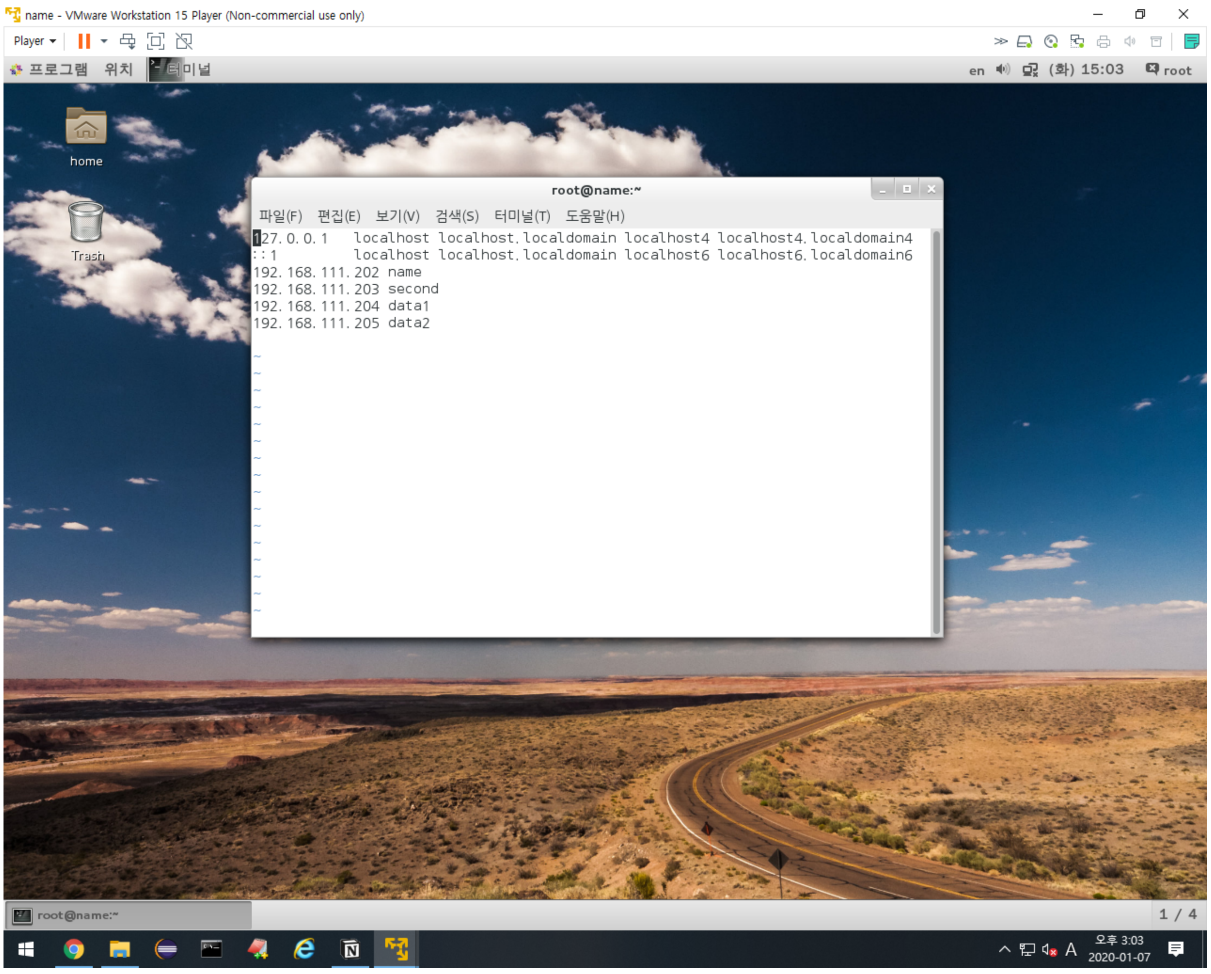
HDFS 구축 (완전분산모드)

1)_ name server setting (hadoopserver page참고)

hadoopserver역할을 할 name서버에는 vi/etc/hosts에 키값을 전달해 줄 서버 ip를 쓴다.


name - VMware Workstation 15 Player (Non-commercial use only)
root@name:~
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

```
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.111.202 name
192.168.111.203 second
192.168.111.204 data1
192.168.111.205 data2
```

2) 각 서버에 SSH 연결

name(hadoopserver1) , second, data1, data2 인 4개의 서버가 있다.

name서버에서 키 값을 공유하여 second,data1,data2에게 퍼블릭 키를 전달한다.

```
[root@name ~]# rm -rf .ssh
=> 원래는 ssh가 없으나 복사한 서버이기 때문에 지우고 시작
[root@name ~]# ssh name
The authenticity of host 'name (192.168.111.202)' can't be established.
ECDSA key fingerprint is e0:ae:a3:86:7f:e7:ff:c8:27:5d:c7:dd:12:da:ce:1a.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'name,192.168.111.202' (ECDSA) to the list of known hosts.
root@name's password:
=> 처음 실행엔 패스워드 입력
[root@name ~]# ssh-keygen -t dsa -P '' -f ~/.ssh/id_dsa
Generating public/private dsa key pair.
Your identification has been saved in /root/.ssh/id_dsa.
Your public key has been saved in /root/.ssh/id_dsa.pub.
The key fingerprint is:
9d:c3:e7:5c:73:32:32:76:5e:0c:32:28:33:38:a0:8b root@name
The key's randomart image is:
+--[ DSA 1024]----+
|    .            |
|    . . .    .      |
|  .    o + . o .  |
| . .    . * . o o |
|E .      S = = * +|
|            * * * |
|             o .  |
|                  |
|                  |
+-----------------+
[root@name ~]# cd .ssh
[root@name .ssh]# ssh-copy-id -i id_dsa.pub root@second
The authenticity of host 'second (192.168.111.203)' can't be established.
ECDSA key fingerprint is e0:ae:a3:86:7f:e7:ff:c8:27:5d:c7:dd:12:da:ce:1a.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@second's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'root@second'"
and check to make sure that only the key(s) you wanted were added.
=> 퍼블릭 키 값을 줄 서버에 모두 copy명령어 실행
[root@name .ssh]# ssh-copy-id -i id_dsa.pub root@data1
[root@name .ssh]# ssh-copy-id -i id_dsa.pub root@data2

[root@name .ssh]# ssh second
Last login: Tue Jan  7 13:47:55 2020
[root@second ~]# exit
logout
Connection to second closed.
[root@name .ssh]# ssh data2
Last login: Tue Jan  7 13:47:42 2020
[root@name ~]# ssh data1
Last login: Tue Jan  7 13:47:49 2020
=> 패스워드 입력하지 않고 ssh 접속되면 ok (recursive와 동일한 방식)
```

3) name 서버 : hadoop 및 JDK 세팅

scp 명령어를 사용하여 어디에 무엇을 넣겠다고 입력하기 (p.51)

```
[root@name ~]# scp /etc/hosts root@second:/etc/
hosts                                         100%  247     0.2KB/s   00:00
[root@name ~]# scp /etc/hosts root@data1:/etc/
hosts                                         100%  247     0.2KB/s   00:00
[root@name ~]# scp /etc/hosts root@data2:/etc/
hosts                                         100%  247     0.2KB/s   00:00
=> /etc/hosts파일을 root권한으로 second서버에 /etc/밑에 넣겠다.
```

p.59 vi편집기를이용하여 conf 밑 xml수정

```
# vi hadoop-env.sh
# The java implementation to use.  Required.
export JAVA_HOME=/usr/local/jdk1.8.0
export HADOOP_HOME_WARN_SUPPRESS="TRUE"
```

p.56

```
[root@name conf]# vi masters
에서 localhost -> second 로 바꾸기
/etc/hosts에 미리 세팅을 해 놓았기 때문에 ip 주소를 쓰지 않아도 된다.

[root@name conf]# vi slaves
data node를 실행할 서버 이름을 적기 : data1,data2,second(second에도 데이터노드를 설정하도록??)

[root@name conf]# vi core-site.xml
client가 들어갈 때 구멍이 9000번이고,  data가 들어갈 때도 9000번이다.
현재 default가 9000번이지만, 다른 곳에서 사용중이라면 포트 번호를 바꿔주거나, 방화벽도 점검해본다.
이 파일에서 서버이름을 name으로 변경한다. (내가 쓰는 서버 이름)

[root@name conf]# vi hdfs-site.xml
<configuration>

<property>
  <name>dfs.replication</name>
  <value>3</value>
</property>
```

```
<property>
  <name>dfs.name.dir</name>
  <value>/usr/local/hadoop-1.2.1/name</value>
</property>
-> name안에 네임노드의 메타정보가 들어간다.
<property>
  <name>dfs.data.dir</name>
  <value>/usr/local/hadoop-1.2.1/data</value>
</property>
-> 실제로 hdfs에 쓰일 데이터가 data에 들어간다.
</configuration>
[root@name conf]# vi hdfs-site.xml
<configuration>
<property>
<name>mapred.job.tracker</name>
<value>name:9001</value>
=>client가 분석요청시에는 namenode가 아닌 jobtracker에게 요청하는데, 그 때 쓰이는 포트번호가 9001이다.
9000번은 namenode에게 요청하는 포트 번호 이다.
=> 보안이 강하면fire-wall로 접속하여 해결한다.
</property>
</configuration>

------------------------세팅완료 ----------------------------
저번에 가상으로 만들었던 name, data, tmp파일을 삭제한다.
[root@name hadoop-1.2.1]# rm -rf data
[root@name hadoop-1.2.1]# rm -rf name
[root@name hadoop-1.2.1]# rm -rf tmp
(정상적으로만들기위해)
```

- dfs.http.address

  : 디폴트가 50070

- dfs.secondary.http.address

  : 디폴트가 50090

  ```
  [root@name local]# ls
  apache-tomcat-9.0.22  etc           include   lib64    share
  bin                   games         jdk1.8.0  libexec  src
  eclipse               hadoop-1.2.1  lib       sbin
  [root@name local]# vi /etc/profile
  JAVA_HOME=/usr/local/jdk1.8.0
  CLASSPATH=$JAVA_HOME/lib
  HADOOP_HOME=/usr/local/hadoop-1.2.1
  PATH=.:$JAVA_HOME/bin:$HADOOP_HOME/bin:$PATH
  export JAVA_HOME CLASSPATH HADOOP_HOME
  export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE HISTCONTROL
  ```

4) hadoop과 JDK를 압축하기

```
[root@name local]# tar cvfJ jdk1.8.0.tar.gz jdk1.8.0/
[root@name local]# tar cvfz hadoop-1.2.1.tar.gz hadoop-1.2.1/

=> 형식은 상관없음
```

5) /etc/profile, hadoop, jdk를 각 시스템에 전송하기

```
[root@name local]# scp /etc/profile root@second:/etc
[root@name local]# scp /etc/profile root@data1:/etc
[root@name local]# scp /etc/profile root@data2:/etc
=> root권한으로 /etc/profile을 second, data1, data2 서버의 /etc 밑에 전송

[root@name local]# scp hadoop-1.2.1.tar.gz root@second:/usr/local
[root@name local]# scp hadoop-1.2.1.tar.gz root@data1:/usr/local
[root@name local]# scp hadoop-1.2.1.tar.gz root@data2:/usr/local
=> root권한으로 hadoop-1.2.1.tar.gz 압축파일을 /usr/local 밑에 second, data1, data2서버에 복사

[root@name local]# scp jdk1.8.0.tar.gz root@data2:/usr/local
```

```
[root@name local]# scp jdk1.8.0.tar.gz root@data1:/usr/local
[root@name local]# scp jdk1.8.0.tar.gz root@second:/usr/local
=> 위와 마찬가지로 jdk 압축파일을 각 서버에 복사한다.
```

6) hadoop, jdk 압축을 해제하기

수백대의 컴퓨터가 있다면 압축을 하나하나 풀 수 없다.

```
[root@name local]# ssh root@second "cd /usr/local; tar xvf jdk1.8.0.tar.gz; rm -rf jdk.1.8.0.tar.gz"
-> ssh로 second 서버에 접속
[root@name local]# ssh root@second "cd /usr/local; tar xvf hadoop-1.2.1.tar.gz; rm -rf hadoop-1.2.1.tar.gz"
[root@name local]# ssh root@data1 "cd /usr/local; tar xvf hadoop-1.2.1.tar.gz; rm -rf hadoop-1.2.1.tar.gz"
[root@name local]# ssh root@data2 "cd /usr/local; tar xvf hadoop-1.2.1.tar.gz; rm -rf hadoop-1.2.1.tar.gz"


[root@name local]# ssh root@second "cd /usr/local; tar xvf jdk1.8.0.tar.gz; rm -rf jdk 1.8.0.tar.gz"
[root@name local]# ssh root@data1 "cd /usr/local; tar xvf jdk1.8.0.tar.gz; rm -rf jdk 1.8.0.tar.gz"
[root@name local]# ssh root@data2 "cd /usr/local; tar xvf jdk1.8.0.tar.gz; rm -rf jdk 1.8.0.tar.gz"

=> ssh 명령어를 이용하여 원격서버에서 명령어를 실행한다.
=> jdk, hadoop의 압축을 풀고 압축파일을 삭제한다. (각 서버에서)
=> ssh 접속계정@접속 호스트명"명령어" ( 세미콜론;을 구분자로 여러개를 실행 가능)
```

///////////////////////////////완전분산모드 완료///////////////////////////////

7) 포맷 후 jps로 확인

```
1. name 서버
[root@name hadoop-1.2.1]# hadoop namenode -format
20/01/07 17:08:42 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = name/192.168.111.202
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 1.2.1
STARTUP_MSG:   build = https://svn.apache.org/repos/asf/hadoop/common/branches/branch-1.2 -r 1503152; compiled by 'mattf' on Mo
STARTUP_MSG:   java = 1.8.0_221
************************************************************/
20/01/07 17:08:43 INFO util.GSet: Computing capacity for map BlocksMap
20/01/07 17:08:43 INFO util.GSet: VM type       = 64-bit
20/01/07 17:08:43 INFO util.GSet: 2.0% max memory = 932184064
20/01/07 17:08:43 INFO util.GSet: capacity      = 2^21 = 2097152 entries
20/01/07 17:08:43 INFO util.GSet: recommended=2097152, actual=2097152
20/01/07 17:08:43 INFO namenode.FSNamesystem: fsOwner=root
20/01/07 17:08:43 INFO namenode.FSNamesystem: supergroup=supergroup
20/01/07 17:08:43 INFO namenode.FSNamesystem: isPermissionEnabled=true
20/01/07 17:08:43 INFO namenode.FSNamesystem: dfs.block.invalidate.limit=100
20/01/07 17:08:43 INFO namenode.FSNamesystem: isAccessTokenEnabled=false accessKeyUpdateInterval=0 min(s), accessTokenLifetime=
20/01/07 17:08:43 INFO namenode.FSEditLog: dfs.namenode.edits.toleration.length = 0
20/01/07 17:08:43 INFO namenode.NameNode: Caching file names occuring more than 10 times
20/01/07 17:08:43 INFO common.Storage: Image file /usr/local/hadoop-1.2.1/name/current/fsimage of size 110 bytes saved in 0 sec
20/01/07 17:08:44 INFO namenode.FSEditLog: closing edit log: position=4, editlog=/usr/local/hadoop-1.2.1/name/current/edits
20/01/07 17:08:44 INFO namenode.FSEditLog: close success: truncate to 4, editlog=/usr/local/hadoop-1.2.1/name/current/edits
20/01/07 17:08:44 INFO common.Storage: Storage directory /usr/local/hadoop-1.2.1/name has been successfully formatted.
20/01/07 17:08:44 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at name/192.168.111.202
************************************************************/
[root@name hadoop-1.2.1]# start-all.sh
starting namenode, logging to /usr/local/hadoop-1.2.1/libexec/../logs/hadoop-root-namenode-name.out
second: starting datanode, logging to /usr/local/hadoop-1.2.1/libexec/../logs/hadoop-root-datanode-second.out
data1: starting datanode, logging to /usr/local/hadoop-1.2.1/libexec/../logs/hadoop-root-datanode-data1.out
data2: starting datanode, logging to /usr/local/hadoop-1.2.1/libexec/../logs/hadoop-root-datanode-data2.out
second: starting secondarynamenode, logging to /usr/local/hadoop-1.2.1/libexec/../logs/hadoop-root-secondarynamenode-second.out
starting jobtracker, logging to /usr/local/hadoop-1.2.1/libexec/../logs/hadoop-root-jobtracker-name.out
data1: starting tasktracker, logging to /usr/local/hadoop-1.2.1/libexec/../logs/hadoop-root-tasktracker-data1.out
data2: starting tasktracker, logging to /usr/local/hadoop-1.2.1/libexec/../logs/hadoop-root-tasktracker-data2.out
second: starting tasktracker, logging to /usr/local/hadoop-1.2.1/libexec/../logs/hadoop-root-tasktracker-second.out
[root@name hadoop-1.2.1]# jps
2499 NameNode
2839 Jps
2700 JobTracker
[root@name hadoop-1.2.1]# ls
CHANGES.txt  conf                        hadoop-minicluster-1.2.1.jar  logs
LICENSE.txt  contrib                     hadoop-test-1.2.1.jar         name
NOTICE.txt   docs                        hadoop-tools-1.2.1.jar        sbin
```

```
README.txt   hadoop-ant-1.2.1.jar       ivy                             share
bin          hadoop-client-1.2.1.jar    ivy.xml                         src
build.xml    hadoop-core-1.2.1.jar      lib                             webapps
c++          hadoop-examples-1.2.1.jar  libexec
```

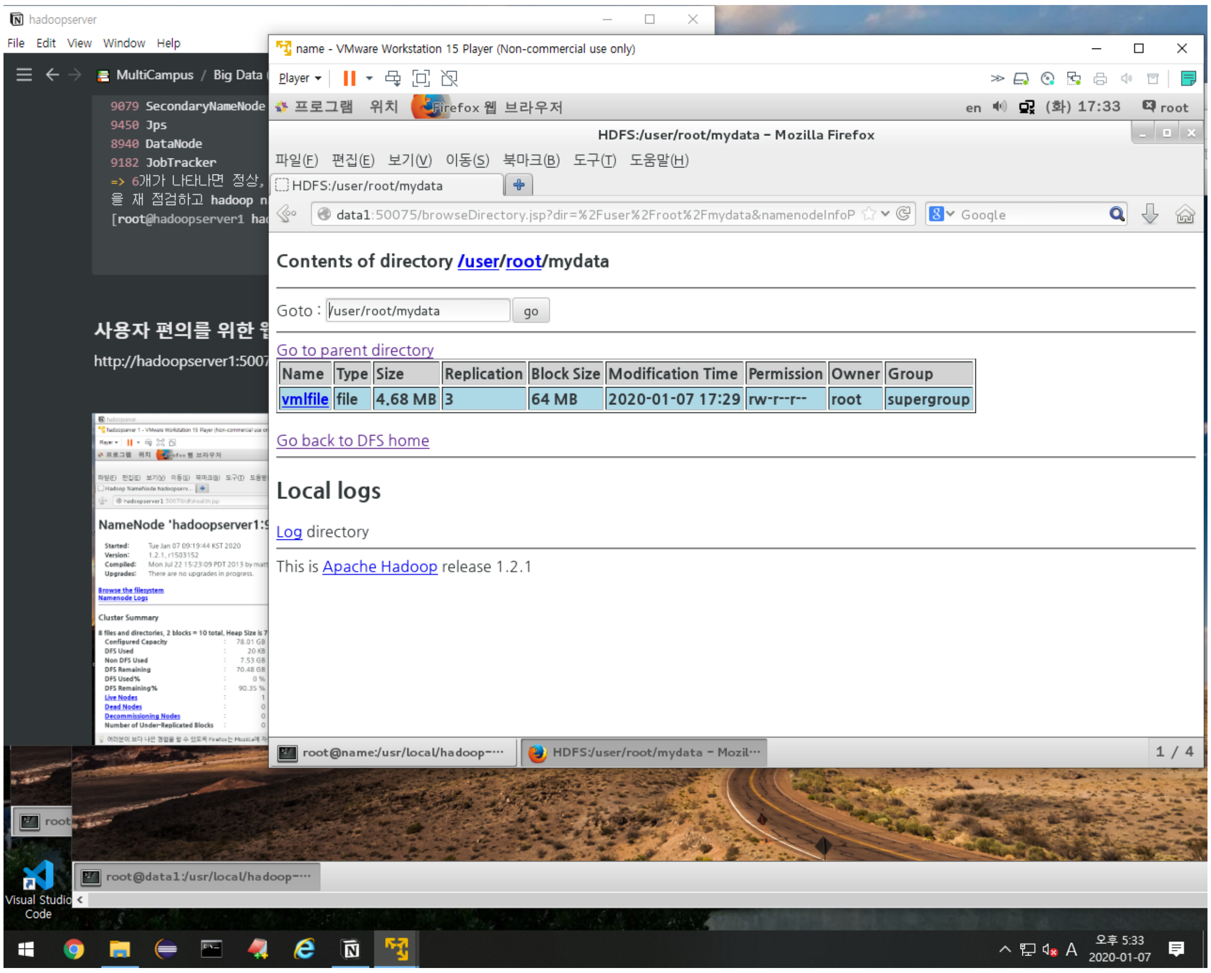
```
2. second 서버
[root@second hadoop-1.2.1]# ls
CHANGES.txt  conf                     hadoop-examples-1.2.1.jar     libexec
LICENSE.txt  contrib                  hadoop-minicluster-1.2.1.jar  logs
NOTICE.txt   data                     hadoop-test-1.2.1.jar         sbin
README.txt   docs                     hadoop-tools-1.2.1.jar        share
bin          hadoop-ant-1.2.1.jar     ivy                           src
build.xml    hadoop-client-1.2.1.jar  ivy.xml                       tmp
c++          hadoop-core-1.2.1.jar    lib                           webapps
[root@second hadoop-1.2.1]# jps
2768 Jps
2438 DataNode
2636 TaskTracker
2511 SecondaryNameNode
```

```
3. data1, data2 서버
[root@data1 hadoop-1.2.1]# ls
CHANGES.txt  conf                     hadoop-examples-1.2.1.jar     libexec
LICENSE.txt  contrib                  hadoop-minicluster-1.2.1.jar  logs
NOTICE.txt   data                     hadoop-test-1.2.1.jar         sbin
README.txt   docs                     hadoop-tools-1.2.1.jar        share
bin          hadoop-ant-1.2.1.jar     ivy                           src
build.xml    hadoop-client-1.2.1.jar  ivy.xml                       tmp
c++          hadoop-core-1.2.1.jar    lib

[root@data1 hadoop-1.2.1]# jps
2673 Jps
2455 DataNode
2554 TaskTracker
```

실습하기

1. wordcount 실행
2. /boot 에 있는 아무파일을 hadoop 시스템에 put또는 get하기(복사)
3. 모니터링 시스템을 통해 각 시스템의 상황을 모니터링 해보기 (50070)

```
[root@name hadoop-1.2.1]# hadoop fs -put /boot/vmlinuz-3.10.0-123.el7.x86_64  mydata/vmlfile
=> vml*파일을 vmlfile이라는 이름으로 mydata안에 저장한다.
[root@name hadoop-1.2.1]# hadoop jar hadoop-examples-1.2.1.jar wordcount mydata/vmlfile wordcount_output
wordcount 실행하기(글자수카운트)

3. 파이어폭스 들어가서 name:50070 접속하여 파일이 생겼는지 확인하고, 크기를 확인한다. (mydata파일크기가 늘어났는지, vmlfile이 있는지)
```



```
9079 SecondaryNameNode
9450 Jps
8940 DataNode
9182 JobTracker
=> 6개가 나타나면 정상,
을 재 점검하고 hadoop n
[root@hadoopserver1 ha
```

사용자 편의를 위한 웹

http://hadoopserver1:5007

NameNode 'hadoopserver1:

Started:	Tue Jan 07 09:19:44 KST 2020
Version:	1.2.1, r1503152
Compiled:	Mon Jul 22 15:23:09 PDT 2013 by matt
Upgrades:	There are no upgrades in progress.

Browse the filesystem
Namenode Logs

Cluster Summary

8 files and directories, 2 blocks = 10 total, Heap Size is 7

Configured Capacity	:	78.01 GB
DFS Used	:	20 KB
Non DFS Used	:	7.53 GB
DFS Remaining	:	70.48 GB
DFS Used%	:	0 %
DFS Remaining%	:	90.35 %
Live Nodes	:	1
Dead Nodes	:	0
Decommissioning Nodes	:	0
Number of Under-Replicated Blocks	:	0

HDFS:/user/root/mydata - Mozilla Firefox

data1:50075/browseDirectory.jsp?dir=%2Fuser%2Froot%2Fmydata&namenodeInfoP

Contents of directory /user/root/mydata

Goto : /user/root/mydata go

Go to parent directory

Name	Type	Size	Replication	Block Size	Modification Time	Permission	Owner	Group
vmlfile	file	4.68 MB	3	64 MB	2020-01-07 17:29	rw-r--r--	root	supergroup

Go back to DFS home

Local logs

Log directory

This is Apache Hadoop release 1.2.1

hadoopserver
File Edit View Window Help
MultiCampus / Big Data
name - VMware Workstation 15 Player (Non-commercial use only)
Player
프로그램 위치 Firefox 웹 브라우저
en (화) 17:33 root
파일(F) 편집(E) 보기(V) 이동(S) 북마크(B) 도구(T) 도움말(H)
HDFS:/user/root/mydata
Google
root@name:/usr/local/hadoop-…
HDFS:/user/root/mydata - Mozil…
1 / 4
root
root@data1:/usr/local/hadoop-…
Visual Studio Code
오후 5:33
2020-01-07