```
 # Step 1 : Commands for removeing lock
       cd ~
        sudo rm /var/lib/apt/lists/lock
        sudo rm /var/cache/apt/archives/lock
        sudo rm /var/lib/dpkg/lock
         sudo rm /var/lib/dpkg/lock-frontend

# Step 2 : Installation of JAVA

        sudo apt-get update
        sudo apt-get install openjdk-8-jdk
        java --version


#Step 3: Adding a dedicated Hadoop Group

       sudo addgroup hadoop


# Step 4 : Installing SSH, Create and Setup SSH Certificates


       sudo apt-get install ssh

     #SSH certificate generation

       ssh-keygen -t rsa -P ""

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

     # to check SSH works or not

       ssh localhost


#Step 5: Install Hadoop

    #select the dir

     cd mapreduce/software/

    #unzip hadoop-2.6.5.tar.gz

      tar xvzf hadoop-2.6.5.tar.gz

    #make a hadoop dir under usr/local dir

        sudo mkdir -p /usr/local/hadoop

    # move to hadoop-2.6.5

       cd hadoop-2.6.5

       sudo mv * /usr/local/hadoop

    #change ownership rights

      sudo chown -R hduser:hadoop /usr/local/hadoop
```

#Step 6: Hadoop Setup Configuration Files

```
sudo nano ~/.bashrc
```

 # insert the following HADOOP VARIABLE export commands in that file

```
#HADOOP VARIABLES START
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
#export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-i386
export HADOOP_INSTALL=/usr/local/hadoop
export PATH=$PATH:$HADOOP_INSTALL/bin
export PATH=$PATH:$HADOOP_INSTALL/sbin
export HADOOP_MAPRED_HOME=$HADOOP_INSTALL
export HADOOP_COMMON_HOME=$HADOOP_INSTALL
export HADOOP_HDFS_HOME=$HADOOP_INSTALL
export YARN_HOME=$HADOOP_INSTALL
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_INSTALL/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_INSTALL/lib"
export HADOOP_CLASSPATH=$(hadoop classpath)
#HADOOP VARIABLES END
```

#execute .bashrc

```
source ~/.bashrc
```

 # edit hadoop-env.sh file

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

#insert the following export command in that file

```
 export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
```

 # To override configuration settings of core-site.xml:

 # create the /app/hadoop/tmp directory to be used to override default settings that Hadoop starts

```
sudo mkdir -p /app/hadoop/tmp
```

 # change the ownership to hduser in hadoop group

```
sudo chown hduser:hadoop /app/hadoop/tmp
```

 # edit core-site.xml file

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

#insert the following statements in that file in between <configuration> </configuration>

```
 <property>
  <name>hadoop.tmp.dir</name>
  <value>/app/hadoop/tmp</value>
  <description>A base for other temporary directories.</description>
 </property>

 <property>
  <name>fs.default.name</name>
  <value>hdfs://localhost:54310</value>
  <description>The name of the default file system.  A URI whose
  scheme and authority determine the FileSystem implementation.  The
  uri's scheme determines the config property (fs.SCHEME.impl) naming
  the FileSystem implementation class.  The uri's authority is used to
  determine the host, port, etc. for a filesystem.</description>
 </property>
```

 # edit mapred-site.xml

 # copy mapred-site.xml.template to mapred-site.xml

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

 # edit core-site.xml file

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

#insert the following statements in that file in between <configuration> </configuration>

```
<property>
  <name>mapred.job.tracker</name>
  <value>localhost:54311</value>
  <description>The host and port that the MapReduce job tracker runs
  at.  If "local", then jobs are run in-process as a single map
  and reduce task.
  </description>
</property>

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

 # create namenode, datanode and change the ownership of hadoop_store to hduser in hadoop group

```
     sudo mkdir -p /usr/local/hadoop_store/hdfs/namenode
     sudo mkdir -p /usr/local/hadoop_store/hdfs/datanode
     sudo chown -R hduser:hadoop /usr/local/hadoop_store
```

 # edit hdfs-site.xml

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

#insert the following statements in that file in between <configuration> </configuration>

```
<property>
  <name>dfs.replication</name>
  <value>1</value>
  <description>Default block replication.
  The actual number of replications can be specified when the file is created.
  The default is used if replication is not specified in create time.
  </description>
 </property>
 <property>
   <name>dfs.namenode.name.dir</name>
   <value>file:/usr/local/hadoop_store/hdfs/namenode</value>
 </property>
 <property>
   <name>dfs.datanode.data.dir</name>
   <value>file:/usr/local/hadoop_store/hdfs/datanode</value>
 </property>
```

 # Format the New Hadoop Filesystem

```
hadoop namenode -format
```

 # edit yarn-site.xml

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

#insert the following statements in that file in between <configuration> </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>
```

#Step 7 : Starting Hadoop

```
start-all.sh
```

or

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

 # to check the execution

```
jps
```

 # display as follows
14306 DataNode
14660 ResourceManager
14505 SecondaryNameNode
14205 NameNode
14765 NodeManager
15166 Jps

#web UI of the NameNode daemon - Type http://localhost:50070/ as url into our browser

#Step 8 : Stoping Hadoop

stop-all.sh

or

stop-dfs.sh
stop-yarn.sh