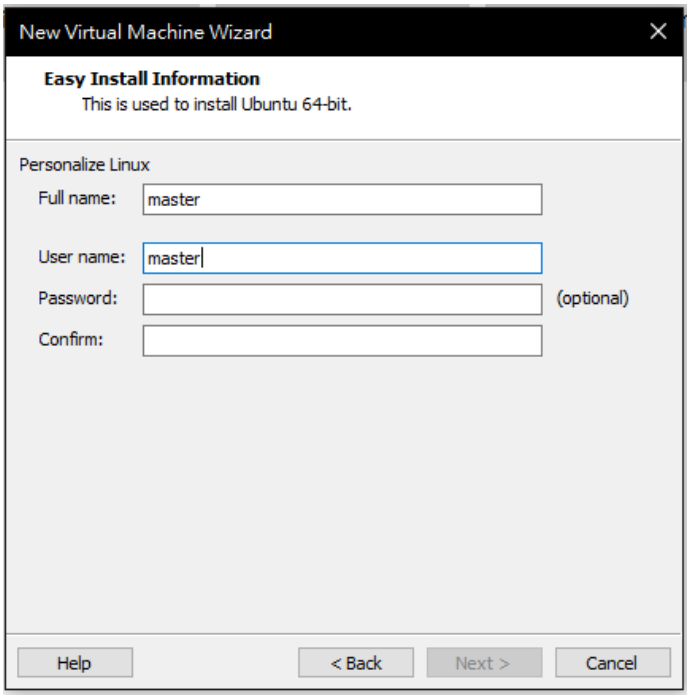
##
#Hadoop 單機版安裝
##

※ 如果 gedit 無法開啟文件，請重開 terminal 再試一次。

新增 master 主機：(設置如下)



1. 使用 Firefox 到官網下載(註冊帳號、登入、下載、存檔)
   https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html

Linux x86 Compressed Archive	186.1 MB	jdk-8u241-linux-i586.tar.gz
Linux x64 RPM Package	170.65 MB	jdk-8u241-linux-x64.rpm
Linux x64 Compressed Archive	185.53 MB	jdk-8u241-linux-x64.tar.gz
macOS x64	254.06 MB	jdk-8u241-macosx-x64.dmg

   下載至 Downloads 資料夾內

##
安裝 JDK
##

2. 開啟終端機

cd ~

cd Downloads

tar zxvf jdk-8u241-linux-x64.tar.gz

sudo mv jdk1.8.0_241 /opt/jdk

sudo gedit ~/.bashrc (配置環境變數)

將以下程式碼加在檔案最下面並且儲存(如果 gedit 無法開啟文件,請重開 terminal)

export CLASS_PATH="/opt/jdk/lib"

export JAVA_HOME="/opt/jdk"

export HADOOP_HOME="/opt/hadoop"

export MAVEN_HOME="/opt/maven"

export MAHOUT_HOME="/opt/mahout"

export

PATH=$PATH:$JAVA_HOME/bin:$HADOOP_HOME/bin:$HADOOP_HOME/sbin:$MAVEN_HOME/bin:$MAHOUT_HOME/bin

export HADOOP_CONF_DIR="/opt/hadoop/conf"

export HADOOP_MAPRED_HOME="/opt/hadoop"

export HADOOP_COMMON_HOME="/opt/hadoop"

export HADOOP_HDFS_HOME="/opt/hadoop"

export HADOOP_LOCAL="/opt/hadoop"

export YARN_HOME="/opt/hadoop"

3. source ~/.bashrc (執行環境變數設定)
4. java -version(檢驗是否成功)

```
master@master:~$ java -version
java version "1.8.0_241"
Java(TM) SE Runtime Environment (build 1.8.0_241-b07)
Java HotSpot(TM) 64-Bit Server VM (build 25.241-b07, mixed mode)
master@master:~$
```

##

安裝 ssh (因 Hadoop 用 ssh 進行通信)

##

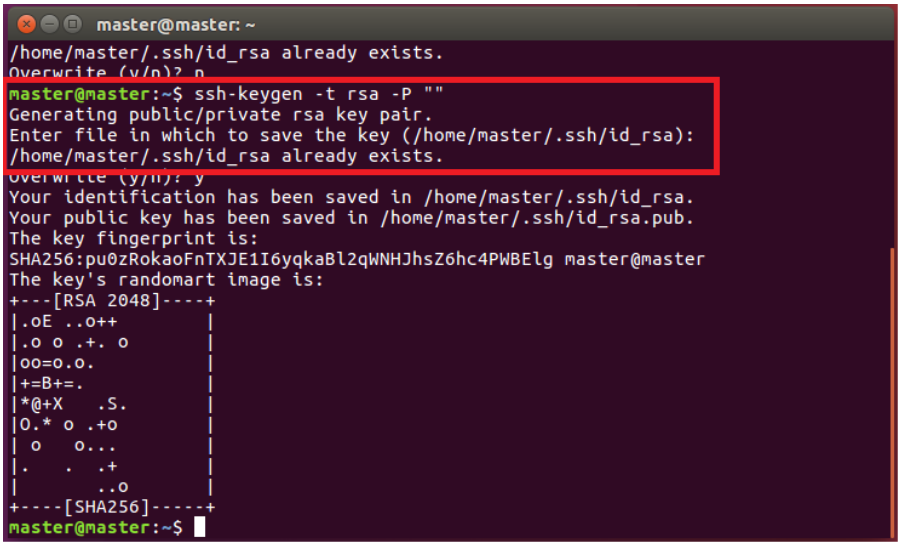
5. sudo apt-get install openssh-server (安裝 ssh)
6. sudo /etc/init.d/ssh start(啟動 ssh 服務)
7. ps -e | grep ssh(查看服務是否正確啟動)

```
master@master: ~
master@master:~$ sudo gedit ~/.bashrc
[sudo] password for master:

(gedit:2478): IBUS-WARNING **: The owner of /home/master/.config/ibus/bus is not
 root!

** (gedit:2478): WARNING **: Set document metadata failed: Setting attribute met
adata::gedit-position not supported
master@master:~$ 7.sudo /etc/init.d/ssh start
7.sudo: command not found
master@master:~$ sudo /etc/init.d/ssh start
[ ok ] Starting ssh (via systemctl): ssh.service.
master@master:~$ ps -e | grep ssh
   980 ?        00:00:00 sshd
master@master:~$
```

14. cd ~(回到根目錄)
15. ssh-keygen -t rsa -P "" (產生兩個檔：id_rsa 和 id_rsa.pub，前為私鑰，後為公鑰)

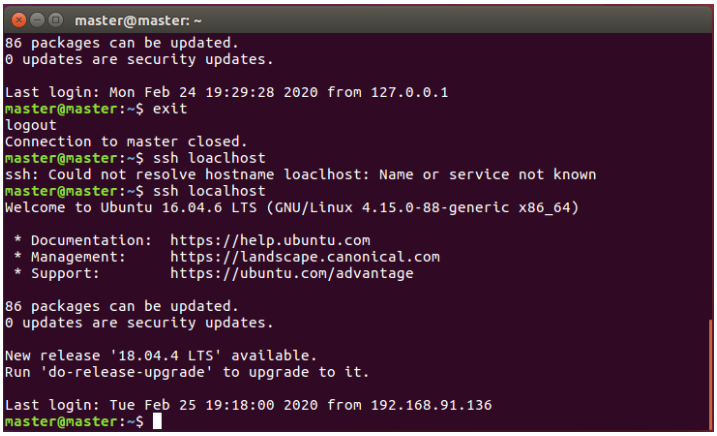


看到以上畫面直接按下 enter 即可

16. cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys(將公鑰追加到 authorized_keys 中)

17. ssh localhost(測試是否不用密碼登入了)

完成畫面：



exit(退出登入)

18. sudo gedit /etc/sysctl.conf(關閉 IPV6)

將以下程式碼加到該檔案最下面並且存檔

net.ipv6.conf.all.disable_ipv6 = 1

net.ipv6.conf.default.disable_ipv6 = 1

net.ipv6.conf.lo.disable_ipv6 = 1

##

安裝 hadoop 3.2.1

##

19. cd ~/Downloads

20. wget http://us.mirrors.quenda.co/apache/hadoop/common/hadoop-3.2.1/hadoop-3.2.1.tar.gz

21. tar zxvf hadoop-3.2.1.tar.gz (解壓縮)

22. sudo mv hadoop-3.2.1 /opt/hadoop (移到/opt 目錄下)

23. sudo gedit /opt/hadoop/etc/hadoop/hadoop-env.sh (設定 hadoop-env.sh)

將以下程式碼加在檔案最前面

export JAVA_HOME="/opt/jdk"

export HADOOP_HOME=/opt/hadoop

export HADOOP_MAPRED_HOME=$HADOOP_HOME

export HADOOP_COMMON_HOME=$HADOOP_HOME

export HADOOP_HDFS_HOME=$HADOOP_HOME

export YARN_HOME=$HADOOP_HOME

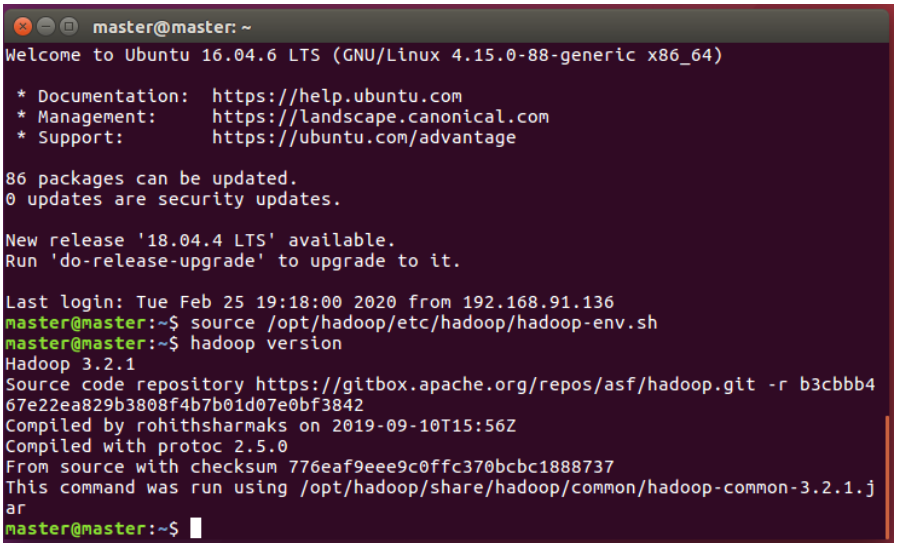
export PATH=$PATH:$HADOOP_HOME/bin

export PATH=$PATH:$HADOOP_HOME/sbin

export HADOOP_CONF_DIR="/opt/hadoop/etc/hadoop"

24. source /opt/hadoop/etc/hadoop/hadoop-env.sh(讓配置生效)

25. hadoop version(測試安裝正常)



26.造 datanode 目錄：

cd ~

mkdir hdfs

cd hdfs

mkdir name

mkdir data

27. 編輯 /opt/hadoop/etc/hadoop/hdfs-site.xml：

sudo gedit /opt/hadoop/etc/hadoop/hdfs-site.xml

#新增<configuration>裡面內容，其中/home/master 要改成/home/xxx，當初設定的使用者 xxx

```
<configuration>
  <property>
    <name>dfs.replication</name>
    <value>1</value>
  </property>
  <property>
    <name>dfs.namenode.name.dir</name>
    <value>file:/home/master/hdfs/name</value>
  </property>
  <property>
    <name>dfs.datanode.data.dir</name>
    <value>file:/home/master/hdfs/data</value>
```

```
  </property>
</configuration>
************************************************
```

28. 編輯/opt/hadoop/etc/hadoop/core-site.xml，新增<configuration>裡面內容，其中 xxx 為使用者名稱

```
sudo gedit /opt/hadoop/etc/hadoop/core-site.xml

<configuration>
    <property>
        <name>fs.defaultFS</name>
        <value>hdfs://localhost:9000</value>
    </property>
    <property>
        <name>hadoop.tmp.dir</name>
        <value>file:/home/master/tmp</value>
        <description>Abase for other temporary directories.</description>
    </property>
</configuration>
*********************************************
```

29. 編輯 /opt/hadoop/etc/hadoop/mapred-site.xml，新增<configuration>裡面內容

```
sudo gedit /opt/hadoop/etc/hadoop/mapred-site.xml

<configuration>
  <property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
  </property>
  <property>
    <name>yarn.app.mapreduce.am.env</name>
    <value>HADOOP_MAPRED_HOME=/opt/hadoop</value>
  </property>
  <property>
    <name>mapreduce.map.env</name>
    <value>HADOOP_MAPRED_HOME=/opt/hadoop</value>
  </property>
  <property>
    <name>mapreduce.reduce.env</name>
    <value>HADOOP_MAPRED_HOME=/opt/hadoop</value>
  </property>
</configuration>
*********************************************
```

30. 編輯/opt/hadoop/etc/hadoop/yarn-site.xml，新增<configuration>裡面內容

```
sudo gedit /opt/hadoop/etc/hadoop/yarn-site.xml
```

```
<configuration>
   <property>
      <name>yarn.nodemanager.aux-services</name>
      <value>mapreduce_shuffle</value>
   </property>
   <property>
      <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>
      <value>org.apache.hadoop.mapred.ShuffleHandler</value>
   </property>
</configuration>
```

****

31. 格式化 HDFS，namenode,secondarynamenode,tasktracker：

source /opt/hadoop/etc/hadoop/hadoop-env.sh

hdfs namenode -format

(每次 namenodeformat 會重新創建一個 namenodeId，而/hdfs/data 下包含上次 format 下的 id，namenode format 清空 namenode 下的資料，但沒有清空 datanode 下的資料，導致啟動時失敗，所要做的是每次 fotmat 前，清空 hdfs 下的所有內容。)

32. 安裝完成，啟動服務

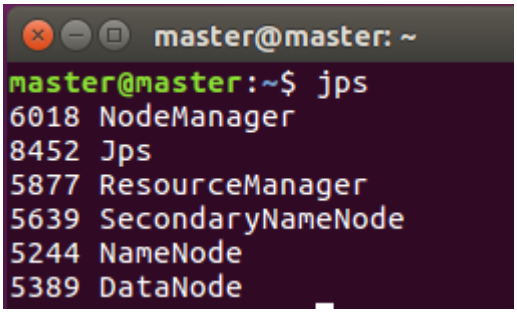
source /opt/hadoop/etc/hadoop/hadoop-env.sh

start-all.sh

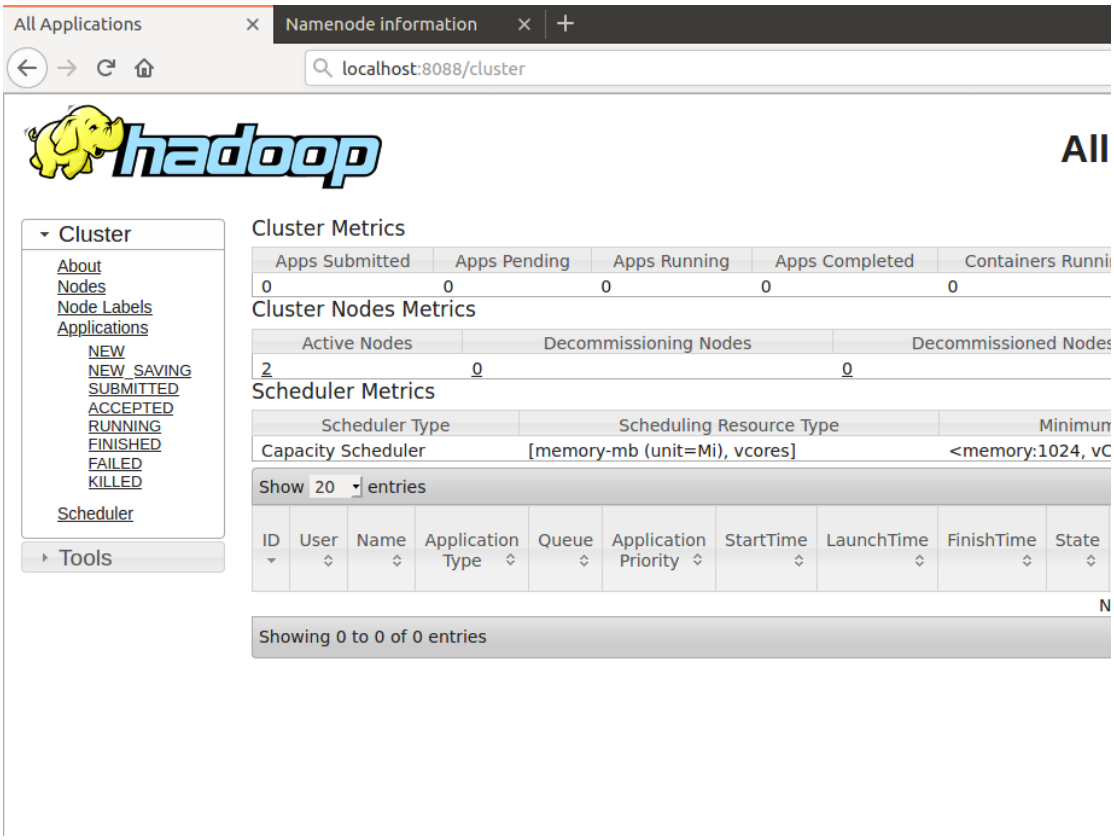
33. 驗證是否安裝成功 (Java Virtual Machine Process Status Tool)：

jps

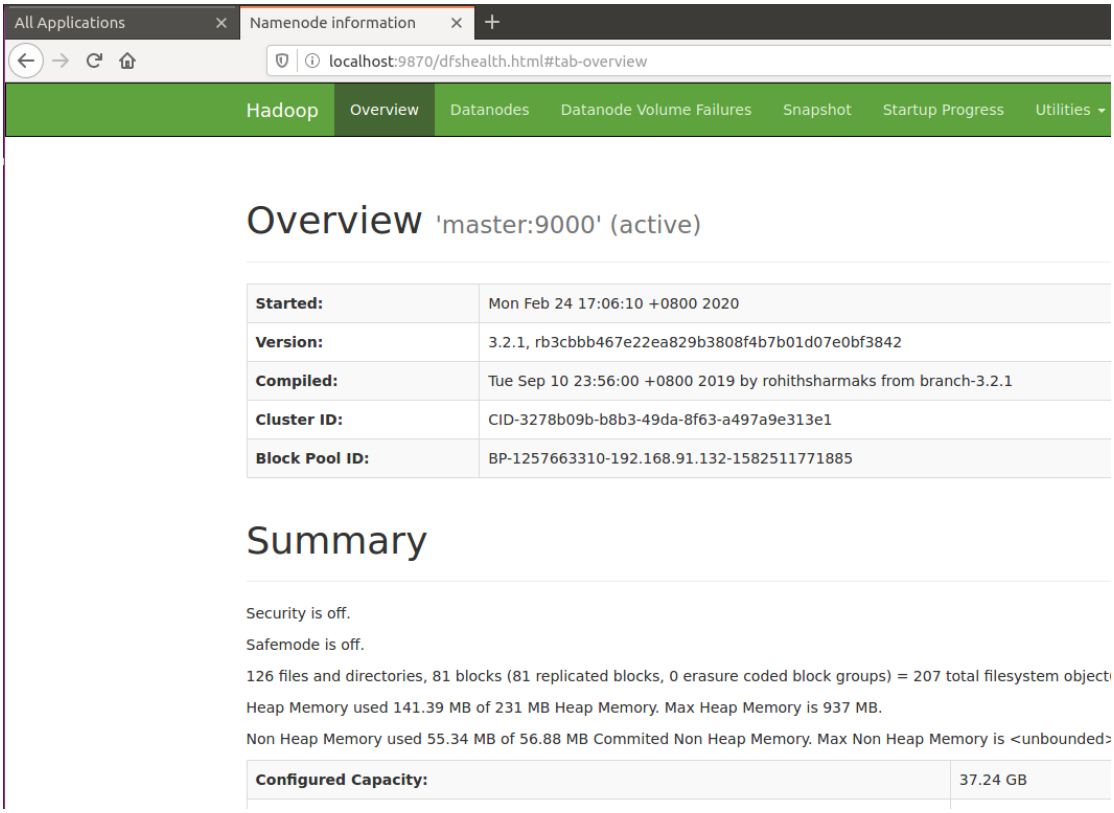
完成畫面：(一定要跑出 6 個東西)



34. 試試以下 2 個網址有無正常運作，http://localhost:8088



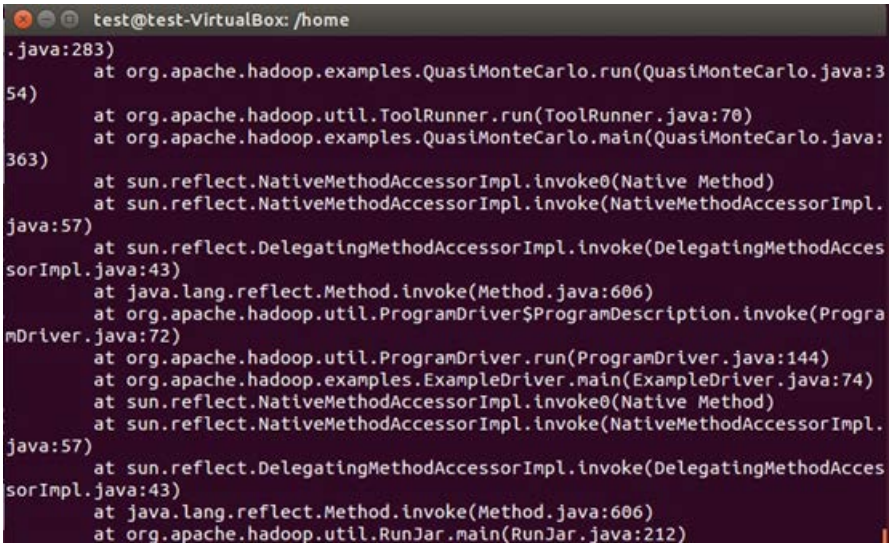
http://localhost:9870



35. 測試範例 I - 計算 PI

hadoop jar /opt/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.2.1.jar pi 2 5

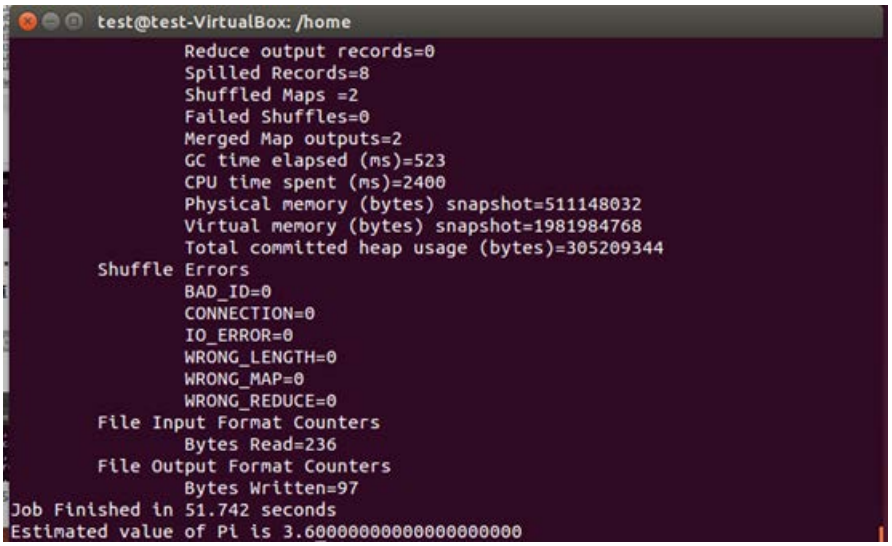
如果出現以下畫面,請輸入以下 2 個指令



source /opt/hadoop/etc/hadoop/hadoop-env.sh

start-all.sh

然後再試一次，成功畫面:



36. 測試範例 II - 計算字數

source /opt/hadoop/etc/hadoop/hadoop-env.sh

cd ~

cd hdfs

gedit word.txt

輸入一串英文字，例如 This is a book that is a cat book is ok

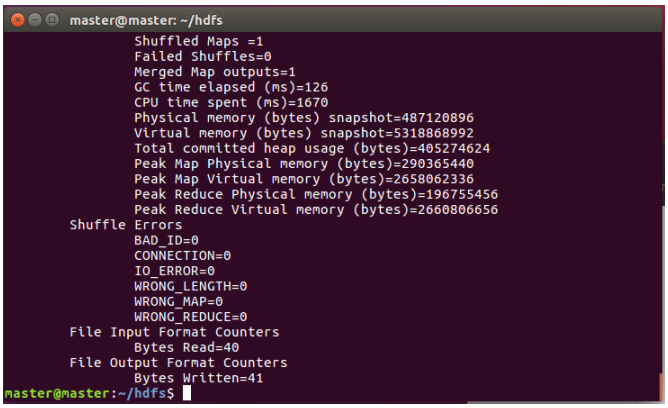
hadoop fs -put word.txt wordtest

hadoop fs -ls

```
master@master: ~/hdfs
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=236
        File Output Format Counters
                Bytes Written=97
Job Finished in 49.871 seconds
2020-02-25 19:29:38,641 INFO sasl.SaslDataTransferClient: SASL encryption trust
check: localHostTrusted = false, remoteHostTrusted = false
Estimated value of Pi is 3.60000000000000000000
master@master:~/hdfs$ hadoop fs -put word.txt wordtest
2020-02-25 19:29:50,150 INFO sasl.SaslDataTransferClient: SASL encryption trust
check: localHostTrusted = false, remoteHostTrusted = false
master@master:~/hdfs$ hadoop fs -put word.txt wordtest
put: `wordtest': File exists
master@master:~/hdfs$ hadoop fs -ls
Found 1 items
-rw-r--r--   2 master supergroup         40 2020-02-25 19:29 wordtest
master@master:~/hdfs$
```

以下是一整個指令,不要分開

hadoop jar /opt/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.2.1.jar wordcount wordtest wordoutput



接著我們可以查看成果

hadoop fs -cat wordoutput/*

```
master@master: ~/hdfs
                Peak Reduce Physical memory (bytes)=196755456
                Peak Reduce Virtual memory (bytes)=2660806656
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=40
        File Output Format Counters
                Bytes Written=41
master@master:~/hdfs$ hadoop fs -cat wordoutput/*
2020-02-25 19:32:54,015 INFO sasl.SaslDataTransferClient: SASL encryption trust
check: localHostTrusted = false, remoteHostTrusted = false
This    1
a       2
book    2
cat     1
is      3
ok      1
that    1
master@master:~/hdfs$
```

##
安裝 MAVEN
##
1. 下載並解壓縮 apache-maven-3.6.3-bin.tar.gz，移到/opt 下
cd ~
cd Downloads
wget http://ftp.mirror.tw/pub/apache/maven/maven-3/3.6.3/binaries/apache-maven-3.6.3-bin.tar.gz
tar zxvf apache-maven-3.6.3-bin.tar.gz
sudo mv apache-maven-3.6.3 /opt/maven
2. 測試是否安裝成功
mvn -version

```
master@master:~/hdfs$ mvn -version
Apache Maven 3.6.3 (cecedd343002696d0abb50b32b541b8a6ba2883f)
Maven home: /opt/maven
Java version: 1.8.0_241, vendor: Oracle Corporation, runtime: /opt/jdk/jre
Default locale: en_US, platform encoding: UTF-8
OS name: "linux", version: "4.15.0-88-generic", arch: "amd64", family: "unix"
```

##
安裝 MAHOUT
##
1. 下載並解壓縮 apache-mahout-distribution-0.12.2.tar.gz，移到/opt 下
cd ~
cd Downloads
wget https://downloads.apache.org/mahout/0.12.2/apache-mahout-distribution-0.12.2.tar.gz
tar zxvf apache-mahout-distribution-0.12.2.tar.gz
sudo mv apache-mahout-distribution-0.12.2 /opt/mahout
2. 測試是否安裝成功
mahout
#出現各種可用的演算法

```
master@master: ~/hdfs
  seq2sparse: : Sparse Vector generation from Text sequence files
  seqdirectory: : Generate sequence files (of Text) from a directory
  seqdumper: : Generic Sequence File dumper
  seqmailarchives: : Creates SequenceFile from a directory containing gzipped ma
il archives
  seqwiki: : Wikipedia xml dump to sequence file
  spectralkmeans: : Spectral k-means clustering
  split: : Split Input data into test and train sets
  splitDataset: : split a rating dataset into training and probe parts
  ssvd: : Stochastic SVD
  streamingkmeans: : Streaming k-means clustering
  svd: : Lanczos Singular Value Decomposition
  testnb: : Test the Vector-based Bayes classifier
  trainAdaptiveLogistic: : Train an AdaptivelogisticRegression model
  trainlogistic: : Train a logistic regression using stochastic gradient descent
  trainnb: : Train the Vector-based Bayes classifier
  transpose: : Take the transpose of a matrix
  validateAdaptiveLogistic: : Validate an AdaptivelogisticRegression model again
st hold-out data set
  vecdist: : Compute the distances between a set of Vectors (or Cluster or Canop
y, they must fit in memory) and a list of Vectors
  vectordump: : Dump vectors from a sequence file to text
  viterbi: : Viterbi decoding of hidden states from given output states sequence
master@master:~/hdfs$
```

3. 測試 K-means 演算法
用 FireFox 下載資料檔：http://archive.ics.uci.edu/ml/databases/synthetic_control/synthetic_control.data
下載至 Downloads
cd ~
cd hdfs
cp ~/Downloads/*.data .

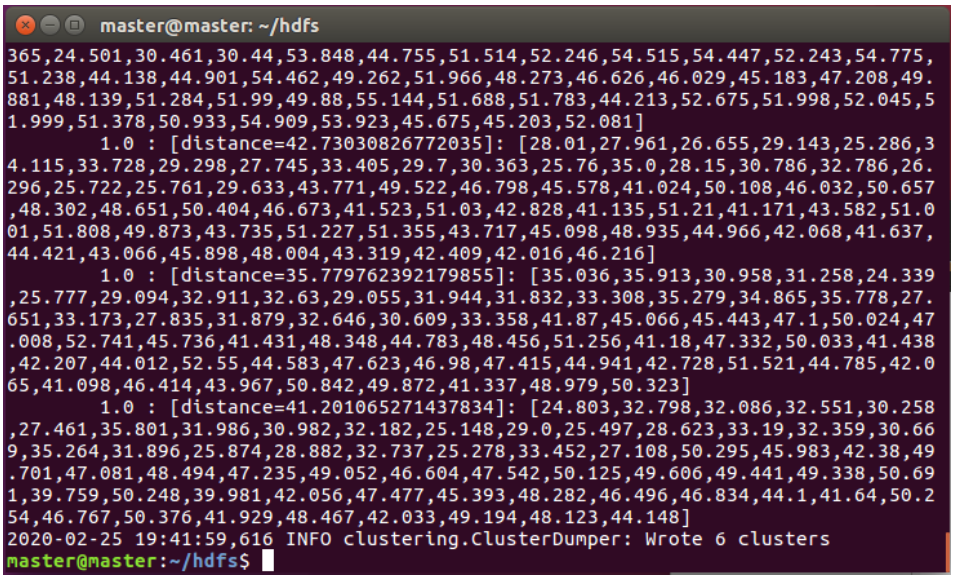
hadoop fs -mkdir testdata #

hadoop fs -put synthetic_control.data testdata

以下是一整個指令,不要分開

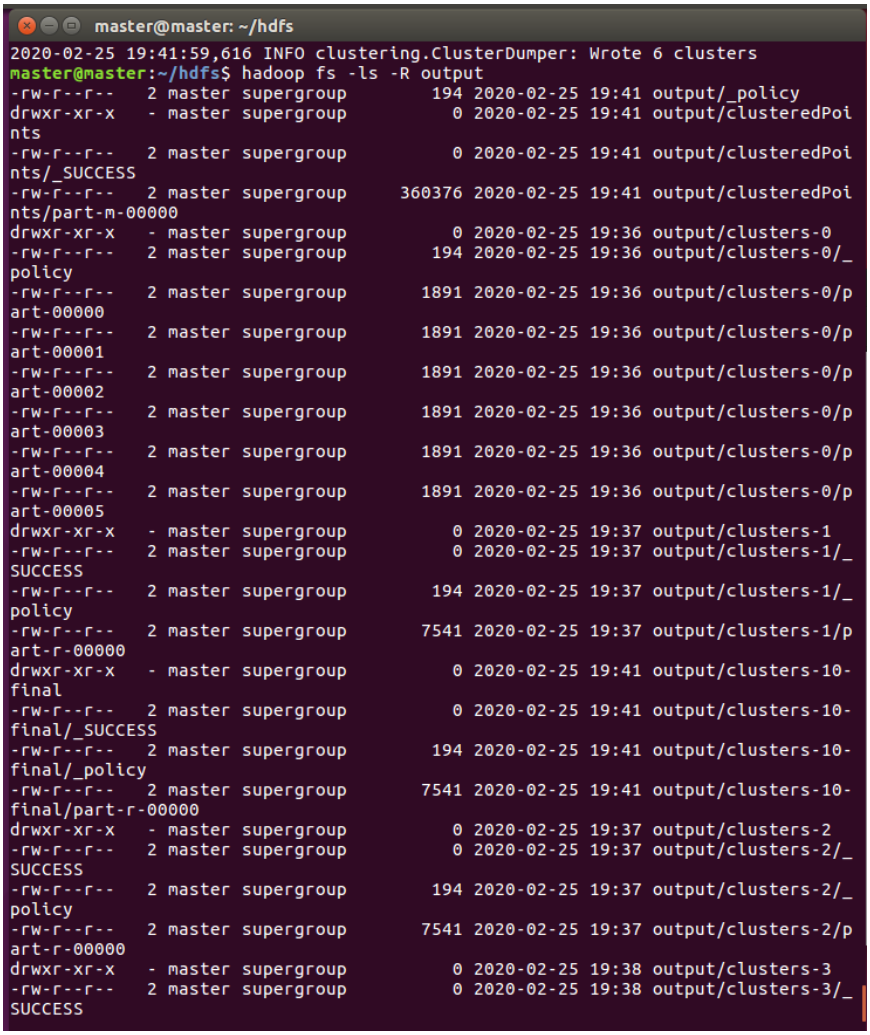
hadoop jar /opt/mahout/mahout-examples-0.12.2-job.jar

org.apache.mahout.clustering.syntheticcontrol.kmeans.Job



觀看結果：

hadoop fs -ls -R output



也可以使用 http://localhost:8088 觀看結果

mahout seqdumper -i output/clusters-9/part-r-00000 -o ~/data/aaa/ttt
（該將 part-r-00000 從分散式上導入到本地）