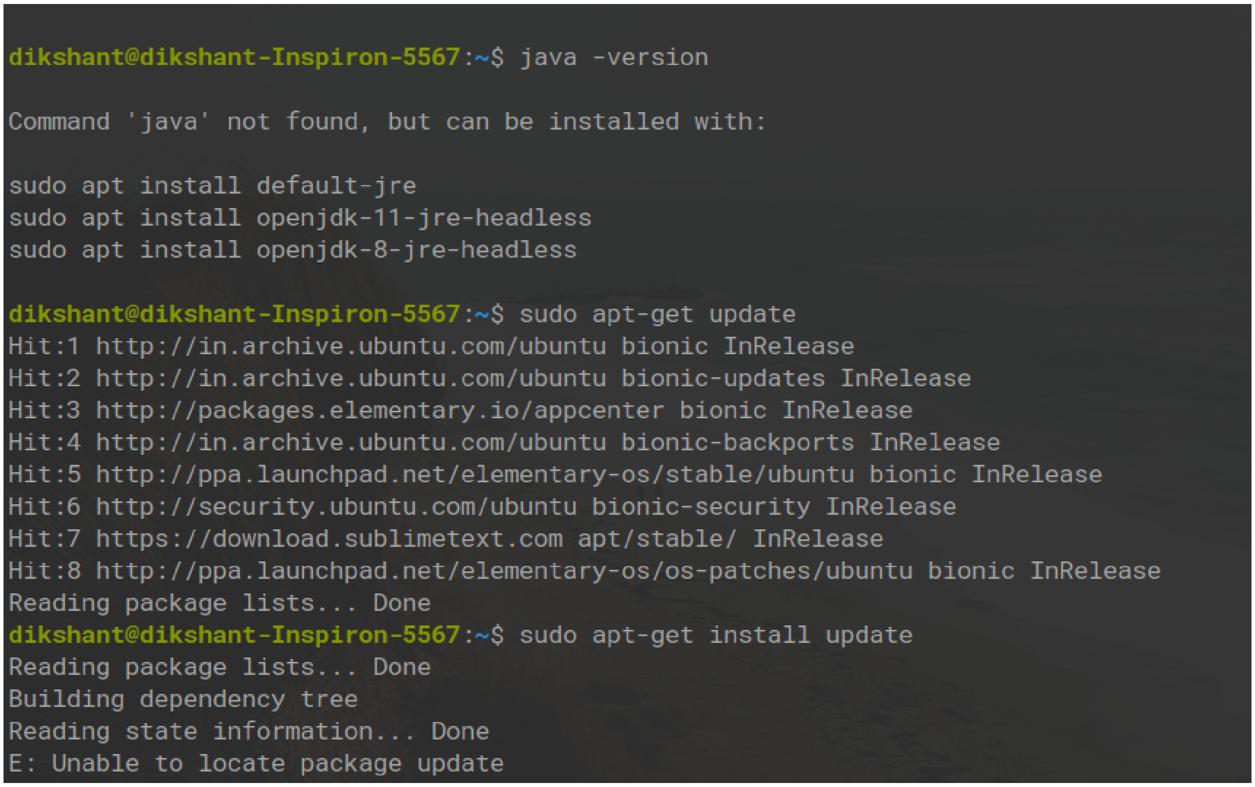
How to Install Hadoop in Linux?

Step 1: Open your terminal and first check whether your system is equipped with Java or not with command

```
java -version
```

Step 2: Update your system using the below 2 commands.

```
sudo apt-get update

sudo apt-get install update
```

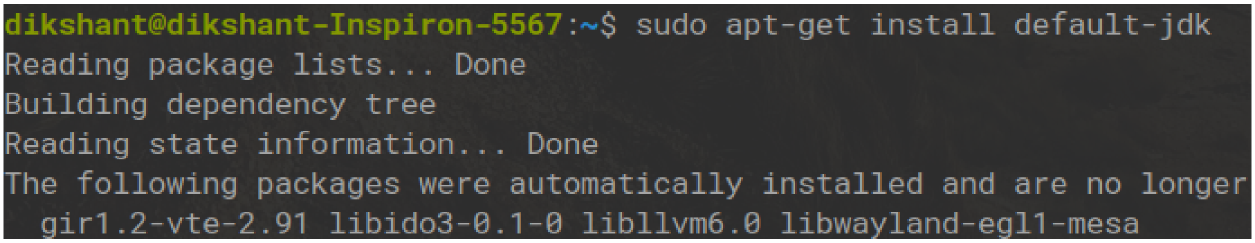


```
dikshant@dikshant-Inspiron-5567:~$ java -version

Command 'java' not found, but can be installed with:

sudo apt install default-jre
sudo apt install openjdk-11-jre-headless
sudo apt install openjdk-8-jre-headless

dikshant@dikshant-Inspiron-5567:~$ sudo apt-get update
Hit:1 http://in.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://in.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://packages.elementary.io/appcenter bionic InRelease
Hit:4 http://in.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:5 http://ppa.launchpad.net/elementary-os/stable/ubuntu bionic InRelease
Hit:6 http://security.ubuntu.com/ubuntu bionic-security InRelease
Hit:7 https://download.sublimetext.com apt/stable/ InRelease
Hit:8 http://ppa.launchpad.net/elementary-os/os-patches/ubuntu bionic InRelease
Reading package lists... Done
dikshant@dikshant-Inspiron-5567:~$ sudo apt-get install update
Reading package lists... Done
Building dependency tree
Reading state information... Done
E: Unable to locate package update
```

Step 3: Now we will install the default JDK for java using the following command:

```
sudo apt-get install default-jdk
```

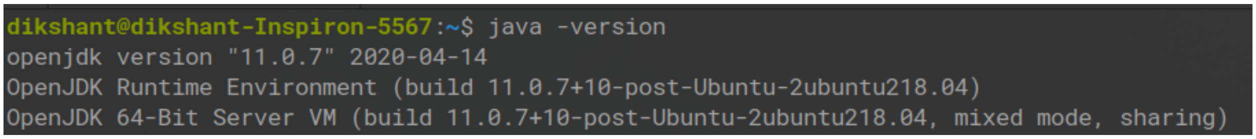
It will ask you for Y/N press Y.



```
dikshant@dikshant-Inspiron-5567:~$ sudo apt-get install default-jdk
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer
  gir1.2-vte-2.91 libido3-0.1-0 libllvm6.0 libwayland-egl1-mesa
```

Step 4: Now check whether Java is installed or not using the command

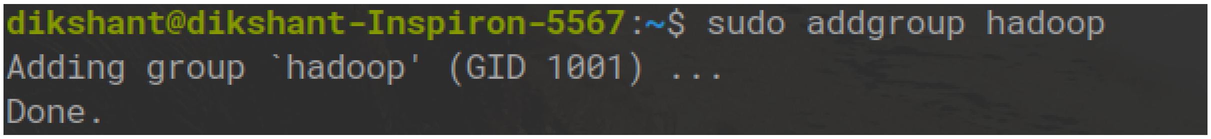
```
java -version
```



```
dikshant@dikshant-Inspiron-5567:~$ java -version
openjdk version "11.0.7" 2020-04-14
OpenJDK Runtime Environment (build 11.0.7+10-post-Ubuntu-2ubuntu218.04)
OpenJDK 64-Bit Server VM (build 11.0.7+10-post-Ubuntu-2ubuntu218.04, mixed mode, sharing)
```

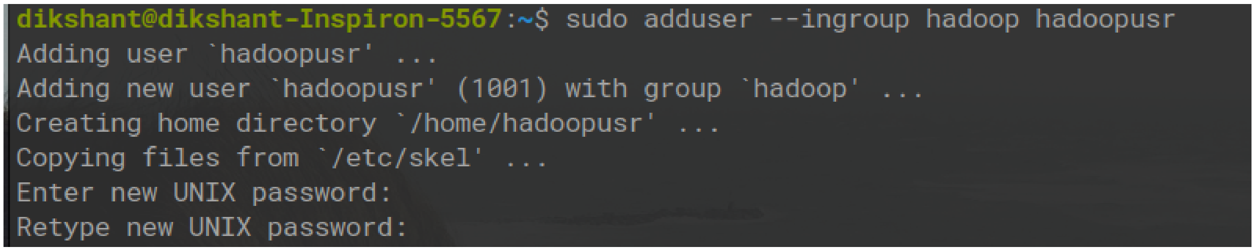
Step 5: Once it installs we require a dedicated user for the same. It is not necessary but it is a good thing to make a dedicated user for the **Hadoop** installation. You can use the following command:

```
sudo addgroup hadoop
```



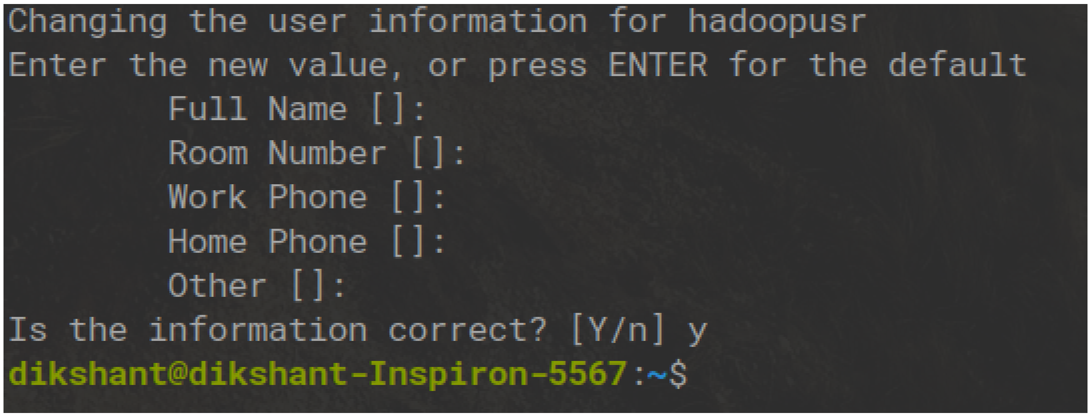
```
dikshant@dikshant-Inspiron-5567:~$ sudo addgroup hadoop
Adding group `hadoop' (GID 1001) ...
Done.
```

```
sudo adduser --ingroup hadoop hadoopusr
```



```
dikshant@dikshant-Inspiron-5567:~$ sudo adduser --ingroup hadoop hadoopusr
Adding user `hadoopusr' ...
Adding new user `hadoopusr' (1001) with group `hadoop' ...
Creating home directory `/home/hadoopusr' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
```

Step 6: Now after running the above 2 commands, you have successfully created a dedicated user with name **hadoopusr**. Now it will ask for a new UNIX password so choose password according to your convenience(make sure sometimes it doesn’t show the character or number you type so please remember whatever you type). Then it will ask you for information like Full Name etc. Keep pressing enter for default then press Y for correct information.

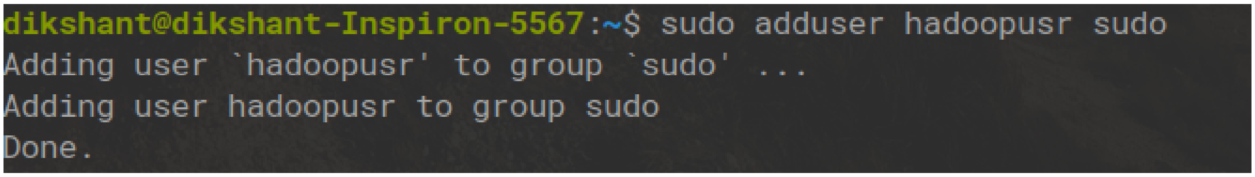


```
Changing the user information for hadoopusr
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
dikshant@dikshant-Inspiron-5567:~$
```

Step 7: Now use the following command:

```
sudo adduser hadoopusr sudo
```

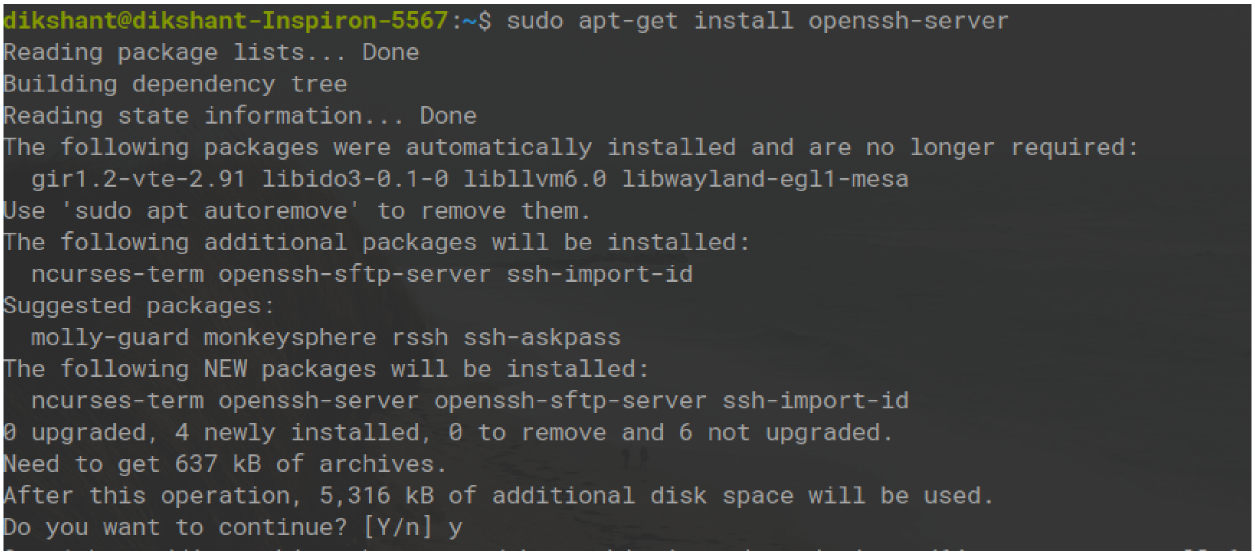
With this command, you add your ‘hadoopusr’ to the ‘sudo’ group so that we can also make it a superuser.



```
dikshant@dikshant-Inspiron-5567:~$ sudo adduser hadoopusr sudo
Adding user `hadoopusr' to group `sudo' ...
Adding user hadoopusr to group sudo
Done.
```

Step 8: Now we also need to install ssh key's that is secured shell.

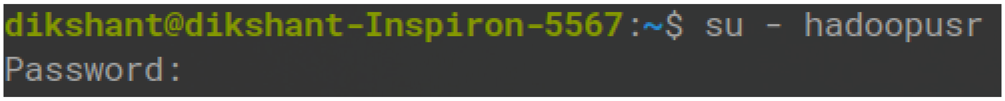
```
sudo apt-get install openssh-server
```



```
dikshant@dikshant-Inspiron-5567:~$ sudo apt-get install openssh-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  gir1.2-vte-2.91 libido3-0.1-0 libllvm6.0 libwayland-egl1-mesa
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard monkeysphere rssh ssh-askpass
The following NEW packages will be installed:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
0 upgraded, 4 newly installed, 0 to remove and 6 not upgraded.
Need to get 637 kB of archives.
After this operation, 5,316 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Step 9: Now it's time for us to switch to new user that is hadoopusr and also enter the password you use above command for switching user:

```
su - hadoopusr
```

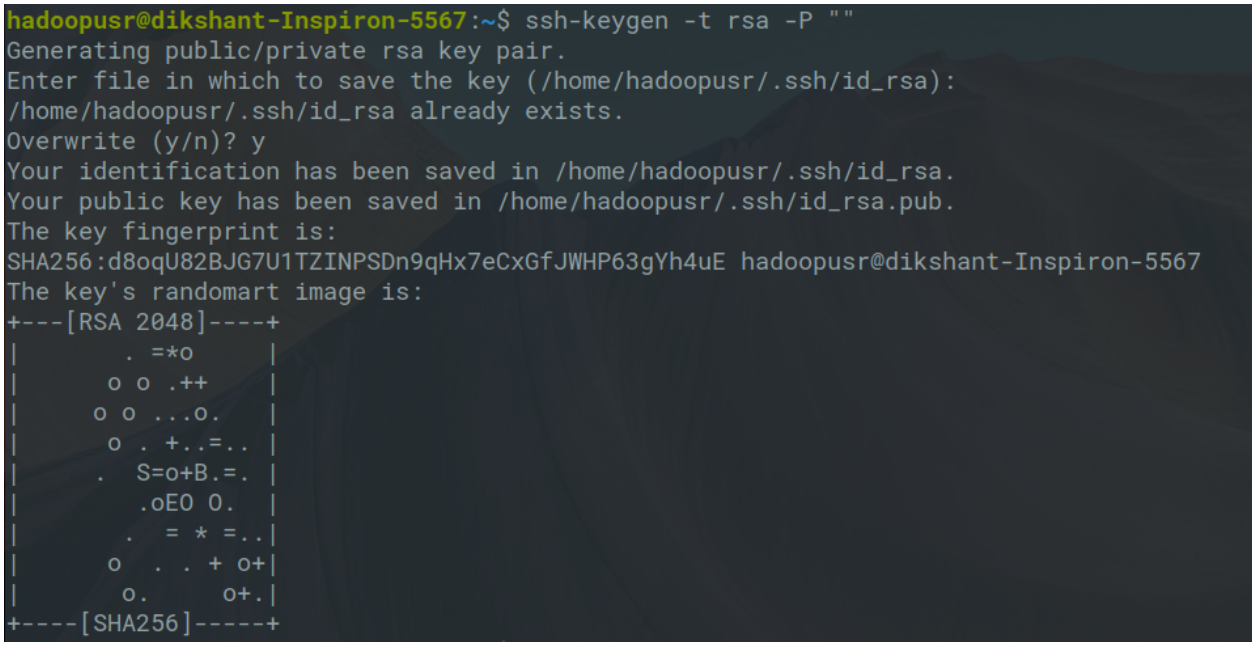


```
dikshant@dikshant-Inspiron-5567:~$ su - hadoopusr
Password:
```

Step 10: Now it's time to generate ssh key because Hadoop requires ssh access to manage it's node, remote or local machine so for our single node of the setup of Hadoop we configure such that we have access to the localhost.

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

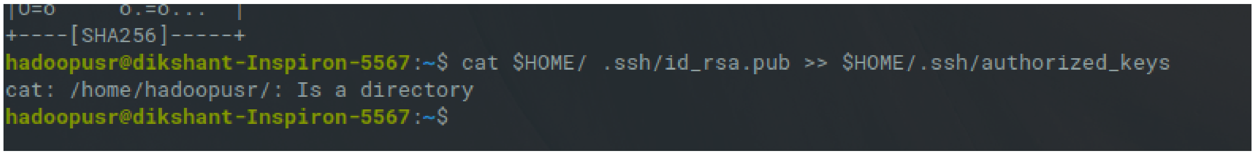
After this command simple press **enter**.



```
hadoopusr@dikshant-Inspiron-5567:~$ ssh-keygen -t rsa -P ""
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hadoopusr/.ssh/id_rsa):
/home/hadoopusr/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Your identification has been saved in /home/hadoopusr/.ssh/id_rsa.
Your public key has been saved in /home/hadoopusr/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:d8oqU82BJG7U1TZINPSDn9qHx7eCxGfJWHP63gYh4uE hadoopusr@dikshant-Inspiron-5567
The key's randomart image is:
+---[RSA 2048]----+
|       . =*o     |
|      o o .++    |
|     o o ...o.   |
|      o . +..=.. |
|     .  S=o+B.=. |
|        .oEO O.  |
|       .  = * =..|
|      o   . . + o+|
|       o.     o+.|
+----[SHA256]-----+
```

Step 11: Now we use the below command because we need to add the public key of the computer to the authorized key file of the compute that you want to access with ssh keys so we fired these command.

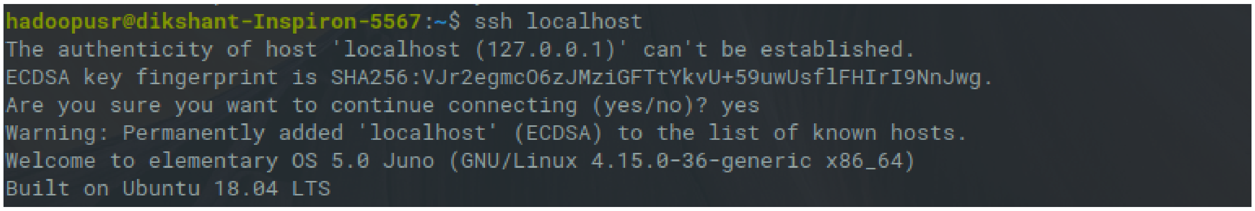
```
cat $HOME/ .ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys
```



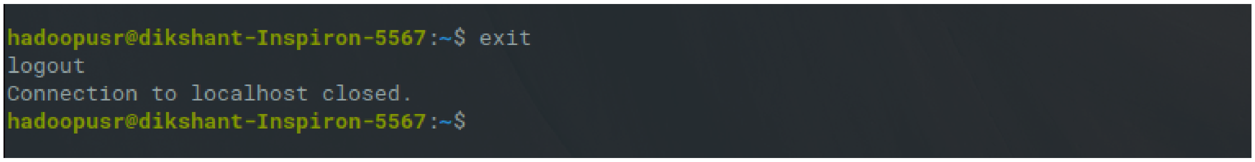
```
|O=o     o.=o...  |
+----[SHA256]-----+
hadoopusr@dikshant-Inspiron-5567:~$ cat $HOME/ .ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys
cat: /home/hadoopusr/: Is a directory
hadoopusr@dikshant-Inspiron-5567:~$
```

Step 12: Now check for the local host i.e. ssh localhost with below command and press *yes* to continue and enter your **password** if it ask then type exit.

```
ssh localhost
```



```
hadoopusr@dikshant-Inspiron-5567:~$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:VJr2egmcO6zJMziGFTtYkvU+59uwUsflFHIrI9NnJwg.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
Welcome to elementary OS 5.0 Juno (GNU/Linux 4.15.0-36-generic x86_64)
Built on Ubuntu 18.04 LTS
```



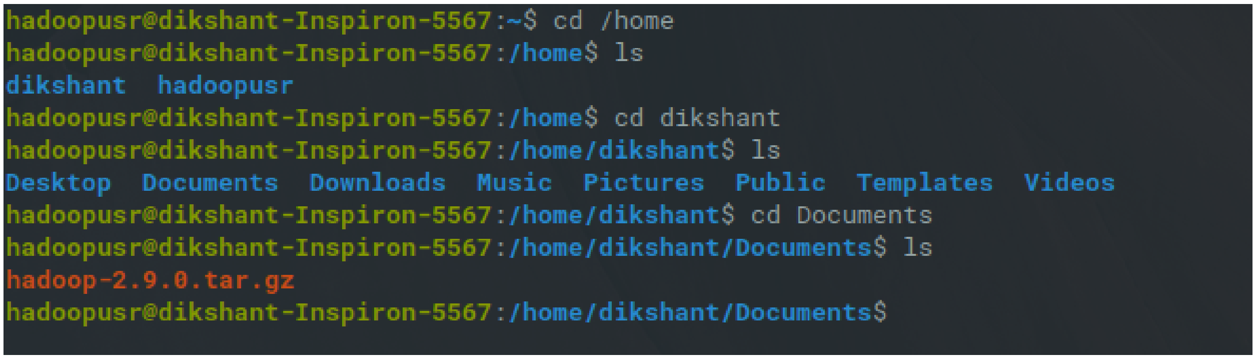
```
hadoopusr@dikshant-Inspiron-5567:~$ exit
logout
Connection to localhost closed.
hadoopusr@dikshant-Inspiron-5567:~$
```

Now you have completed the basic requirement for **Hadoop** installation.

Step 13: Now download the package that you will going to install . download it from Hadoop-2.9.0 by clicking to the file shown in below image.

```
hadoop-2.9.0.tar.gz            2017-11-17 23:10   350M
```

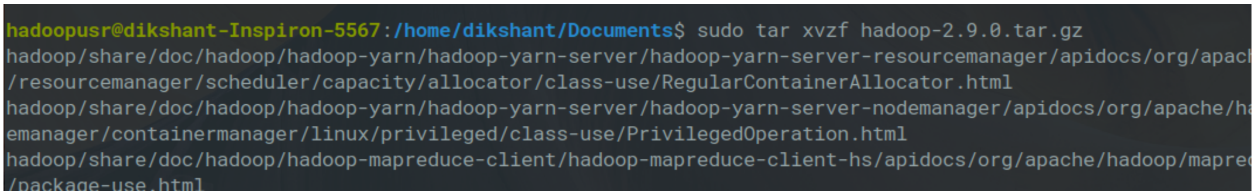
Step 14: Once you have download *hadoop-2.9.0.tar.gz* then place this tar file to your preferred location then extract it with below commands. In my case I moved it to the */Documents* folder.



```
hadoopusr@dikshant-Inspiron-5567:~$ cd /home
hadoopusr@dikshant-Inspiron-5567:/home$ ls
dikshant  hadoopusr
hadoopusr@dikshant-Inspiron-5567:/home$ cd dikshant
hadoopusr@dikshant-Inspiron-5567:/home/dikshant$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
hadoopusr@dikshant-Inspiron-5567:/home/dikshant$ cd Documents
hadoopusr@dikshant-Inspiron-5567:/home/dikshant/Documents$ ls
hadoop-2.9.0.tar.gz
hadoopusr@dikshant-Inspiron-5567:/home/dikshant/Documents$
```

Now we extract this file with below command and enter your *hadoopusr* password. If you don’t know the password don’t worry you can simply switch your user and change password according to yourself.

```
      sudo tar xvzf hadoop-2.9.0.tar.gz
```



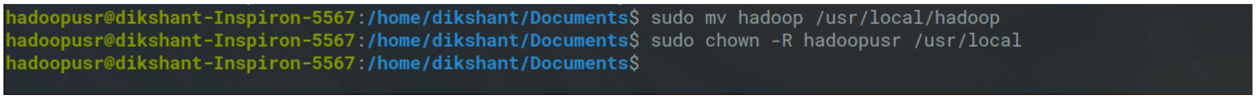
```
hadoopusr@dikshant-Inspiron-5567:/home/dikshant/Documents$ sudo tar xvzf hadoop-2.9.0.tar.gz
hadoop/share/doc/hadoop/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/apidocs/org/apach
/resourcemanager/scheduler/capacity/allocator/class-use/RegularContainerAllocator.html
hadoop/share/doc/hadoop/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-nodemanager/apidocs/org/apache/ha
emanager/containermanager/linux/privileged/class-use/PrivilegedOperation.html
hadoop/share/doc/hadoop/hadoop-mapreduce-client/hadoop-mapreduce-client-hs/apidocs/org/apache/hadoop/mapred
/package-use.html
```

Step 15: Now we need to move this extracted folder to the *hadoopusr* user so for that type below command(make sure name of your extracted folder is *hadoop*):

```
sudo mv hadoop /usr/local/hadoop
```

Step 16: Now we need to change the ownership so for that command is:

```
sudo chown -R hadoopusr /usr/local
```



```
hadoopusr@dikshant-Inspiron-5567:/home/dikshant/Documents$ sudo mv hadoop /usr/local/hadoop
hadoopusr@dikshant-Inspiron-5567:/home/dikshant/Documents$ sudo chown -R hadoopusr /usr/local
hadoopusr@dikshant-Inspiron-5567:/home/dikshant/Documents$
```

Note: For the next couple of steps, we will be using a text editor to set up configurations. The pictures show gedit but since I was having some issues with gedit, I used vim instead. You can use whatever you're comfortable with.

To make changes in vim:

Press 'i' to enter editing mode.

Copy and paste whatever needs to be added.

Press Esc to exit editing mode.

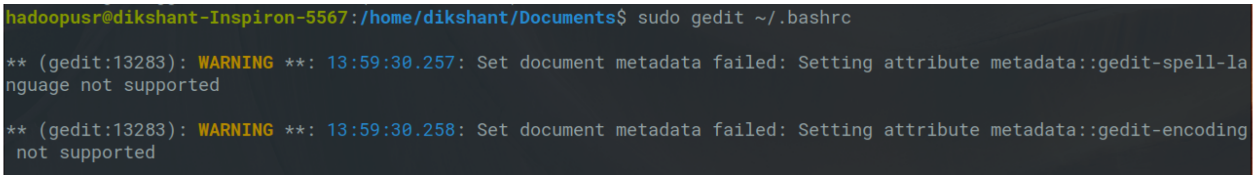
Press ':' +'w' to save the file.

Press ':' + 'q' to exit.

Step 17: This is the most important Step i.e. now we are going to configure some files.

First we configure our **./bashrc** file so that to open that file type the below command:

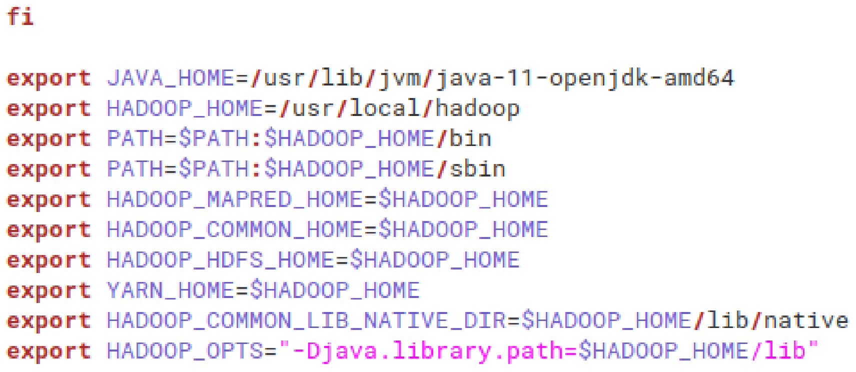
```
sudo vim ~/.bashrc
```



```
hadoopusr@dikshant-Inspiron-5567:/home/dikshant/Documents$ sudo gedit ~/.bashrc

** (gedit:13283): WARNING **: 13:59:30.257: Set document metadata failed: Setting attribute metadata::gedit-spell-la
nguage not supported

** (gedit:13283): WARNING **: 13:59:30.258: Set document metadata failed: Setting attribute metadata::gedit-encoding
 not supported
```

Then a **./bashrc file** is open then copy the below command inside this file (change java version according to your PC java version like it might be java-8-openjdk-amd64).

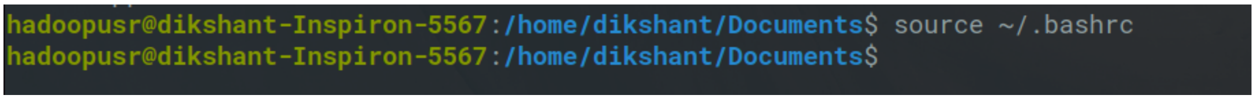
```
export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
export HADOOP_HOME=/usr/local/hadoop
export PATH=$PATH:$HADOOP_HOME/bin
export PATH=$PATH:$HADOOP_HOME/sbin
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib"
```



```
fi

export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
export HADOOP_HOME=/usr/local/hadoop
export PATH=$PATH:$HADOOP_HOME/bin
export PATH=$PATH:$HADOOP_HOME/sbin
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=$HADOOP_HOME/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib"
```

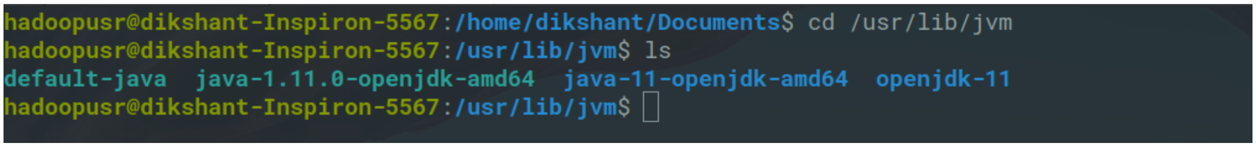
Then check whether you have configured it correctly or not.

```
    source ~/.bashrc
```



```
hadoopusr@dikshant-Inspiron-5567:/home/dikshant/Documents$ source ~/.bashrc
hadoopusr@dikshant-Inspiron-5567:/home/dikshant/Documents$
```

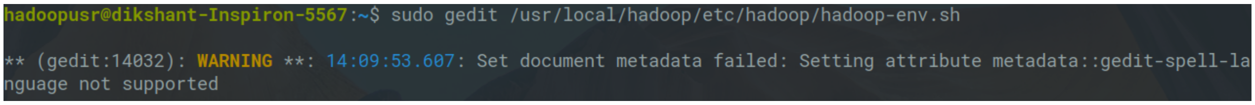
Step 18: Before configuring more file first we ensure which version of java we have installed for that go to the location */usr/lib/jvm* and after going to this location type *ls* command to list down the file inside it now see the java version, In my case it is **java-11-openjdk-amd64**.



```
hadoopusr@dikshant-Inspiron-5567:/home/dikshant/Documents$ cd /usr/lib/jvm
hadoopusr@dikshant-Inspiron-5567:/usr/lib/jvm$ ls
default-java  java-1.11.0-openjdk-amd64  java-11-openjdk-amd64  openjdk-11
hadoopusr@dikshant-Inspiron-5567:/usr/lib/jvm$ 
```

Step 19: Now we will configure *hadoop-env.sh*. For that open the file using below command.

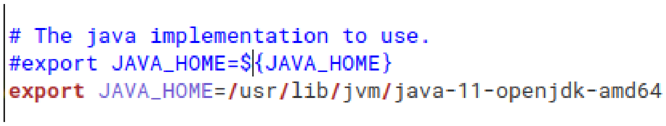
```
sudo vim /usr/local/hadoop/etc/hadoop/hadoop-env.sh
```



```
hadoopusr@dikshant-Inspiron-5567:~$ sudo gedit /usr/local/hadoop/etc/hadoop/hadoop-env.sh

** (gedit:14032): WARNING **: 14:09:53.607: Set document metadata failed: Setting attribute metadata::gedit-spell-la
nguage not supported
```

Once the file opened, copy the below export command inside it and make sure to comment the already existing export command with **JAVA_HOME**:

```
export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
```

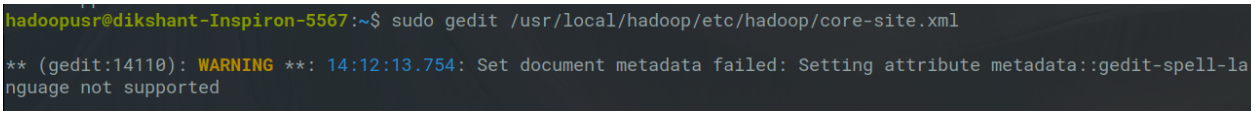


```
# The java implementation to use.
#export JAVA_HOME=${JAVA_HOME}
export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
```

Don't forget to **save**.

Step 20: Now we will configure the core-site.xml. For that open that file using below command:

```
sudo vim /usr/local/hadoop/etc/hadoop/core-site.xml
```

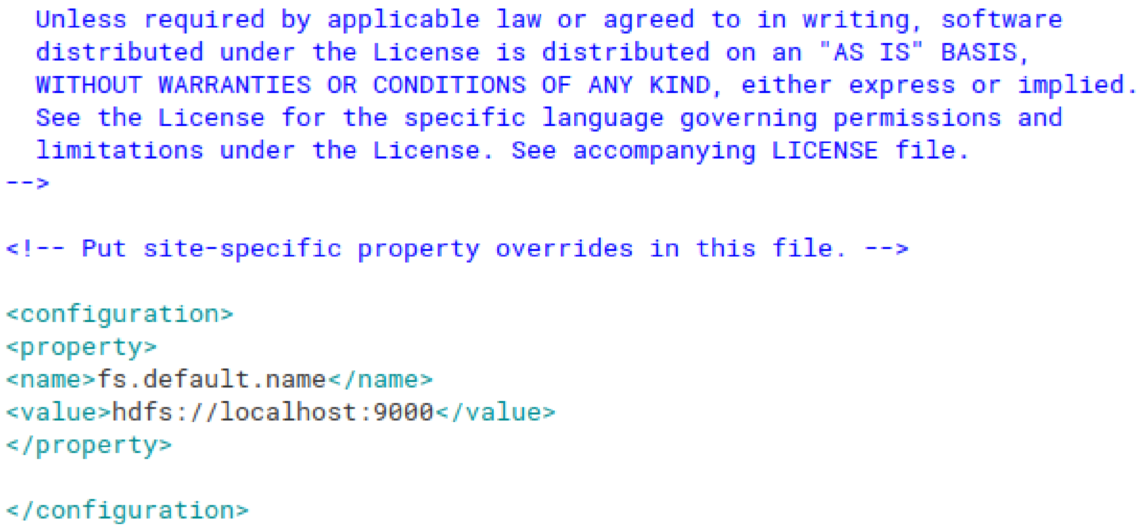


```
hadoopusr@dikshant-Inspiron-5567:~$ sudo gedit /usr/local/hadoop/etc/hadoop/core-site.xml

** (gedit:14110): WARNING **: 14:12:13.754: Set document metadata failed: Setting attribute metadata::gedit-spell-la
nguage not supported
```

once the file opens copy the below text **inside the configuration tag**

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

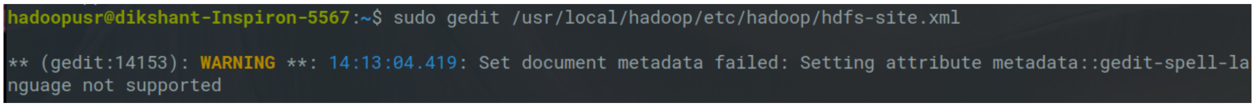
See the below image for better understanding:



```
  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>
```

Step 21: Now we will configure the *hdfs-site.xml* for that open that file using below command.

```
sudo vim /usr/local/hadoop/etc/hadoop/hdfs-site.xml
```

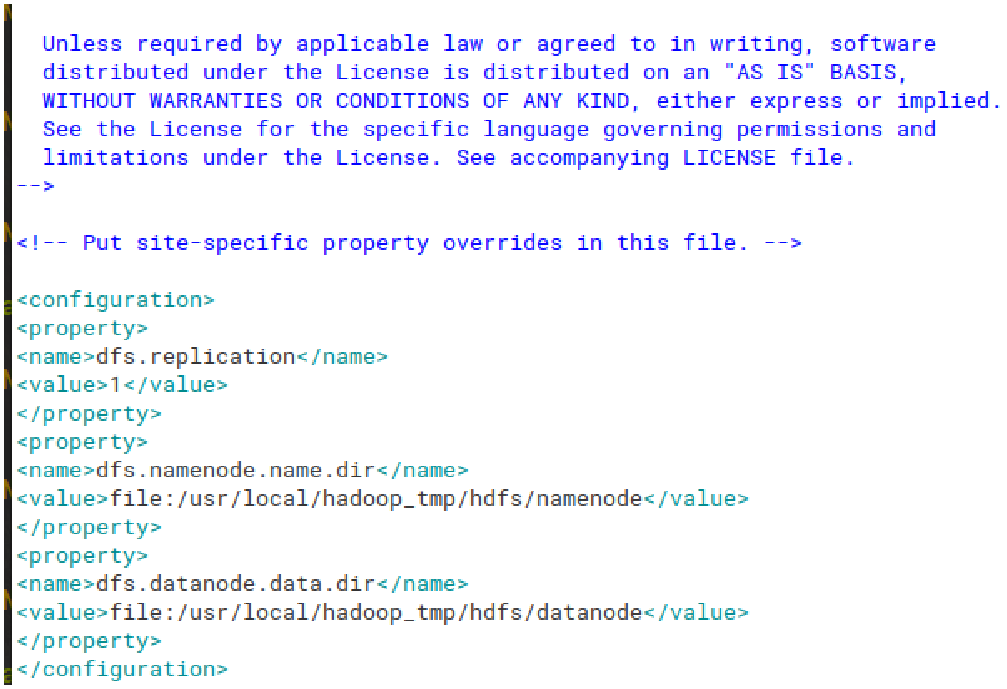


```
hadoopusr@dikshant-Inspiron-5567:~$ sudo gedit /usr/local/hadoop/etc/hadoop/hdfs-site.xml

** (gedit:14153): WARNING **: 14:13:04.419: Set document metadata failed: Setting attribute metadata::gedit-spell-la
nguage not supported
```

Once the file opens copy the below text **inside the configuration tag**

```
<!--
<property>
<name>dfs.replication</name>
<value>1</value>
</property>
<property>
<name>dfs.namenode.name.dir</name>
<value>file:/usr/local/hadoop_tmp/hdfs/namenode</value>
</property>
<property>
<name>dfs.datanode.data.dir</name>
<value>file:/usr/local/hadoop_tmp/hdfs/datanode</value>
</property>
 -->
```

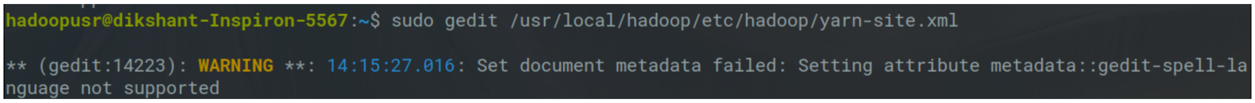
See the below image for better understanding:



```
  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>dfs.replication</name>
<value>1</value>
</property>
<property>
<name>dfs.namenode.name.dir</name>
<value>file:/usr/local/hadoop_tmp/hdfs/namenode</value>
</property>
<property>
<name>dfs.datanode.data.dir</name>
<value>file:/usr/local/hadoop_tmp/hdfs/datanode</value>
</property>
</configuration>
```

Step 22: Now we will configure the yarn-site.xml which is responsible for the execution of file in the Hadoop environment. For that open that file using below command:

```
sudo vim /usr/local/hadoop/etc/hadoop/yarn-site.xml
```

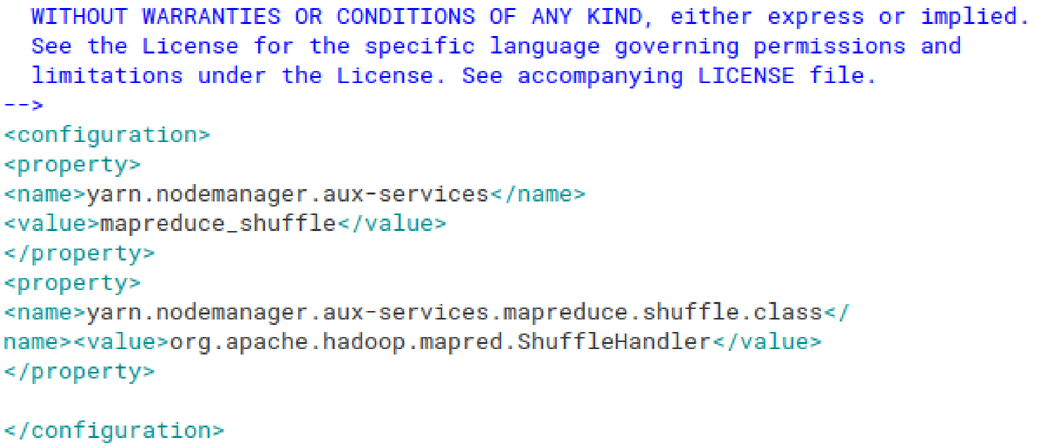


```
hadoopusr@dikshant-Inspiron-5567:~$ sudo gedit /usr/local/hadoop/etc/hadoop/yarn-site.xml

** (gedit:14223): WARNING **: 14:15:27.016: Set document metadata failed: Setting attribute metadata::gedit-spell-la
nguage not supported
```

once the file opens copy the below text **inside the configuration tag**

```
<!--

<property>

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

<value>mapreduce_shuffle</value>

</property>

<property>

<name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</n
ame>

<value>org.apache.hadoop.mapred.ShuffleHandler</value>

</property>

 -->
```

See the below image for better understanding:

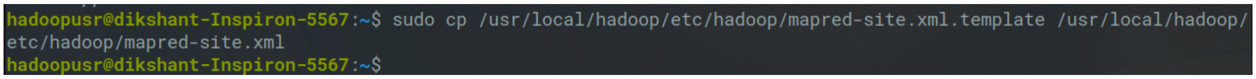


```
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->
<configuration>
<property>
<name>yarn.nodemanager.aux-services</name>
<value>mapreduce_shuffle</value>
</property>
<property>
<name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</
name><value>org.apache.hadoop.mapred.ShuffleHandler</value>
</property>

</configuration>
```

Step 23: Now the last file to configure is mapred-site.xml. For that we have *mapred-site.xml.template* so we need to locate that file then copy this file to that location and then *rename* it.

So to locate the file we need to go to the location */usr/local/hadoop/etc/hadoop/* so to copy this file and also rename the file the single, use the following command

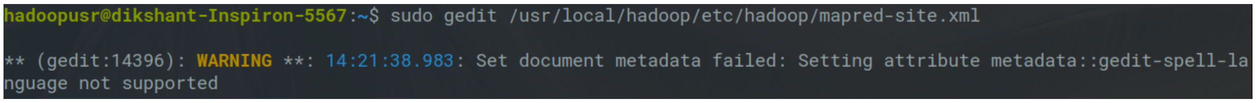
```
     sudo cp
     /usr/local/hadoop/etc/hadoop/mapred-site.xml.template
     /usr/local/hadoop/etc/hadoop/mapred-site.xml
```



```
hadoopusr@dikshant-Inspiron-5567:~$ sudo cp /usr/local/hadoop/etc/hadoop/mapred-site.xml.template /usr/local/hadoop/
etc/hadoop/mapred-site.xml
hadoopusr@dikshant-Inspiron-5567:~$
```

once the file gets copied or renamed now open that file using the following command:

```
sudo vim /usr/local/hadoop/etc/hadoop/mapred-site.xml
```

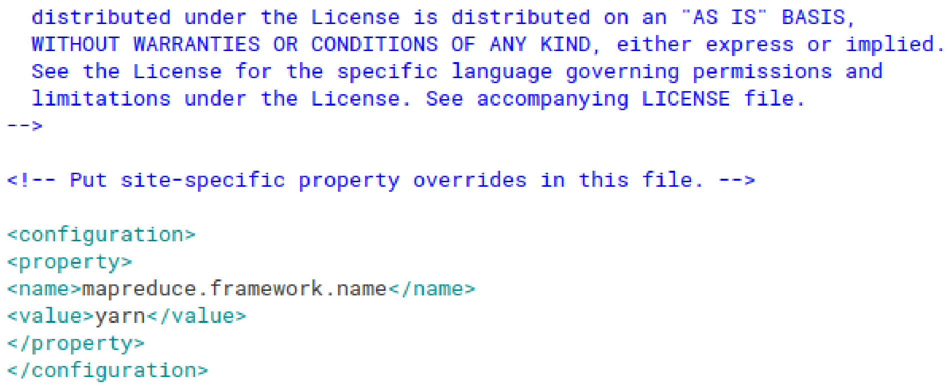


```
hadoopusr@dikshant-Inspiron-5567:~$ sudo gedit /usr/local/hadoop/etc/hadoop/mapred-site.xml

** (gedit:14396): WARNING **: 14:21:38.983: Set document metadata failed: Setting attribute metadata::gedit-spell-la
nguage not supported
```

And then place the below content inside its **configuration tag**.

```
 <!--

 <property>

 <name>mapreduce.framework.name</
 name>

 <value>yarn</value>

 </property>

  -->
```

See the below image for better understanding:



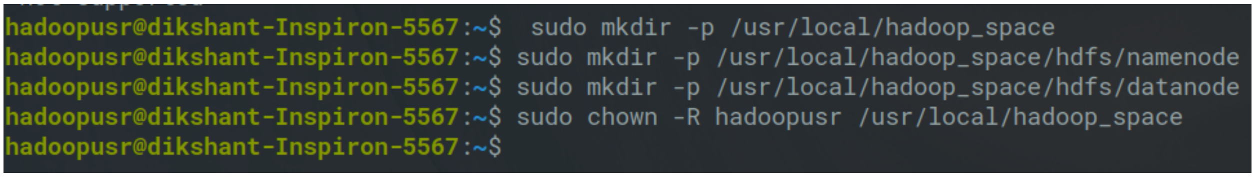
```
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

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

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

Step 24: Now we have successfully configured all the files. So now it is time to check our installation. As we know that in Hadoop architecture we have name node and other blocks so we need to make one directory i.e. hadoop_space. Inside this directory we make another directory i.e. hdfs and namenode and datanode. The command to make directory is given below:

```
sudo mkdir -p /usr/local/hadoop_space

sudo mkdir -p /usr/local/hadoop_space/hdfs/namenode

sudo mkdir -p /usr/local/hadoop_space/hdfs/datanode
```

Now we need to give permission for that commands are below:

```
sudo chown -R hadoopusr /usr/local/hadoop_space
```

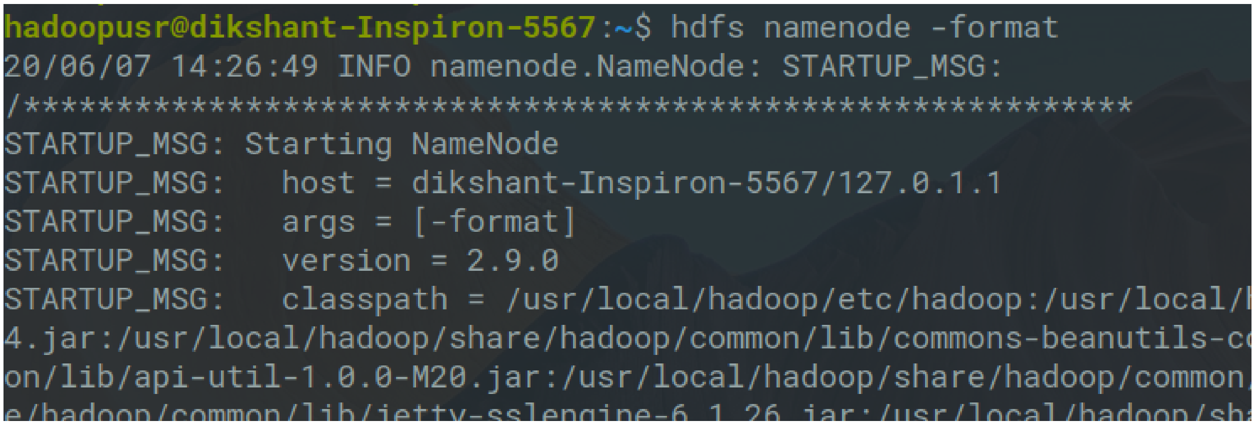


```
hadoopusr@dikshant-Inspiron-5567:~$  sudo mkdir -p /usr/local/hadoop_space
hadoopusr@dikshant-Inspiron-5567:~$ sudo mkdir -p /usr/local/hadoop_space/hdfs/namenode
hadoopusr@dikshant-Inspiron-5567:~$ sudo mkdir -p /usr/local/hadoop_space/hdfs/datanode
hadoopusr@dikshant-Inspiron-5567:~$ sudo chown -R hadoopusr /usr/local/hadoop_space
hadoopusr@dikshant-Inspiron-5567:~$
```

Running Hadoop

1. First, we need to format the namenode then you need to run the below command for first time when you starting the cluster if you use it again then all your metadata will get erase.

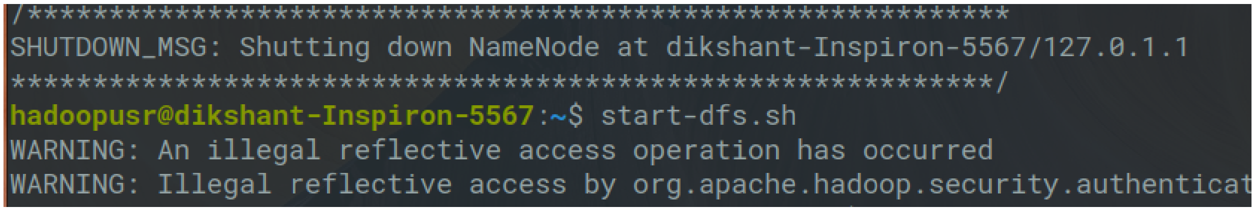
```
hdfs namenode -format
```



```
hadoopusr@dikshant-Inspiron-5567:~$ hdfs namenode -format
20/06/07 14:26:49 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = dikshant-Inspiron-5567/127.0.1.1
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 2.9.0
STARTUP_MSG:   classpath = /usr/local/hadoop/etc/hadoop:/usr/local/h
4.jar:/usr/local/hadoop/share/hadoop/common/lib/commons-beanutils-co
on/lib/api-util-1.0.0-M20.jar:/usr/local/hadoop/share/hadoop/common/
e/hadoop/common/lib/jetty-sslengine-6.1.26.jar:/usr/local/hadoop/sha
```

2. Now we need to start the DFS i.e. Distributed File System.

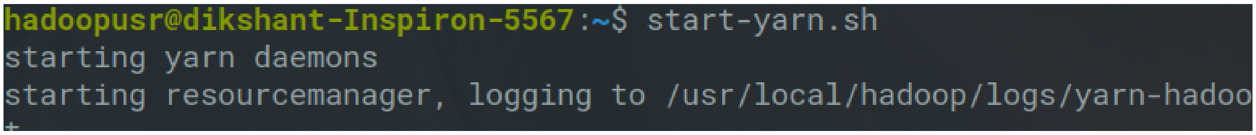
```
start-dfs.sh
```



```
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at dikshant-Inspiron-5567/127.0.1.1
************************************************************/
hadoopusr@dikshant-Inspiron-5567:~$ start-dfs.sh
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authenticat
```

3. now the last thing you need to start is *yarn*

```
start-yarn.sh
```

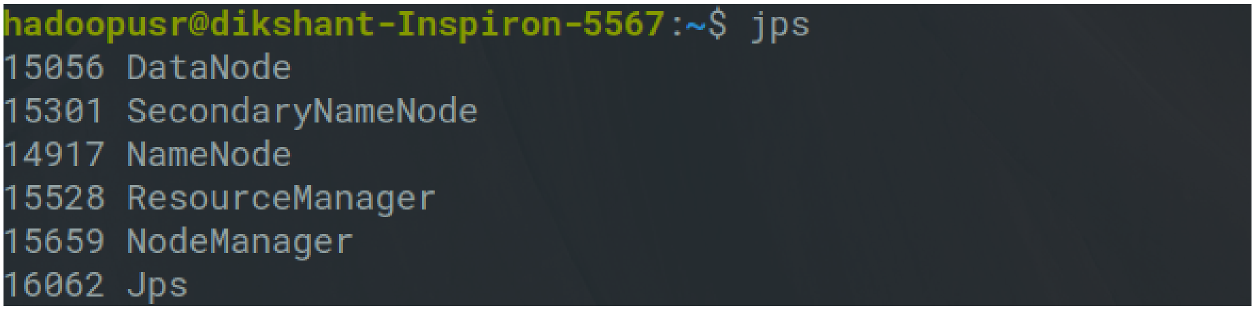


```
hadoopusr@dikshant-Inspiron-5567:~$ start-yarn.sh
starting yarn daemons
starting resourcemanager, logging to /usr/local/hadoop/logs/yarn-hadoo
```

4. Now use the following command:

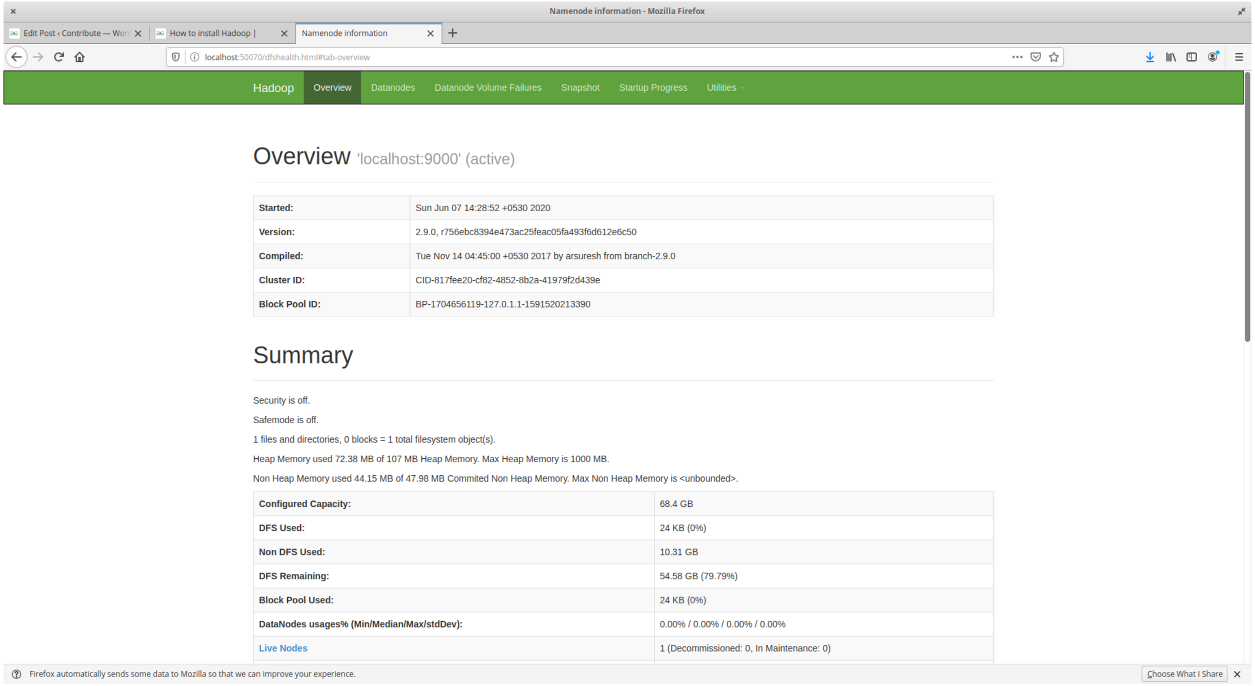
```
jps
```

Now you will be able to see the SecondaryNameNode, NodeManager, ResourceManager, NameNode, jpd, and DataNode which means you will have successfully installed Hadoop.



```
hadoopusr@dikshant-Inspiron-5567:~$ jps
15056 DataNode
15301 SecondaryNameNode
14917 NameNode
15528 ResourceManager
15659 NodeManager
16062 Jps
```

5. You have successfully installed *hadoop* on your system. Now to check all you cluster information you can use **localhost:50070** in your browser. The Interface will look like as:



Namenode information - Mozilla Firefox
Edit Post ‹ Contribute — Wor
How to install Hadoop |
Namenode information
localhost:50070/dfshealth.html#tab-overview
Hadoop
Overview
Datanodes
Datanode Volume Failures
Snapshot
Startup Progress
Utilities
Firefox automatically sends some data to Mozilla so that we can improve your experience.
Choose What I Share

Overview 'localhost:9000' (active)

Started:	Sun Jun 07 14:28:52 +0530 2020
Version:	2.9.0, r756ebc8394e473ac25feac05fa493f6d612e6c50
Compiled:	Tue Nov 14 04:45:00 +0530 2017 by arsuresh from branch-2.9.0
Cluster ID:	CID-817fee20-cf82-4852-8b2a-41979f2d439e
Block Pool ID:	BP-1704656119-127.0.1.1-1591520213390

Summary

Security is off.

Safemode is off.

1 files and directories, 0 blocks = 1 total filesystem object(s).

Heap Memory used 72.38 MB of 107 MB Heap Memory. Max Heap Memory is 1000 MB.

Non Heap Memory used 44.15 MB of 47.98 MB Commited Non Heap Memory. Max Non Heap Memory is <unbounded>.

Configured Capacity:	68.4 GB
DFS Used:	24 KB (0%)
Non DFS Used:	10.31 GB
DFS Remaining:	54.58 GB (79.79%)
Block Pool Used:	24 KB (0%)
DataNodes usages% (Min/Median/Max/stdDev):	0.00% / 0.00% / 0.00% / 0.00%
Live Nodes	1 (Decommissioned: 0, In Maintenance: 0)

Note: If your localhost is not connecting then make sure you have placed the correct version of java in the property files.

Source: https://www.geeksforgeeks.org/how-to-install-hadoop-in-linux/