### **Install java with the commands below with the admin/root password.**

sudo apt-get update && apt-get dist-upgrade

sudo apt-get install openjdk-8-jdk

### **To create JAVA\_HOME environment variable**

cd

vi .bashrc

export JAVA\_HOME=/usr/lib/jvm/java-8-openjdk-amd64

### **To save the environment variable - Esc:wq**

source .bashrc

### **Install openssh to ssh with via port 22 (run on terminal)**

sudo apt install openssh-server openssh-client -y

### **Configure password-less SSH (run on terminal)**

cat /dev/zero | ssh-keygen -q -N ""

cat ~/.ssh/id\_rsa.pub >> ~/.ssh/authorized\_keys

### **Installing Hadoop/HDF without hadoop user in /opt**

cd /opt

### **Download Hadoop (run on terminal)**

wget <https://archive.apache.org/dist/hadoop/common/hadoop-3.3.0/hadoop-3.3.0.tar.gz>

tar -xzvf hadoop-3.3.0.tar.gz

### **Create symbolic link (run on terminal)**

ln -s hadoop-3.3.0 hadoop

### **Add hadoop variables setup**

cd

vi .bashrc

# set up HADOOP\_HOME

export HADOOP\_HOME=/opt/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/bin

export PATH=$PATH:$HADOOP\_HOME/sbin

export HADOOP\_CONF\_DIR=$HADOOP\_HOME/etc/hadoop

export HADOOP\_OPTS="-Djava.library.path=$HADOOP\_HOME/lib/native"

### **Make hadoop and java variables available to system (run on terminal)**

source .bashrc

### **Now let's edit our configurations (run on terminal)**

cd /opt/hadoop/etc/hadoop/

### **Edit hadoop-env.sh inline (run on terminal)**

vi hadoop-env.sh

export JAVA\_HOME=/usr/lib/jvm/java-8-openjdk-amd64

export HADOOP\_CONF\_DIR=/opt/hadoop/etc/hadoop

**To save the environment variable - Esc:wq**

### 

### 

### 

### 

### **Edit core-site.xml inline (run on terminal)**

vi core-site.xml

<configuration>

<property>

<name>fs.defaultFS</name>

<value>hdfs://localhost:9000</value>

</property>

<property>

<name>hadoop.tmp.dir</name>

<value>/opt/hadoop/hadoop\_tmp</value>

</property>

</configuration>

**To save - Esc:wq**

mkdir /opt/hadoop/hadoop\_tmp

### **Edit hdfs-site.xml (run on terminal)**

vi hdfs-site.xml

<configuration>

<property>

<name>dfs.replication</name>

<value>1</value>

</property>

<property>

<name>dfs.name.dir</name>

<value>file:///opt/hadoop/hdfs/namenode</value>

</property>

<property>

<name>dfs.data.dir</name>

<value>file:///opt/hadoop/hdfs/datanode</value>

</property>

</configuration>

**To save - Esc:wq**

### **create datanode and namenode directory (run on terminal)**

**mkdir -p /opt/hadoop/hdfs/namenode**

**mkdir -p /opt/hadoop/hdfs/datanode**

### **Edit mapred-site.xml (run on terminal)**

vi mapred-site.xml

<configuration>

<property>

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

<value>yarn</value>

</property>

</configuration>

**To save - Esc:wq**

### **Edit yarn-site.xml (run on terminal)**

Vi yarn-site.xml

<configuration>

<property>

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

<value>mapreduce\_shuffle</value>

</property>

</configuration>

**To save - Esc:wq**

### **Test hadoop and hdfs variables (run on terminal)**

hadoop version

hdfs version

### 

### **Formatting hadoop/hdfs (run on terminal)**

hdfs namenode -format

**should have status 0 at the end**

### **Start HDFS services (run on terminal)**

start-dfs.sh

start-yarn.sh

### **Test hadoop / HDFS is working (run on terminal)**

hdfs dfs -mkdir /tmp /user /data /hive

hdfs dfs -ls /

### **Congratulations, hadoop/HDFS is installed.**

### **Check hadoop UI at localhost:9870**

### 