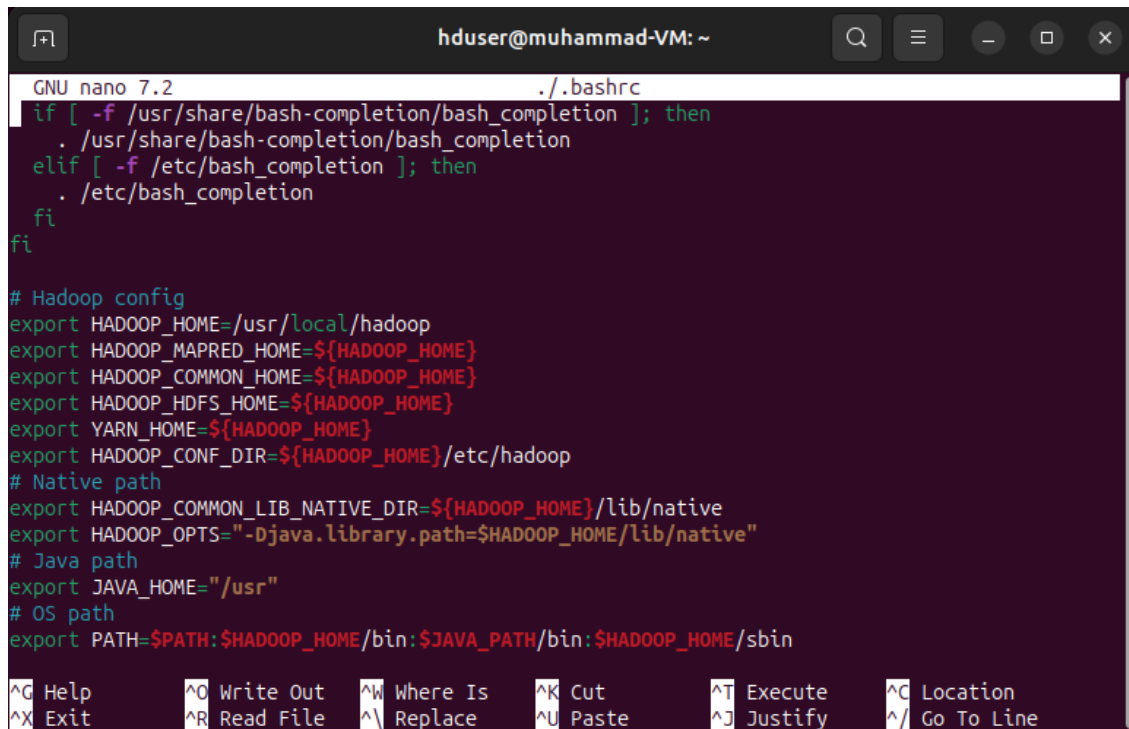
# Hadoop config
export HADOOP_HOME=/usr/local/hadoop
export HADOOP_MAPRED_HOME=${HADOOP_HOME}
export HADOOP_COMMON_HOME=${HADOOP_HOME}
export HADOOP_HDFS_HOME=${HADOOP_HOME}
export YARN_HOME=${HADOOP_HOME}
export HADOOP_CONF_DIR=${HADOOP_HOME}/etc/hadoop

# Native path
export HADOOP_COMMON_LIB_NATIVE_DIR=${HADOOP_HOME}/lib/native
export HADOOP_OPTS="-Djava.library.path=${HADOOP_HOME}/lib/native"

# Java path
export JAVA_HOME="/usr"

# OS path
export PATH=$PATH:${HADOOP_HOME}/bin:${JAVA_PATH}/bin:${HADOOP_HOME}/sbin

```

```

GNU nano 7.2                                ~/.bashrc
if [ -f /usr/share/bash-completion/bash_completion ]; then
. /usr/share/bash-completion/bash_completion
elif [ -f /etc/bash_completion ]; then
. /etc/bash_completion
fi
fi

# Hadoop config
export HADOOP_HOME=/usr/local/hadoop
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_CONF_DIR=${HADOOP_HOME}/etc/hadoop
# Native path
export HADOOP_COMMON_LIB_NATIVE_DIR=${HADOOP_HOME}/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
# Java path
export JAVA_HOME="/usr"
# OS path
export PATH=$PATH:$HADOOP_HOME/bin:$JAVA_HOME/bin:$HADOOP_HOME/sbin

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File  ^_ Replace    ^U Paste      ^J Justify    ^_ Go To Line

```

31) After writing above script, press **ctrl + x** and press **"y"** to save all these lines using the nano editor and **source** it again to be available to **hduser**.

```
$source ~/.bashrc
```

[Details for the configuration files can be obtained from the website:
<https://hadoop.apache.org/docs/stable/>]

32) Open the shell script file (**hadoop-env.sh**) to check that path for **JAVA** is set or not

```
$nano /usr/local/hadoop/etc/hadoop/hadoop-env.sh
```

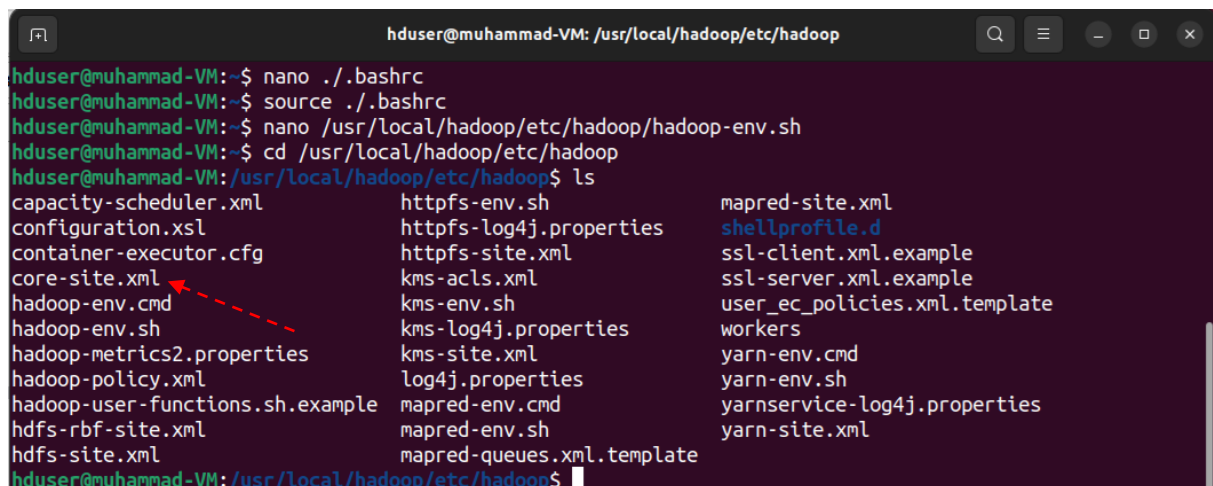
Append the following line at the end of this file and save as the method of **nano** editor.

```
export JAVA_HOME="/usr"
```

33) Configure the Hadoop, change the directory to

```
$cd /usr/local/hadoop/etc/hadoop
```

all files to set the configuration are present here



```

hduser@muhammad-VM: /usr/local/hadoop/etc/hadoop
hduser@muhammad-VM:~$ nano ~/.bashrc
hduser@muhammad-VM:~$ source ~/.bashrc
hduser@muhammad-VM:~$ nano /usr/local/hadoop/etc/hadoop/hadoop-env.sh
hduser@muhammad-VM:~$ cd /usr/local/hadoop/etc/hadoop
hduser@muhammad-VM:/usr/local/hadoop/etc/hadoop$ ls
capacity-scheduler.xml      https-env.sh              mapred-site.xml
configuration.xml           https-log4j.properties    shellprofile.d
container-executor.cfg      https-site.xml             ssl-client.xml.example
core-site.xml               kms-acls.xml               ssl-server.xml.example
hadoop-env.cmd              kms-env.sh                 user_ec_policies.xml.template
hadoop-env.sh               kms-log4j.properties      workers
hadoop-metrics2.properties kms-site.xml               yarn-env.cmd
hadoop-policy.xml           log4j.properties          yarn-env.sh
hadoop-user-functions.sh.example mapred-env.cmd             yarnservice-log4j.properties
hdfs-rbf-site.xml           mapred-env.sh              yarn-site.xml
hdfs-site.xml               mapred-queues.xml.template
hduser@muhammad-VM:/usr/local/hadoop/etc/hadoop$

```


34) Open a file at the following path as mentioned with red arrow.

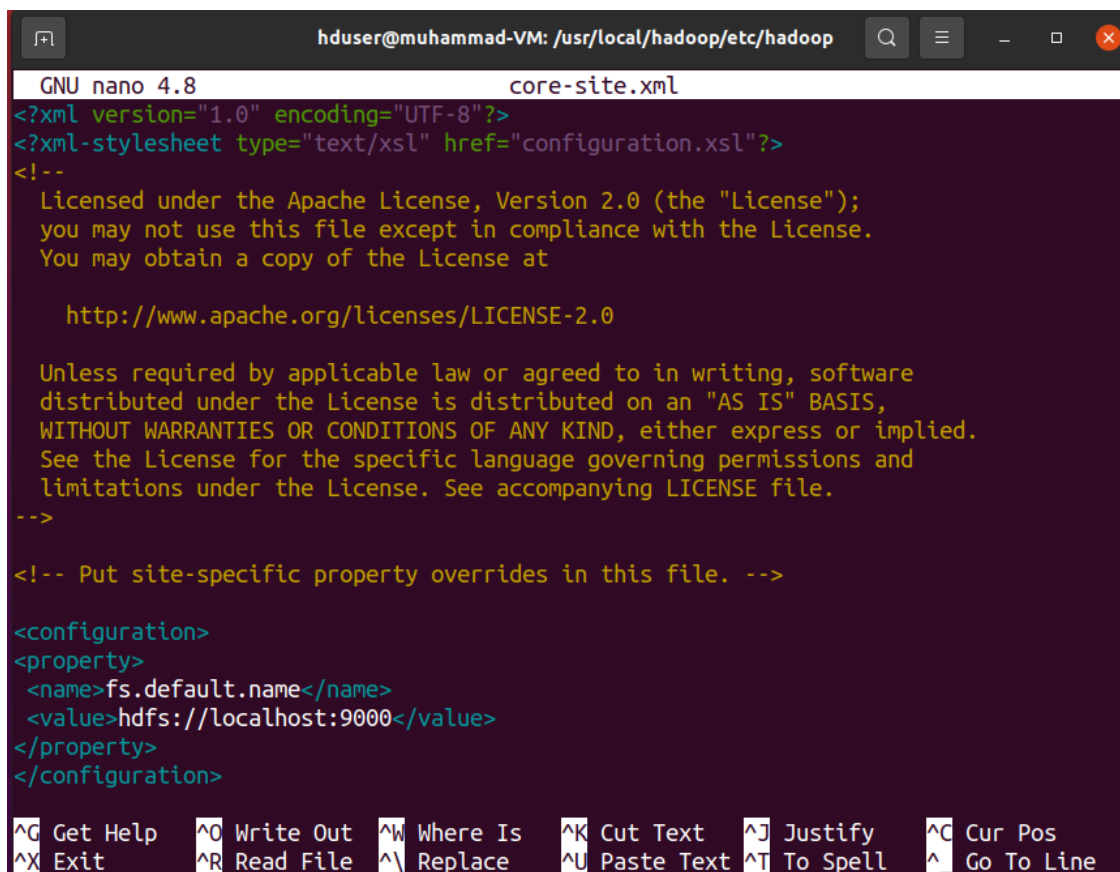
\$nano core-site.xml and a file will be opened and write the following lines at the end of the file as shown below

```
<configuration>
</configuration>
```

The core-site.xml contain green coloured **xml** opening and closing tags (<configuration>). Please insert the **property, name and value tags** along with the specified values and your Hadoop core-site.xml file should look like as mentioned below

```
<configuration>
<property>
  <name>fs.default.name</name>
  <value>hdfs://localhost:9000</value>
</property>
</configuration>
```

After all updates, the core-site.xml will look like as mentioned below in the screenshot.



```
GNU nano 4.8 core-site.xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>
<property>
  <name>fs.default.name</name>
  <value>hdfs://localhost:9000</value>
</property>
</configuration>
```

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Paste Text ^T To Spell ^_ Go To Line

After writing the above script, press **ctrl + x** and write **y** to save the above modifications in the file.

- Similarly, Update the file (**hdfs-site.xml**) as mentioned below

```
$nano hdfs-site.xml
```

```
<configuration>
<property>
  <name>dfs.replication</name>
  <value>1</value>
</property>

<property>
  <name>dfs.name.dir</name>
  <value>file:/usr/local/hadoop/hadoopdata/hdfs/namenode</value>
</property>

<property>
  <name>dfs.data.dir</name>
  <value>file:/usr/local/hadoop/hadoopdata/hdfs/datanode</value>
</property>
</configuration>
```

After writing the above script, Press **ctrl + x** and write **y** to save the above modifications in the file.

35) **mapred-site.xml.template** is present in the **/usr/local/hadoop/etc/hadoop/** folder and first make a copy of a file as (**mapred-site.xml**) by using the following command for safe purpose.

```
$cp mapred-site.xml mapred-site.xml.template
```

```
hduser@muhammad-VM:/usr/local/hadoop/etc/hadoop$ cp mapred-site.xml mapred-site.xml.template
```

- open a file and add the highlighted text into **mapred-site.xml** as mentioned below

```
$nano mapred-site.xml
```

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

After writing the above script, press **ctrl + x** and write **y** to save the above modifications in the file.

- open another file in the same folder (**/usr/local/hadoop/etc/hadoop/**) as

```
$nano yarn-site.xml
```

and add the following code in the Configuration section.

```
<configuration>
<property>
  <name>yarn.nodemanager.aux-services</name>
  <value>mapreduce_shuffle</value>
</property>
</configuration>
```

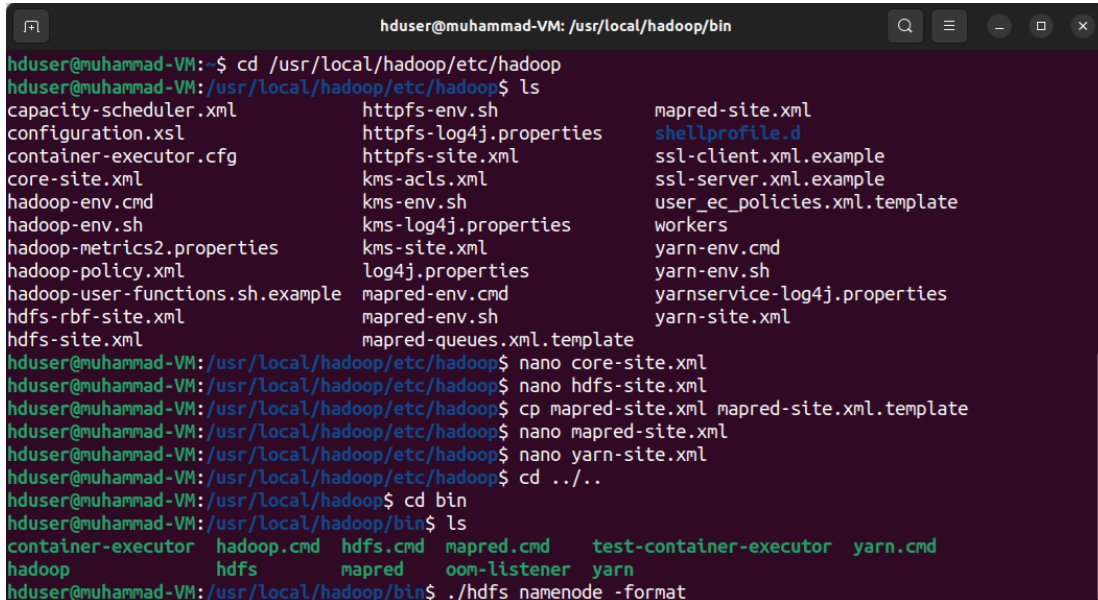
After writing the above script, press **ctrl + x** and write **y** to save the above modifications in the file.

36) Move to the main hadoop directory by using the command and it is shown in the screenshot in step 37

```
$cd ../../
$cd bin
```

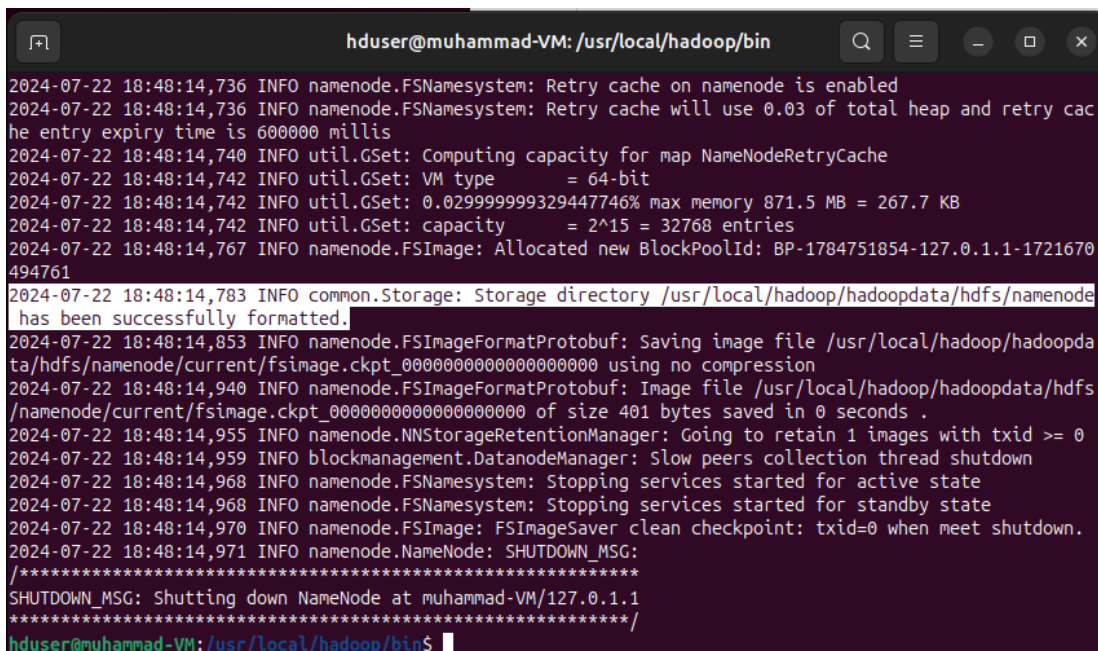
37) Execute the command for the formatting of Hadoop distributed file system (**hdfs**)

```
$./hdfs namenode -format
```



```
hduser@muhammad-VM: /usr/local/hadoop/bin
hduser@muhammad-VM:~$ cd /usr/local/hadoop/etc/hadoop
hduser@muhammad-VM:/usr/local/hadoop/etc/hadoop$ ls
capacity-scheduler.xml      httpfs-env.sh              mapred-site.xml
configuration.xsl           httpfs-log4j.properties   shellprofile.d
container-executor.cfg      httpfs-site.xml           ssl-client.xml.example
core-site.xml               kms-acls.xml              ssl-server.xml.example
hadoop-env.cmd              kms-env.sh                user_ec_policies.xml.template
hadoop-env.sh               kms-log4j.properties      workers
hadoop-metrics2.properties  kms-site.xml              yarn-env.cmd
hadoop-policy.xml           log4j.properties          yarn-env.sh
hadoop-user-functions.sh.example  mapred-env.cmd           yarnservice-log4j.properties
hdfs-rbf-site.xml           mapred-env.sh             yarn-site.xml
hdfs-site.xml               mapred-queues.xml.template
hduser@muhammad-VM:/usr/local/hadoop/etc/hadoop$ nano core-site.xml
hduser@muhammad-VM:/usr/local/hadoop/etc/hadoop$ nano hdfs-site.xml
hduser@muhammad-VM:/usr/local/hadoop/etc/hadoop$ cp mapred-site.xml mapred-site.xml.template
hduser@muhammad-VM:/usr/local/hadoop/etc/hadoop$ nano mapred-site.xml
hduser@muhammad-VM:/usr/local/hadoop/etc/hadoop$ nano yarn-site.xml
hduser@muhammad-VM:/usr/local/hadoop/etc/hadoop$ cd ../../
hduser@muhammad-VM:/usr/local/hadoop$ cd bin
hduser@muhammad-VM:/usr/local/hadoop/bin$ ls
container-executor  hadoop.cmd  hdfs.cmd  mapred.cmd  test-container-executor  yarn.cmd
hadoop              hdfs        mapred    oom-listener  yarn
hduser@muhammad-VM:/usr/local/hadoop/bin$ ./hdfs namenode -format
```

38) You will see a message as mentioned below on screen after successful completion of the formatting command, and you can find the highlighted lines on your screen as shown below



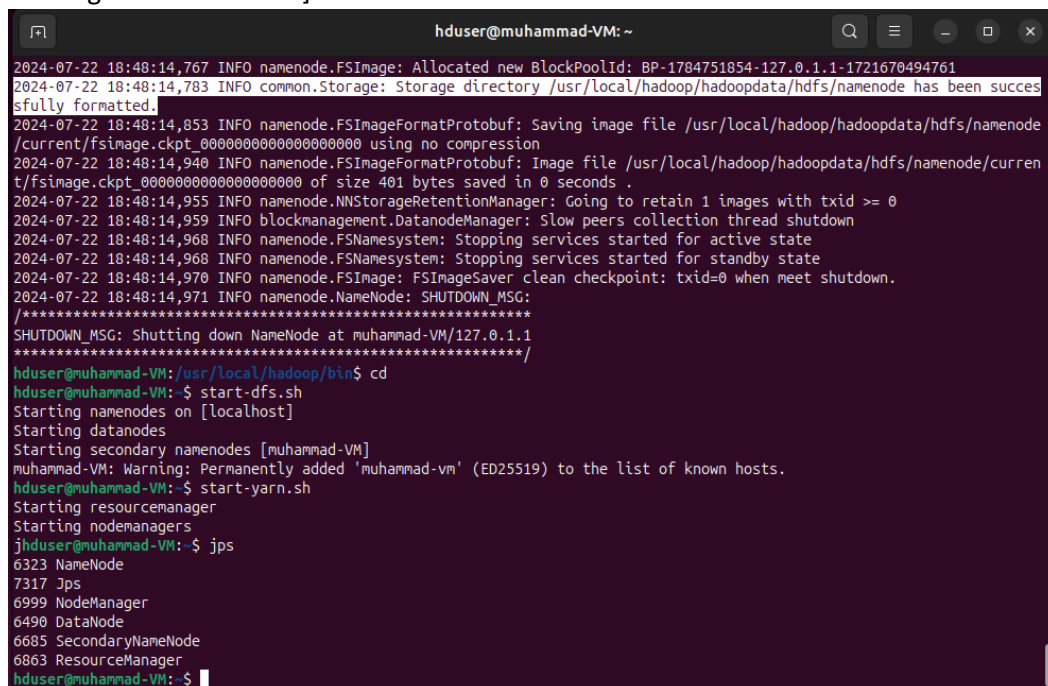
```
hduser@muhammad-VM: /usr/local/hadoop/bin
2024-07-22 18:48:14,736 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
2024-07-22 18:48:14,736 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total heap and retry cache entry expiry time is 600000 millis
2024-07-22 18:48:14,740 INFO util.GSet: Computing capacity for map NameNodeRetryCache
2024-07-22 18:48:14,742 INFO util.GSet: VM type = 64-bit
2024-07-22 18:48:14,742 INFO util.GSet: 0.0299999999329447746% max memory 871.5 MB = 267.7 KB
2024-07-22 18:48:14,742 INFO util.GSet: capacity = 2^15 = 32768 entries
2024-07-22 18:48:14,767 INFO namenode.FSImage: Allocated new BlockPoolId: BP-1784751854-127.0.1.1-1721670494761
2024-07-22 18:48:14,783 INFO common.Storage: Storage directory /usr/local/hadoop/hadoopdata/hdfs/namenode has been successfully formatted.
2024-07-22 18:48:14,853 INFO namenode.FSImageFormatProtobuf: Saving image file /usr/local/hadoop/hadoopdata/hdfs/namenode/current/fsimage.ckpt_00000000000000000000 using no compression
2024-07-22 18:48:14,940 INFO namenode.FSImageFormatProtobuf: Image file /usr/local/hadoop/hadoopdata/hdfs/namenode/current/fsimage.ckpt_00000000000000000000 of size 401 bytes saved in 0 seconds .
2024-07-22 18:48:14,955 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
2024-07-22 18:48:14,959 INFO blockmanagement.DatanodeManager: Slow peers collection thread shutdown
2024-07-22 18:48:14,968 INFO namenode.FSNamesystem: Stopping services started for active state
2024-07-22 18:48:14,968 INFO namenode.FSNamesystem: Stopping services started for standby state
2024-07-22 18:48:14,970 INFO namenode.FSImage: FSImageSaver clean checkpoint: txid=0 when meet shutdown.
2024-07-22 18:48:14,971 INFO namenode.NameNode: SHUTDOWN_MSG:
/*****
SHUTDOWN_MSG: Shutting down NameNode at muhammad-VM/127.0.1.1
*****/
hduser@muhammad-VM:/usr/local/hadoop/bin$
```

Now **namenode** is ready for the Hadoop platform. Move to the original directory by using the command

```
$cd and press Enter key
```

```
hduser@muhammad-VM: /usr/local/hadoop/bin$ cd
hduser@muhammad-VM: ~$ pwd
/home/hduser
hduser@muhammad-VM: ~$
```

- 39) **\$start-dfs.sh** and it takes a little while. In case of **yes/no** option asked at the terminal, write **yes** on the terminal
- 40) **\$start-yarn.sh** and press Enter key
[Used to start and stop hadoop daemons all at once. Issuing it on the master machine will start/stop the daemons on all the nodes of a cluster.]
- 41) After completion of this process, write the command and also mentioned below in the screen shot
\$jps
[**jps** (Java Virtual Machine Process Status Tool) is a command which is used to check all the Hadoop daemons like NameNode, DataNode, ResourceManager, NodeManager etc. that are running on the machine.]



```
hduser@muhammad-VM: ~
2024-07-22 18:48:14,767 INFO namenode.FSImage: Allocated new BlockPoolId: BP-1784751854-127.0.1.1-1721670494761
2024-07-22 18:48:14,783 INFO common.Storage: Storage directory /usr/local/hadoop/hadoopdata/hdfs/namenode has been successfully formatted.
2024-07-22 18:48:14,853 INFO namenode.FSImageFormatProtobuf: Saving image file /usr/local/hadoop/hadoopdata/hdfs/namenode/current/fsimage.ckpt_000000000000000000 using no compression
2024-07-22 18:48:14,940 INFO namenode.FSImageFormatProtobuf: Image file /usr/local/hadoop/hadoopdata/hdfs/namenode/current/fsimage.ckpt_000000000000000000 of size 481 bytes saved in 0 seconds.
2024-07-22 18:48:14,955 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
2024-07-22 18:48:14,959 INFO blockmanagement.DatanodeManager: Slow peers collection thread shutdown
2024-07-22 18:48:14,968 INFO namenode.FSNamesystem: Stopping services started for active state
2024-07-22 18:48:14,968 INFO namenode.FSNamesystem: Stopping services started for standby state
2024-07-22 18:48:14,970 INFO namenode.FSImage: FSImageSaver clean checkpoint: txid=0 when meet shutdown.
2024-07-22 18:48:14,971 INFO namenode.NameNode: SHUTDOWN_MSG:
/*****
SHUTDOWN_MSG: Shutting down NameNode at muhammad-VM/127.0.1.1
*****/
hduser@muhammad-VM: /usr/local/hadoop/bin$ cd
hduser@muhammad-VM: ~$ start-dfs.sh
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [muhammad-VM]
muhammad-VM: Warning: Permanently added 'muhammad-vm' (ED25519) to the list of known hosts.
hduser@muhammad-VM: ~$ start-yarn.sh
Starting resourcemanager
Starting nodemanagers
hduser@muhammad-VM: ~$ jps
6323 NameNode
7317 Jps
6999 NodeManager
6490 DataNode
6685 SecondaryNameNode
6863 ResourceManager
hduser@muhammad-VM: ~$
```

- 42) If the above six processes are shown on your system, It means that the hadoop is working perfectly and id's of these processes are different on each VMs.
- 43) Use the command to check the root directory of hadoop distributed file system (hdfs)
\$hadoop fs -ls /
you do not see any output because the Hadoop directory is empty.
- 44) Move a file named "Hadoop-3.4.0.tar.gz" to hadoop by using
\$cd
Hit the Enter Key and move to the **/home/hduser/** and verify this using **pwd** command
\$pwd
\$cd Downloads
\$hadoop fs -put ./hadoop-3.4.0.tar.gz /
- 45) To check that the file is moved on the hadoop or not, again use the same command (step 43) as mentioned below
\$hadoop fs -ls /
- 46) If you would like to remove the file from the hadoop, use the command below
\$hadoop fs -rm /hadoop-3.4.0.tar.gz

and you can check again

\$hadoop fs -ls /

```
hduser@muhammad-VM:~$ jps
6323 NameNode
7317 Jps
6999 NodeManager
6490 DataNode
6685 SecondaryNameNode
6863 ResourceManager
hduser@muhammad-VM:~$ cd Downloads
hduser@muhammad-VM:~/Downloads$ hadoop fs -ls /
hduser@muhammad-VM:~/Downloads$ ls
hadoop-3.4.0.tar.gz 'Tutorial 2 (HDFS Framework) - 2024.pdf'
hduser@muhammad-VM:~/Downloads$ hadoop fs -put ./hadoop-3.4.0.tar.gz /
hduser@muhammad-VM:~/Downloads$ hadoop fs -ls /
Found 1 items
-rw-r--r-- 1 hduser supergroup 965537117 2024-07-22 18:55 /hadoop-3.4.0.tar.gz
hduser@muhammad-VM:~/Downloads$ hadoop fs -rm /hadoop-3.4.0.tar.gz
Deleted /hadoop-3.4.0.tar.gz
hduser@muhammad-VM:~/Downloads$ hadoop fs -ls /
hduser@muhammad-VM:~/Downloads$
```

- 47) Whenever you will start the hadoop, the following two commands must be used as mentioned below

\$start-dfs.sh

\$start-yarn.sh

All hadoop process can be checked by using **jps** command. After completion of your work, you must stop the hadoop processes before shutting down VM. The default port for hadoop access using Google/Edge/Mozilla Firefox browser is 9870, for example: localhost:9870. You can get detailed understanding of distributed hadoop clusters (NameNodes and DataNodes).

- 48) To stop the services of hadoop, use the following commands one by one as

\$stop-dfs.sh

\$stop-yarn.sh

If you would like to explore further, the following website might be useful as mentioned below

References:

- <https://hadoop.apache.org/docs/stable/>
- <https://ricma.co/install-apache-hadoop-27-on-buntu-1604.html>
- <https://www.youtube.com/watch?v=Y6oit3rCsZo>