

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

가상 머신 만들기

이름 및 운영 체제

새 가상 머신을 나타내는 이름을 입력하고 설치할 운영 체제를 선택하십시오. 입력한 이름은 VirtualBox에서 가상 머신을 식별하는데 사용됩니다.

이름(N):

종류(T): 64

버전(V):

메모리 크기

가상 머신에 할당할 메모리(RAM) 크기를 메가바이트 단위로 입력하십시오.

추천 메모리 크기는 1024 MB입니다.

4 MB 1024 MB 8192 MB

전문가 모드(E) 다음(N) 취소

다음(N) 취소

가상 하드 디스크 만들기

파일 위치 및 크기

새 가상 하드 디스크 파일의 이름을 아래 상자에 입력하거나 폴더 아이콘을 눌러서 파일을 생성할 폴더를 지정할 수 있습니다.

새 가상 하드 디스크 크기를 메가바이트 단위로 입력하십시오. 가상 머신에서 가상 하드 드라이브에 저장할 수 있는 데이터의 최대 크기입니다.

4.00 MB 20.00 GB 2.00 TB

만들기 취소

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

? X

← 가상 머신 선택

새 머신 이름

새 가상 머신의 이름을 입력하십시오. 새 머신은 머신 **hadoop01**의 복제본이 될 것입니다.

hadoop02

☒ 모든 네트워크 카드의 MAC 주소 초기화(R)

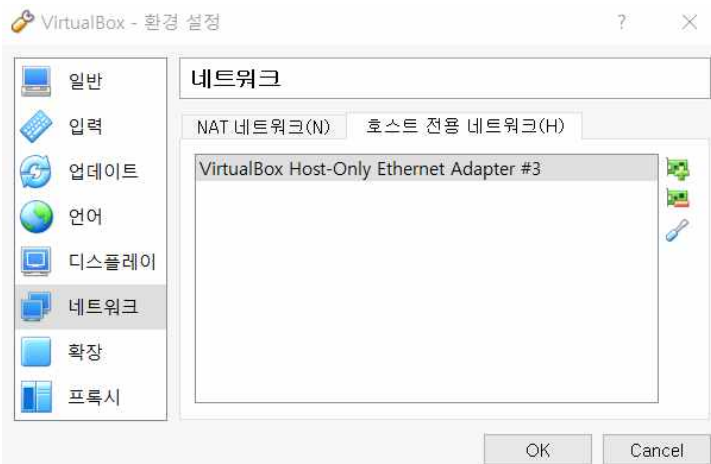
전문가 모드(E)

다음(N)

취소

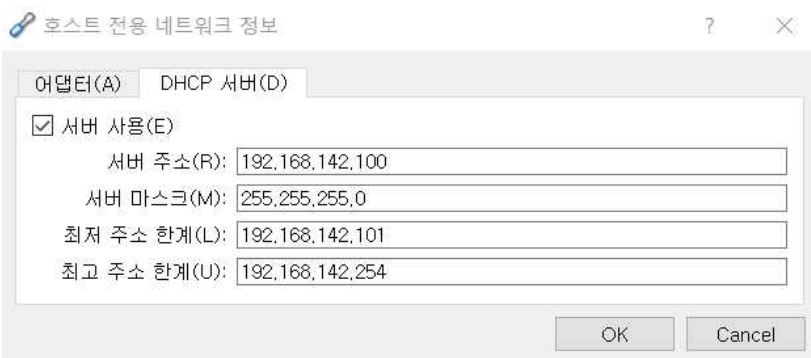
1-4. 네트워크 설정

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



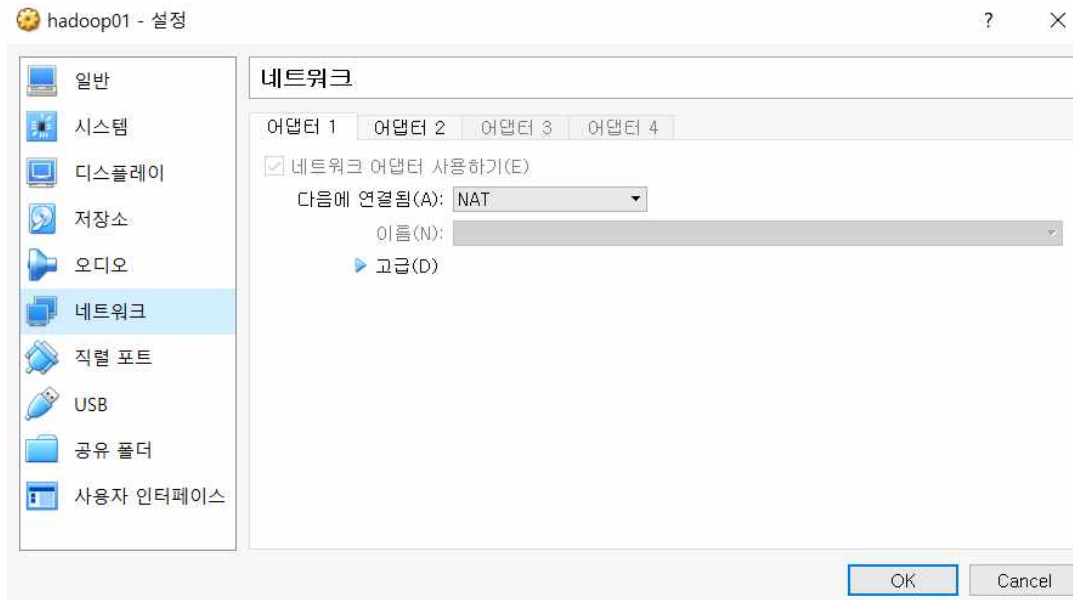
편집(드라이버모양) -> DHCP 서버 선택

아래와 같이 설정



hadoop01 설정 -> 네트워크

어댑터 1은 NAT, 어댑터 2는 호스트 전용 어댑터로 설정



hadoop02, hadoop03, hadoop04 설정 -> 네트워크

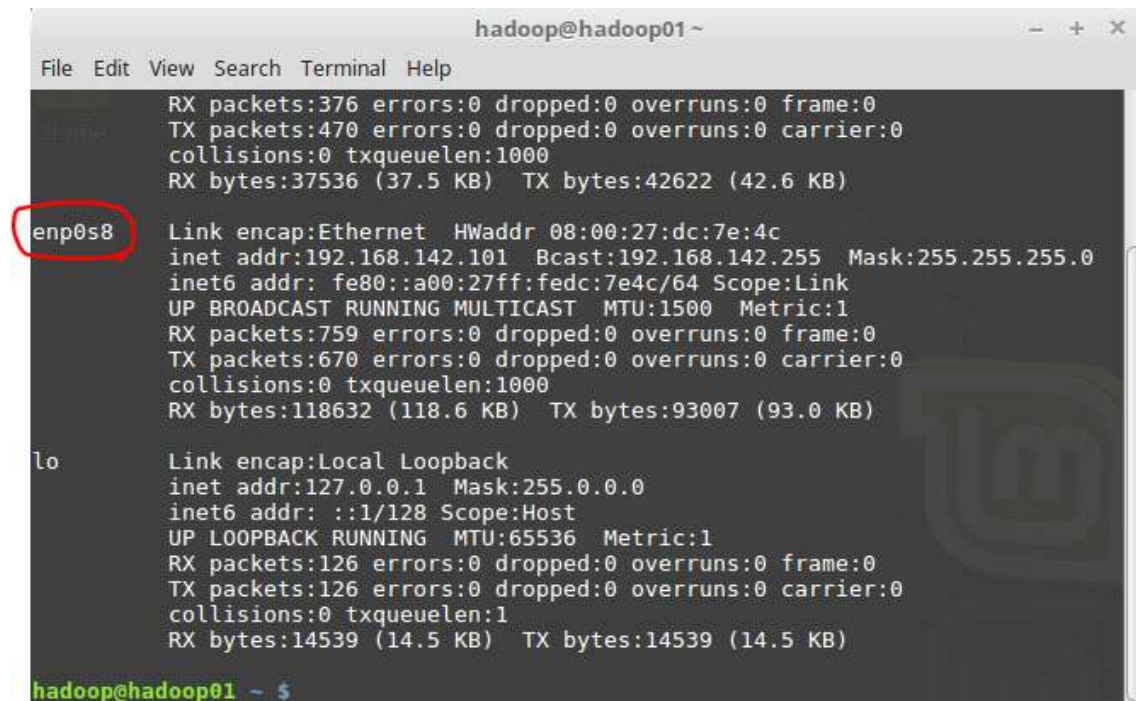
호스트 전용 어댑터만 설정



가상머신에 고정 IP 할당

sudo vi /etc/network/interfaces -> 고정 IP할당

hadoop01



```
hadoop@hadoop01 ~  
File Edit View Search Terminal Help  
1 interfaces(5) file used by ifup(8) and ifdown(8)  
2 auto lo  
3 iface lo inet loopback  
4  
5 auto enp0s8  
6 iface enp0s8 inet static  
7 address 192.168.142.101  
8 netmask 255.255.255.0
```

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

hadoop02, hadoop03, hadoop04

```
hadoop@hadoop02 ~ $ ifconfig  
enp0s3: Link encap:Ethernet HWaddr 08:00:27:51:96:31  
inet addr:192.168.142.102 Bcast:192.168.142.255 Mask:255.255.255.0  
inet6 addr: fe80::a00:27ff:fe51:9631/64 Scope:Link  
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
RX packets:405 errors:0 dropped:0 overruns:0 frame:0  
TX packets:287 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:48295 (48.2 KB) TX bytes:46765 (46.7 KB)  
  
lo: Link encap:Local Loopback  
inet addr:127.0.0.1 Mask:255.0.0.0  
inet6 addr: ::1/128 Scope:Host  
UP LOOPBACK RUNNING MTU:65536 Metric:1  
RX packets:3232 errors:0 dropped:0 overruns:0 frame:0  
TX packets:3232 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1  
RX bytes:257935 (257.9 KB) TX bytes:257935 (257.9 KB)  
  
hadoop@hadoop02 ~ $
```

```
hadoop@hadoop02 ~  
File Edit View Search Terminal Help  
1 interfaces(5) file used by ifup(8) and ifdown(8)  
2 auto lo  
3 iface lo inet loopback  
4  
5 auto enp0s3  
6 iface enp0s3 inet static  
7 address 192.168.142.102  
8 netmask 255.255.255.0
```

```
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
8
```

```
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/authorized_keys
```

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

```
hadoop@hadoop01 ~/.ssh $ scp id_rsa.pub hadoop@hadoop04:/home/hadoop/.ssh/authorized_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/hadoop-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
```

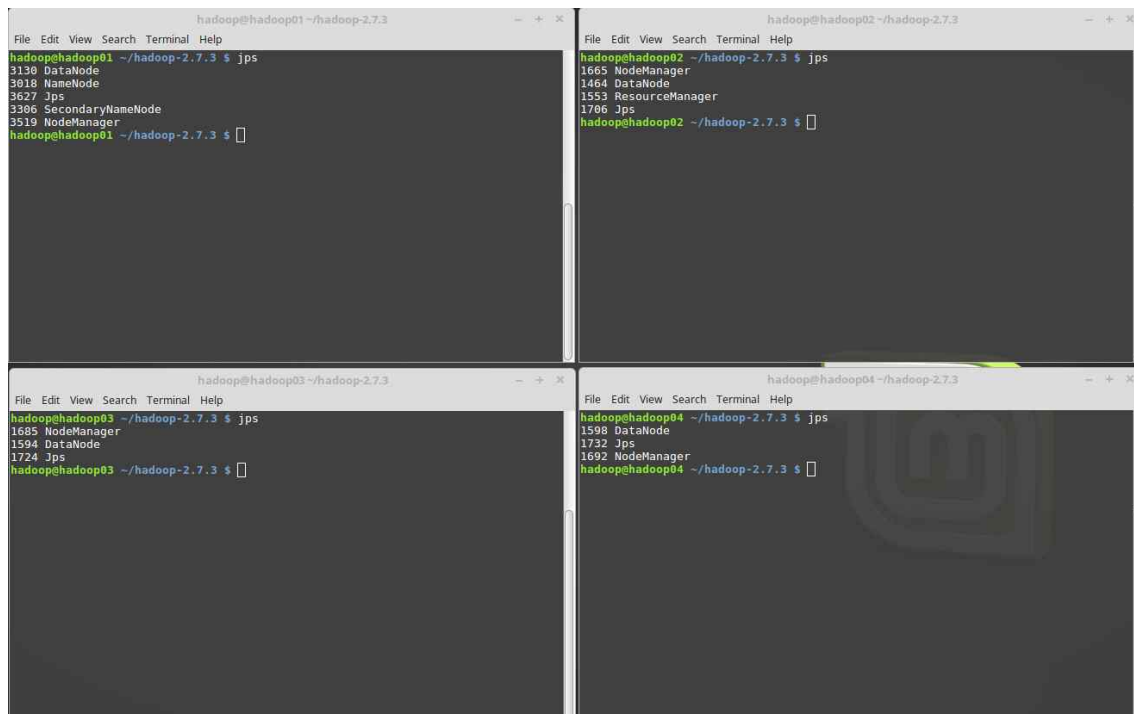
```
sbin/start-dfs.sh
```

hadoop02에서 다음 명령어 실행

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

```
sbin/start-yarn.sh
```

jps 확인



hadoop01:50070 확인

hadoop01:50070dfshealth.html#tab-overview

Overview

Datanodes

Datanode Volume Failures

Snapshot

Startup Progress

Utilities

Overview "hadoop01:9000" (active)

Started:

Mon May 15 15:32:00 KST 2017

Version:

2.7.3, rbaa91f7c6bc9cb2be5982de4719c1c8af91ccff

Compiled:

2016-08-18T01:41Z by root from branch-2.7.3

Cluster ID:

CID-ca56731c-0b92-4c8e-b472-aea3c6db44da

Block Pool ID:

BP-654884909-127.0.1.1-1494763970592

Summary

Security is off.
Safemode is off.
48 files and directories, 33 blocks = 81 total filesystem object(s).
Heap Memory used 28.44 MB of 52.14 MB Heap Memory. Max Heap Memory is 966.69 MB.
Non Heap Memory used 31.78 MB of 33.19 MB Committed Non Heap Memory. Max Non Heap Memory is 130 MB.

Configured Capacity:

74.3 GB

DFS Used:

1.99 MB (0%)

Non DFS Used:

28.12 GB

DFS Remaining:

46.18 GB (62.15%)

Block Pool Used:

1.99 MB (0%)

hadoop02:8088 확인

hadoop02:8088/cluster

hadoop

All Applications

Logged in as: drwho

Cluster

About

Nodes

Node Labels

Applications

NEW

NEW, SAVING

SUBMITTED

ACCEPTED

RUNNING

FINISHED

FAILED

KILLED

Scheduler

Tools

Cluster Metrics

Apps Submitted: 0

Apps Pending: 0

Apps Running: 0

Apps Completed: 0

Containers Running: 0

Memory Used: 0 B

Memory Total: 32 GB

Memory Reserved: 0 B

VCores Used: 0

VCores Total: 32

VCores Reserved: 0

Active Nodes: 4

Decommissioned Nodes: 0

Lost Nodes: 0

Unhealthy Nodes: 0

Rebooted Nodes: 0

Scheduler Metrics

Capacity Scheduler

Scheduler Type: [MEMORY]

Scheduling Resource Type: <memory:1024, vCores:1>

Minimum Allocation: <memory:8192, vCores:8>

Maximum Allocation: <memory:8192, vCores:8>

Show 30 entries

ID

User

Name

Application Type

Queue

StartTime

FinishTime

State

FinalStatus

Progress

Tracking UI

Blacklisted Nodes

No data available in table

Showing 0 to 0 of 0 entries

First

Previous

Next

Last

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-mapreduce-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/*
```