# Big Data (ECE 595-004/007), Fall 2018
# Hands-on 1, Part2: Hadoop Set-up (15 pts)
### Due date: September 21, 2018, 11:59 PM

This assignment is straightforward. You have to set-up Hadoop in your virtual machine. For grading, submit the screenshot of the steps that you followed for set-up in one word file and upload the PDF for it.

## Virtual machine

The tutorial for VMWare player has been given in the tutorial document.
Those who have Mac, they can install Virtual Box and create a virtual machine using it.
Link for Virtual Box installation: https://www.youtube.com/watch?v=lEvM-No4eQo
Download Ubuntu 16: http://releases.ubuntu.com/16.04/ubuntu-16.04.5-desktop-amd64.iso if your machine is 64-bit or http://releases.ubuntu.com/16.04/ubuntu-16.04.5-desktop-i386.iso if your machine is 32-bit.
Create Ubuntu virtual machine: https://www.youtube.com/watch?v=fh8OdDd0K30

## Hadoop set-up

1. Install Java: `sudo apt-get install default-jdk`
   You can confirm the installation: `java -version`
2. Install SSH server: `sudo apt-get install openssh-server`
3. Download Hadoop 2.9.1 from the following link:
   https://www.apache.org/dyn/closer.cgi/hadoop/common/hadoop-2.9.1/hadoop-2.9.1.tar.gz
4. Go to the Downloads directory: `cd Downloads`
5. Extract Hadoop files from the downloaded file: `tar xzvf hadoop-2.9.1.tar.gz`
6. Move extracted Hadoop directory to /usr/local directory:
   ```
   sudo mv hadoop-2.9.1 /usr/local/hadoop
   ```
   Change ownership of Hadoop directory:
   ```
   sudo chown -R bigdata:bigdata /usr/local/hadoop
   ```
   Note: **bigdata** is a user here. It may be different in your case.
7. Find directory for Java: `readlink -f /usr/bin/java | sed "s:/bin/java::"`
   The command output will be the directory for Java.
8. Go to home directory: `cd`
9. Open .bashrc file: `gedit .bashrc`
   Add following lines:
   ```
   export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64/jre
   export HADOOP_HOME=/usr/local/Hadoop
   export PATH=$PATH:$HADOOP_HOME/bin:$HADOOP_HOME/sbin
   ```
   Note: Value for JAVA_HOME is the output of the previous command for finding Java directory
   Close .bashrc file and execute command: `. .bashrc`
   You can verify the changes with the command: `echo $JAVA_HOME`
10. Open /usr/local/hadoop/etc/hadoop/hadoop-env.sh:
    ```
    gedit /usr/local/hadoop/etc/hadoop/hadoop-env.sh
    ```

Replace **export JAVA_HOME=${JAVA_HOME}** with

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

Also add this: **export HADOOP_OPTS=-Djava.net.preferIPv4Stack=true**

Close the file

11. Create a directory /app/hadoop/tmp: **sudo mkdir -p /app/hadoop/tmp**
    Change ownership: **sudo chown –R bigdata:bigdata /app/hadoop/tmp**
12. Open /usr/local/hadoop/etc/hadoop/core-site.xml:
    ```
    gedit /usr/local/hadoop/etc/hadoop/core-site.xml
    ```
    Add these lines inside <configuration> </configuration> tags
    ```
    <property>
    <name>hadoop.tmp.dir</name>
    <value>/app/hadoop/tmp</value>
    </property>
    <property>
    <name>fs.default.name</name>
    <value>hdfs://localhost:9000</value>
    </property>
    ```
    Close the file
13. Open /usr/local/hadoop/etc/hadoop/hdfs-site.xml:
    ```
    gedit /usr/local/hadoop/etc/hadoop/hdfs-site.xml
    ```
    Add these lines inside <configuration> </configuration> tags
    ```
    <property>
    <name>dfs.replication</name>
    <value>1</value>
    </property>
    ```
    Close the file.
14. Open /usr/local/hadoop/etc/hadoop/mapred-site.xml:
    ```
    gedit /usr/local/hadoop/etc/hadoop/mapred-site.xml
    ```
    Add these lines inside <configuration> </configuration> tags
    ```
    <property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
    </property>
    ```
    Note: mapred-site.xml is not available. Rename mapred-site.xml.template file to mapred-site.xml file: mv mapred-site.xml.template mapred-site.xml
    Close the file.
15. Open /usr/local/hadoop/etc/hadoop/yarn-site.xml:
    ```
    gedit /usr/local/hadoop/etc/hadoop/yarn-site.xml
    ```
    Add these lines inside <configuration> </configuration> tags
    ```
    <property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
    </property>
    ```
16. Perform following commands to set-up SSH public key authentication
    ```
    ssh-keygen -t rsa -P ""
    cat /home/bigdata/.ssh/id_rsa.pub >> /home/bigdata/.ssh/authorized_keys
    ```
    Note: **bigdata** is a user here. It may be different in your case.
17. Format HDFS: **hadoop namenode -format**
18. Start service: **start-all.sh**
19. Execute command to check services are running: **jps**

20. Once your job is done, you can stop services: `stop-all.sh`
    Do not execute this command now, first do something in HDFS cluster as described below.

## HDFS command

1. Assume you are at your home directory or execute: `cd`
2. Examine files in HDFS cluster: `hdfs dfs -ls`
   You will see nothing in the output as you have not created anything yet.
3. Create a directory in HDFS cluster: `hdfs dfs -mkdir /user`
4. Examine files again in HDFS cluster: `hdfs dfs -ls`
   You will see the listing of /user directory in the output
5. Create a directory inside /user directory using your user name:
   ```
   hdfs dfs -mkdir /user/bigdata
   ```
6. Examine that directory in HDFS cluster: `hdfs dfs -ls /user/`
7. Create a text file, say `demo.txt`, using gedit and write something in it and save it.
8. Upload the file in the cluster:
   ```
   hdfs dfs -copyFromLocal demo.txt /user/bigdata/
   ```
9. List the file in the cluster: `hdfs dfs -ls /user/bigdata/`
10. See the content of file: `hdfs dfs -cat /user/bigdata/demo.txt`