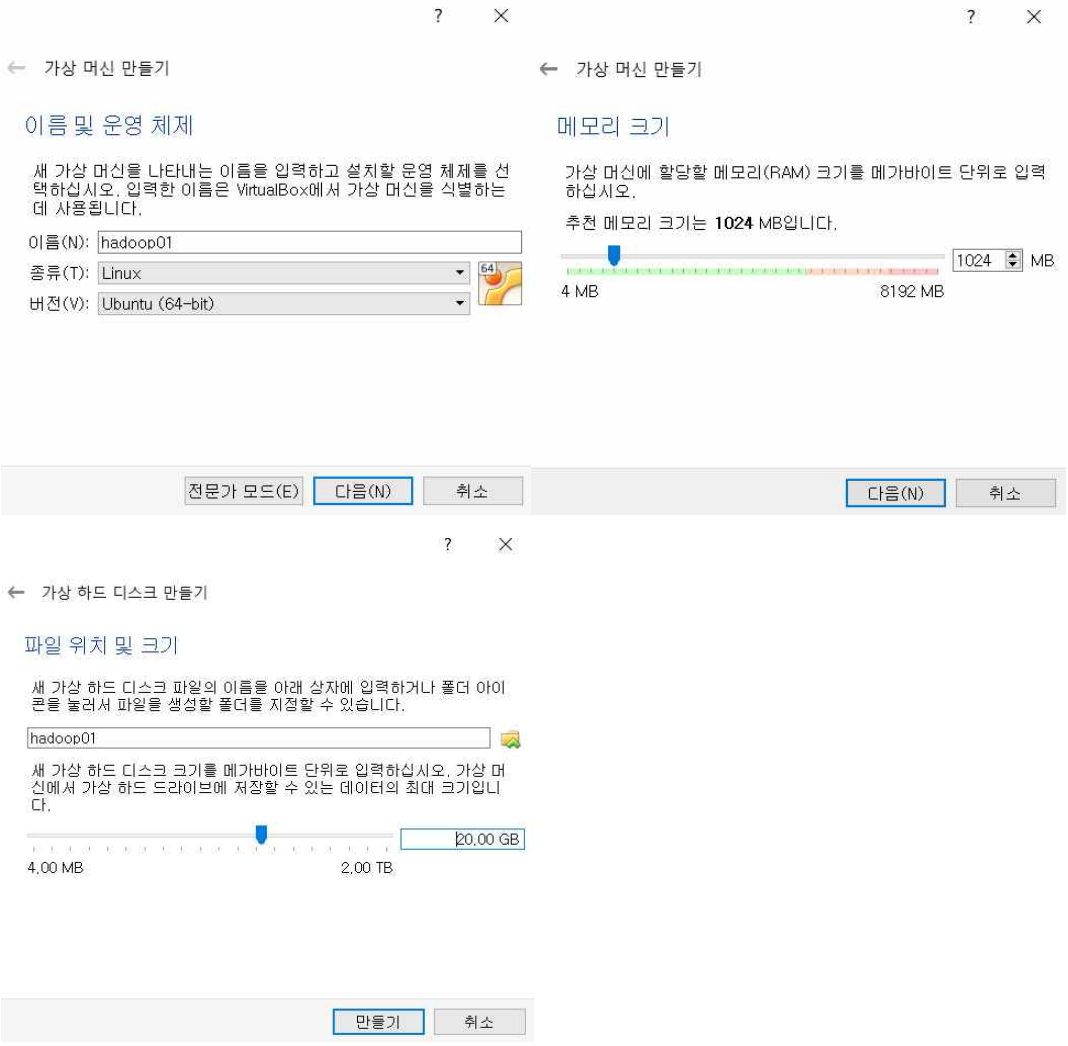
1. Hadoop Fully-Distribution Setting

hadoop01	192.168.142.101	namenode, secondarynamenode, nodemanager, datanode
hadoop02	192.168.142.102	resourcemanager, nodemanager, datanode
hadoop03	192.168.142.103	nodemanager, datanode
hadoop04	192.168.142.104	nodemanager, datanode

VM : VirtualBox

OS : Linux-mint-18.1-cinnamon-64bit

1-1. 가상머신 만들고 OS 설치



가상머신 이름 : hadoop01

메모리 : 1GB

하드 디스크 : 20GB

username : hadoop

hostname : hadoop01

1-2. java, ssh, rsync 설치

기존 openjdk 1.8 제거

```
hadoop@hadoop01 ~ $ sudo apt-get remove openjdk-8*
```

```
sudo apt-get remove openjdk-8*
```

오라클 자바 1.7버전 설치

```
hadoop@hadoop01 ~ $ sudo add-apt-repository ppa:webupd8team/java
hadoop@hadoop01 ~ $ sudo apt-get update
hadoop@hadoop01 ~ $ sudo apt-get install oracle-java7-installer
```

```
sudo add-apt-repository ppa:webupd8team/java
sudo apt-get update
sudo apt-get install oracle-java7-installer
```

~/.bashrc에 JAVA_HOME 등록

```
123 if [ -x /usr/bin/mint-fortune ]; then
124      /usr/bin/mint-fortune
125 fi
126
127 export JAVA_HOME=/usr/lib/jvm/java-7-oracle/
```

```
export JAVA_HOME=/usr/lib/jvm/java-7-oracle/
```

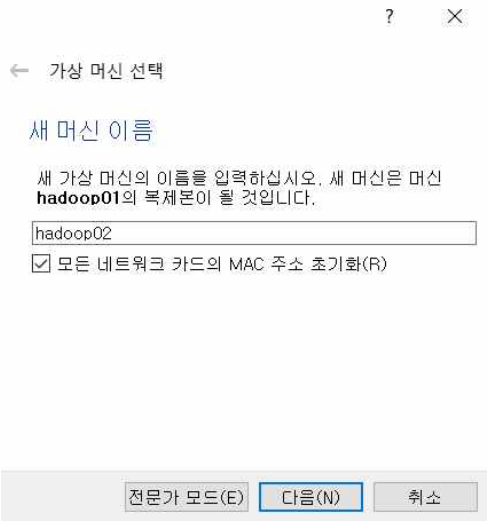
ssh, rsync 설치

```
hadoop@hadoop01 ~ $ sudo apt-get install ssh
hadoop@hadoop01 ~ $ sudo apt-get install rsync
```

```
sudo apt-get install ssh
sudo apt-get install rsync
```

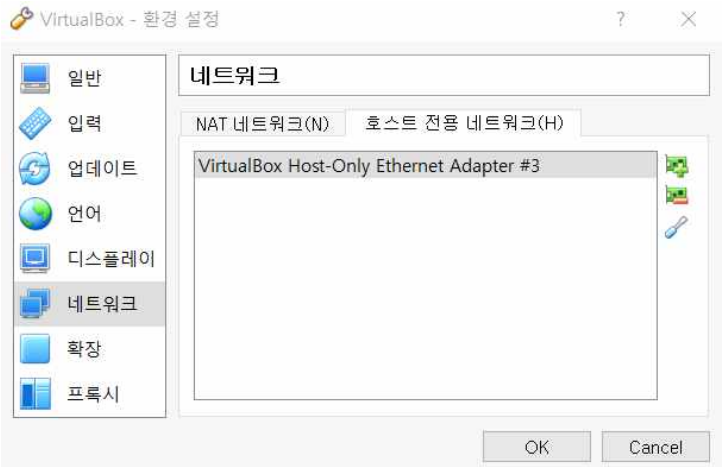
1-3. 가상머신 복제(hadoop02, hadoop03, hadoop04)

가상머신 hadoop01 우클릭 -> 이름변경, MAC주소 초기화 체크



1-4. 네트워크 설정

VirtualBox 파일 -> 환경설정 -> 네트워크 -> 호스트 전용 네트워크 선택

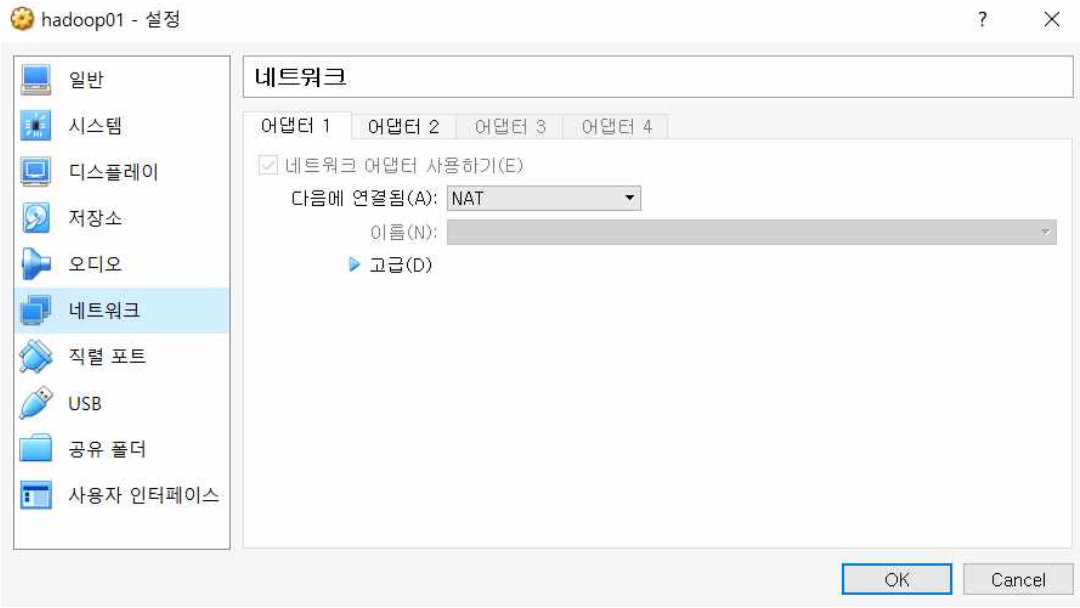


편집(드라이버모양) -> DHCP 서버 선택

아래와 같이 설정

hadoop01 설정 -> 네트워크

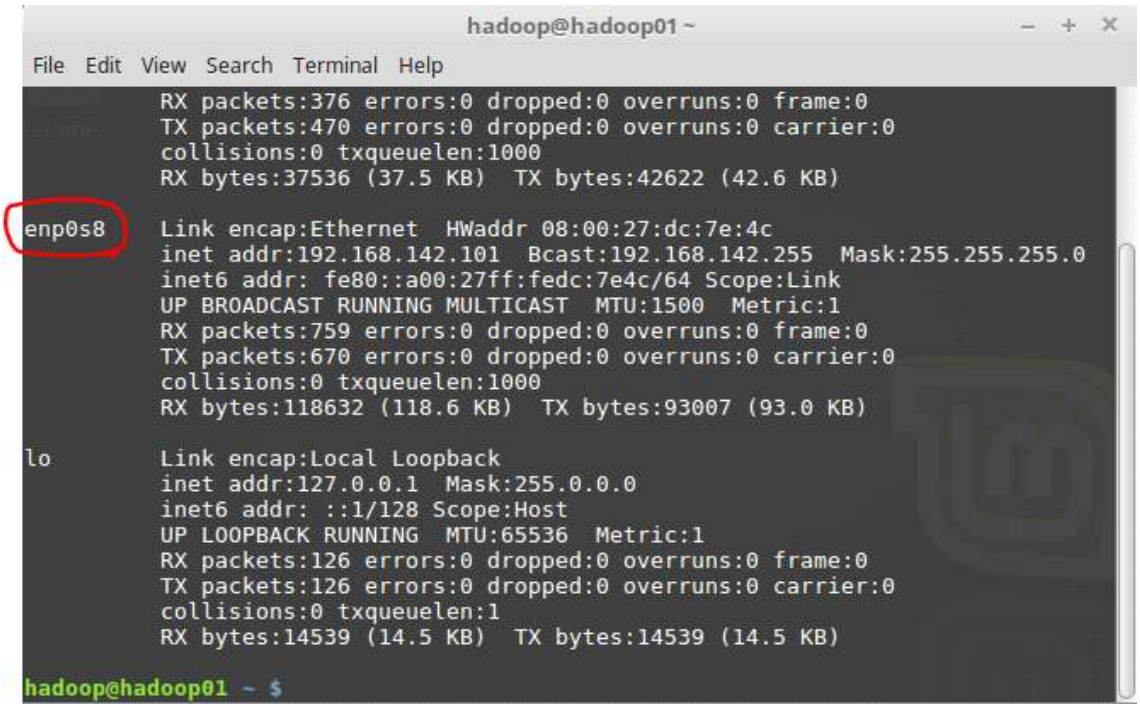
어댑터 1은 NAT, 어댑터 2는 호스트 전용 어댑터로 설정

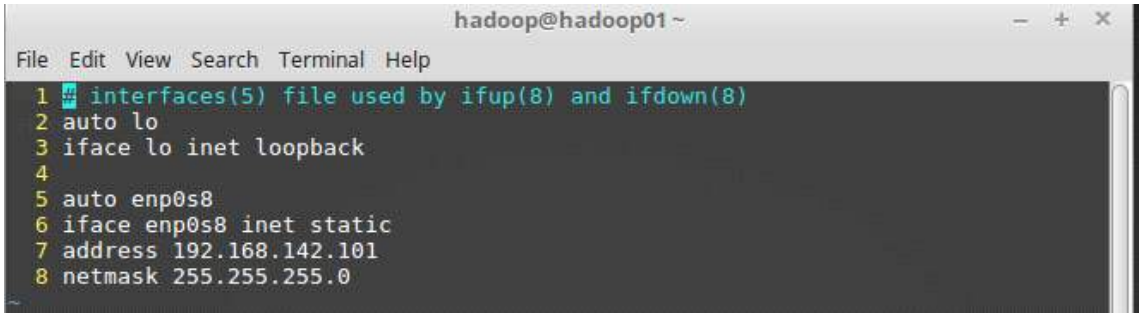


hadoop02, hadoop03, hadoop04 설정 -> 네트워크
호스트 전용 어댑터만 설정

가상머신에 고정 IP 할당
sudo vi /etc/network/interfaces -> 고정 IP할당

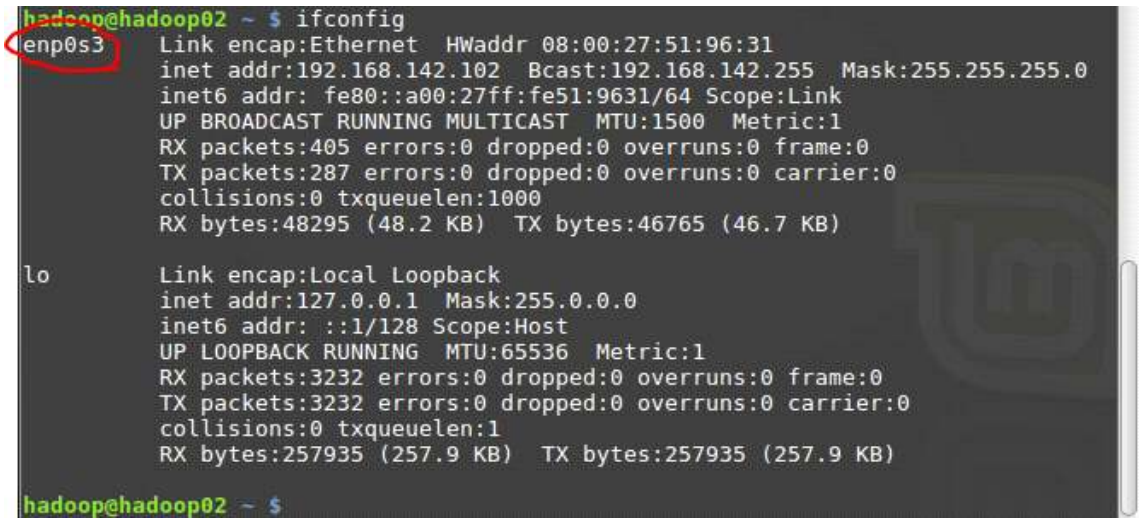
hadoop01

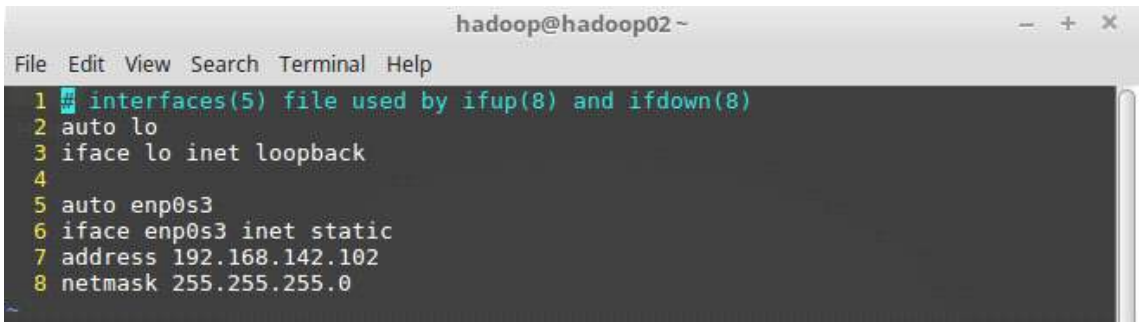





```
auto enp0s8
iface enp0s8 inet static
address 192.168.142.101
netmask 255.255.255.0
```

hadoop02, hadoop03, hadoop04






```
auto enp0s3
iface enp0s3 inet static
address 192.168.142.102
netmask 255.255.255.0
```

모든 가상머신에서

sudo vi /etc/hostname -> hostname 변경

```
1 hadoop02
```

sudo vi /etc/hosts -> hostname 변경

```
1 127.0.0.1   localhost
2 127.0.1.1   hadoop02
```

sudo vi /etc/hosts -> host 설정(127.0.1.1 주석처리)

```
1 127.0.0.1   localhost
2 #127.0.1.1  hadoop02
3
4 192.168.142.101 hadoop01
5 192.168.142.102 hadoop02
6 192.168.142.103 hadoop03
7 192.168.142.104 hadoop04
```

```
192.168.142.101 hadoop01
192.168.142.102 hadoop02
192.168.142.103 hadoop03
192.168.142.104 hadoop04
```

sudo vi /etc/hosts -> host 설정 후 재시작

hadoop01, hadoop02 ssh 설정 및 키 생성, 배포

```
hadoop@hadoop01 ~ $ ssh-keygen -t rsa
```

hadoop02, hadoop03, hadoop04에 ~/.ssh 디렉토리 생성

```
hadoop@hadoop01 ~/.ssh $ scp id_rsa.pub hadoop@hadoop02:/home/hadoop/.ssh/author
ized_keys
```

```
hadoop@hadoop01 ~/.ssh $ scp id_rsa.pub hadoop@hadoop03:/home/hadoop/.ssh/author
ized_keys
```

```
hadoop@hadoop01 ~/.ssh $ scp id_rsa.pub hadoop@hadoop04:/home/hadoop/.ssh/author
ized_keys
```

```
ssh-keygen -t rsa
cd ~/.ssh
scp id_rsa.pub /home/hadoop/.ssh/authorized_keys
scp id_rsa.pub hadoop@hadoop02:/home/hadoop/.ssh/authorized_keys
scp id_rsa.pub hadoop@hadoop03:/home/hadoop/.ssh/authorized_keys
scp id_rsa.pub hadoop@hadoop04:/home/hadoop/.ssh/authorized_keys
```

hadoop02의 키는 복사하여 authorized_keys에 붙여넣기

1-5. hadoop 설치

hadoop-2.7.3 설치

```
hadoop@hadoop01 ~ $ wget http://apache.mirror.cdnetworks.com/hadoop/common/hadoo
p-2.7.3/hadoop-2.7.3.tar.gz
```

```
hadoop@hadoop01 ~ $ tar -zxvf hadoop-2.7.3.tar.gz
```

```
wget http://apache.mirror.cdnetworks.com/hadoop/common/hadoop-2.7.3/hadoop-2.7.3.tar.gz
tar -zxvf hadoop-2.7.3.tar.gz
```

1-6. 환경변수 등록

hadoop-2.7.3/etc/hadoop 디렉토리에 있는 hadoop-env.sh, yarn-env.sh 파일에 JAVA_HOME 설정

hadoop-env.sh

```
24 # The java implementation to use.
25 export JAVA_HOME=/usr/lib/jvm/java-7-oracle/
```

```
export JAVA_HOME=/usr/lib/jvm/java-7-oracle/
```

yarn-env.sh

```
22 # some Java parameters
23 export JAVA_HOME=/usr/lib/jvm/java-7-oracle/
```

```
export JAVA_HOME=/usr/lib/jvm/java-7-oracle/
```

1-7. 하둡 설정

hadoop-2.7.3/etc/hadoop 디렉토리 내의 다음 파일들을 수정

core-site.xml

```
<configuration>
    <property>
        <name>fs.defaultFS</name>
        <value>hdfs://hadoop01:9000</value>
    </property>
</configuration>
```

```
<configuration>
    <property>
        <name>fs.defaultFS</name>
        <value>hdfs://hadoop01:9000</value>
    </property>
</configuration>
```

hdfs-site.xml

```
<configuration>
    <property>
        <name>dfs.replication</name>
        <value>4</value>
    </property>
    <property>
        <name>dfs.namenode.name.dir</name>
        <value>/home/hadoop/hadoop-2.7.3/hdfs/name</value>
    </property>
    <property>
        <name>dfs.datanode.data.dir</name>
        <value>/home/hadoop/hadoop-2.7.3/hdfs/data</value>
    </property>
</configuration>
```

```
<configuration>
    <property>
        <name>dfs.replication</name>
        <value>4</value>
    </property>
    <property>
        <name>dfs.namenode.name.dir</name>
        <value>/home/hadoop/hadoop-2.7.3/hdfs/name</value>
    </property>
    <property>
        <name>dfs.datanode.data.dir</name>
        <value>/home/hadoop/hadoop-2.7.3/hdfs/data</value>
    </property>
</configuration>
```

slaves

```
1 hadoop01
2 hadoop02
3 hadoop03
4 hadoop04
```

```
hadoop01
hadoop02
hadoop03
hadoop04
```

yarn-site.xml

```
<configuration>

<!-- Site specific YARN configuration properties -->
    <property>
        <name>yarn.nodemanager.aux-services</name>
        <value>mapreduce_shuffle</value>
    </property>
    <property>
        <name>yarn.resourcemanager.hostname</name>
        <value>hadoop02</value>
    </property>
</configuration>
```

```
<configuration>
    <property>
        <name>yarn.nodemanager.aux-services</name>
        <value>mapreduce_shuffle</value>
    </property>
    <property>
        <name>yarn.resourcemanager.hostname</name>
        <value>hadoop02</value>
    </property>
</configuration>
```

mapred-site.xml (cp mapred-site.xml.template mapred-site.xml)

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

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

1-8. 하둡 배포

하둡 디렉토리 배포

```
hadoop@hadoop01 ~ $ scp -r /home/hadoop/hadoop-2.7.3 hadoop@hadoop02:~
hadoop@hadoop01 ~ $ scp -r /home/hadoop/hadoop-2.7.3 hadoop@hadoop03:~
hadoop@hadoop01 ~ $ scp -r /home/hadoop/hadoop-2.7.3 hadoop@hadoop04:~
```

```
scp -r /home/hadoop/hadoop-2.7.3 hadoop@hadoop02:~
scp -r /home/hadoop/hadoop-2.7.3 hadoop@hadoop03:~
scp -r /home/hadoop/hadoop-2.7.3 hadoop@hadoop04:~
```

1-9. hadoop02, hadoop03, hadoop04 수정

hdfs-site.xml에서 namenode 삭제

```
<configuration>
    <property>
        <name>dfs.replication</name>
        <value>4</value>
    </property>
    <property>
        <name>dfs.datanode.data.dir</name>
        <value>/home/hadoop/hadoop-2.7.3/hdfs/data</value>
    </property>
</configuration>
```

1-10. 하둡 실행

hadoop01에서 다음 명령어 실행

```
hadoop@hadoop01 ~/hadoop-2.7.3 $ bin/hdfs namenode -format
hadoop@hadoop01 ~/hadoop-2.7.3 $ sbin/start-dfs.sh
```

bin/hdfs namenode -format sbin/start-dfs.sh

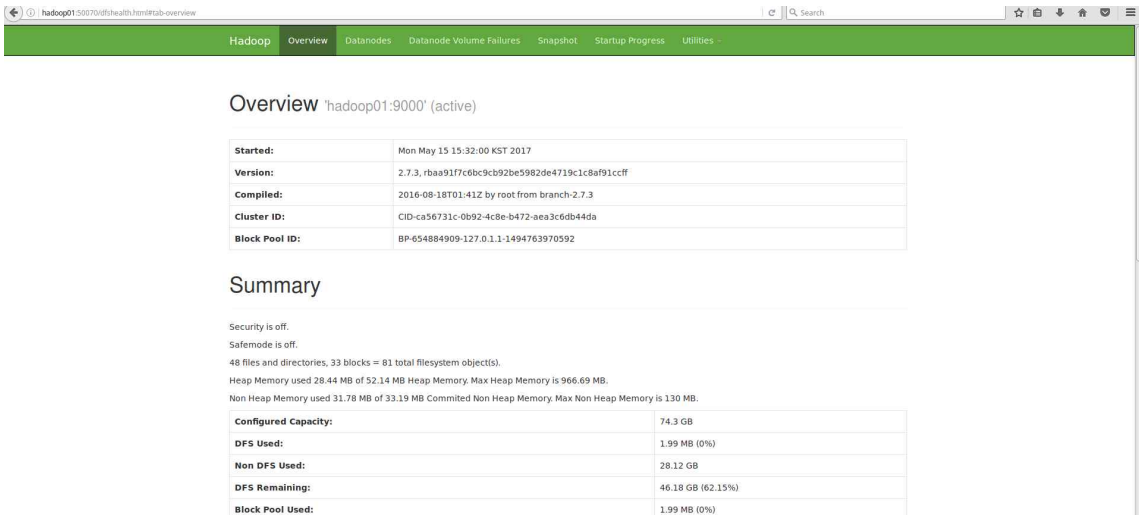
hadoop02에서 다음 명령어 실행

```
hadoop@hadoop02 ~/hadoop-2.7.3 $ sbin/start-yarn.sh
```

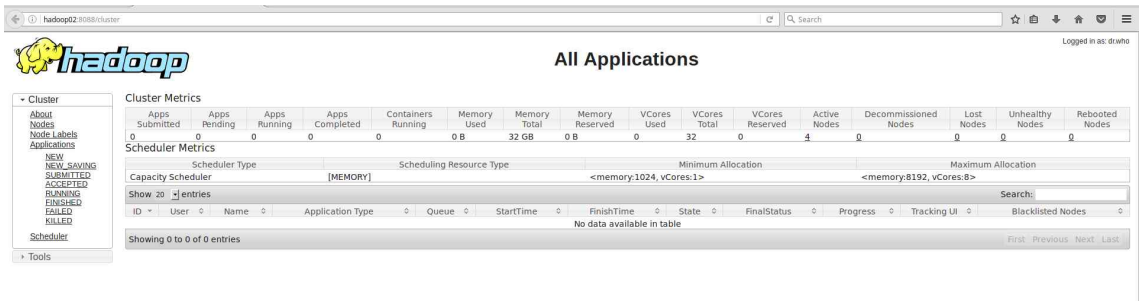
sbin/start-yarn.sh

jps 확인

hadoop01:50070 확인



hadoop02:8088 확인



2. WordCount

```
hadoop@hadoop01:~/hadoop-2.7.3$ bin/hdfs dfs -mkdir /user
hadoop@hadoop01:~/hadoop-2.7.3$ bin/hdfs dfs -mkdir /user/hadoop
hadoop@hadoop01:~/hadoop-2.7.3$ bin/hdfs dfs -put etc/hadoop input
hadoop@hadoop01:~/hadoop-2.7.3$ bin/hadoop jar share/hadoop/mapreduce/hadoop-map
reduce-examples-2.7.3.jar grep input output 'dfs[a-z.]+'
```

```
bin/hdfs dfs -mkdir /user
bin/hdfs dfs -mkdir /user/hadoop
bin/hdfs dfs -put etc/hadoop input
bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.3.jar
grep input output 'dfs[a-z.]+'
```

output파일 확인

```
hadoop@hadoop01:~/hadoop-2.7.3$ bin/hdfs dfs -ls /user/hadoop
Found 2 items
drwxr-xr-x   - hadoop supergroup          0 2017-05-16 15:50 /user/hadoop/input
drwxr-xr-x   - hadoop supergroup          0 2017-05-16 15:53 /user/hadoop/output
```

```
bin/hdfs dfs -ls /user/hadoop
```

결과 확인

```
hadoop@hadoop01:~/hadoop-2.7.3$ bin/hdfs dfs -cat output/*
6       dfs.audit.logger
4       dfs.class
3       dfs.server.namenode.
2       dfs.audit.log.maxbackupindex
2       dfs.period
2       dfs.audit.log.maxfilesize
1       dfsmetrics.log
1       dfsadmin
1       dfs.servers
1       dfs.replication
1       dfs.file
1       dfs.datanode.data.dir
1       dfs.namenode.name.dir
```

```
bin/hdfs dfs -cat output/*
```