

## Hadoop\_Hive

- 開啟 Terminal，安裝 Hive 前，先啟動 hadoop。

`source /opt/hadoop/etc/hadoop/hadoop-env.sh` (新開 terminal 欲使用 hadoop 都須下此指令)

`start-all.sh` (啟動輸入一次即可)

- 查看是否正確啟動

`jps`

```
bao@master: ~
bash: /opt/hadoop/etc/hadoop/Hadoop-env.sh: No such file or directory
bao@master:~$ jps
3627 Jps
bao@master:~$ source /opt/hadoop/etc/hadoop/hadoop-env.sh
bao@master:~$ start-all.sh
start-all.sh: command not found
bao@master:~$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as bao in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [master]
Starting datanodes
Starting secondary namenodes [master]
Starting resourcemanager
Starting nodemanagers
slave: nodemanager is running as process 2250. Stop it first.
bao@master:~$ jps
5043 Jps
4200 SecondaryNameNode
4008 DataNode
4651 NodeManager
3870 NameNode
4510 ResourceManager
bao@master:~$
```

- 開啟網頁確認有無正常運作

<http://master:8088>

All Applications - Mozilla Firefox

master:8088/cluster

# hadoop

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed
0	0	0	0

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes
2	0

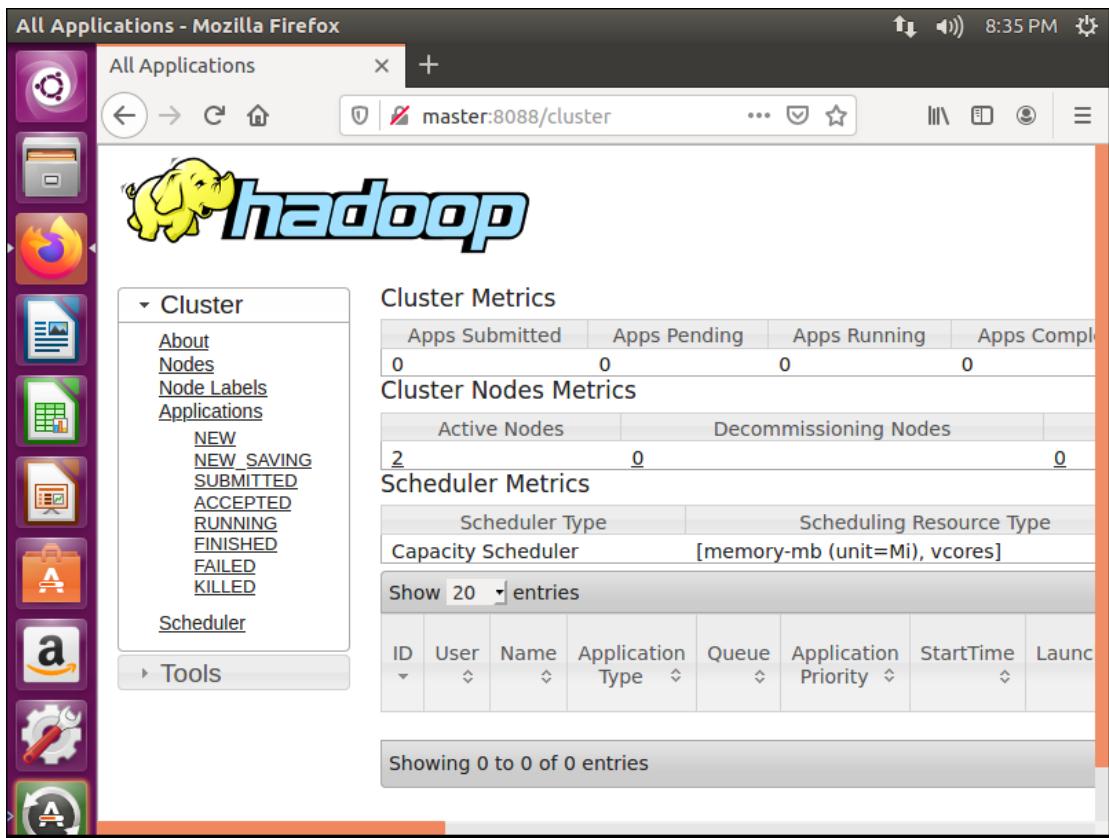
Scheduler Metrics

Scheduler Type	Scheduling Resource Type
Capacity Scheduler	[memory-mb (unit=Mi), vcores]

Show 20 entries

ID	User	Name	Application Type	Queue	Application Priority	StartTime	Launc

Showing 0 to 0 of 0 entries



<http://master:9870>

Namenode information

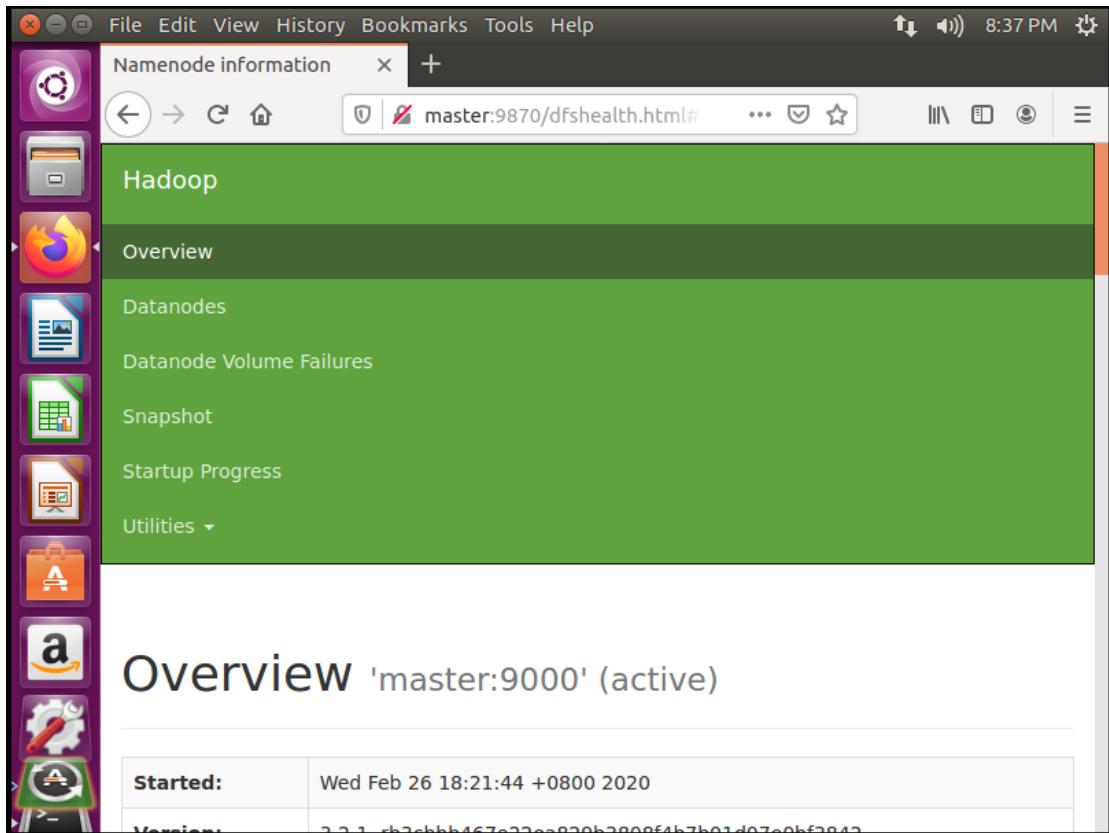
master:9870/dfshealth.html#

## Hadoop

- Overview
- Datanodes
- Datanode Volume Failures
- Snapshot
- Startup Progress
- Utilities ▾

## Overview 'master:9000' (active)

Started:	Wed Feb 26 18:21:44 +0800 2020
Version:	2.2.1-dbd2cb8b467e22ea020b2009f4b7b01d07e0bf2047



4. 安裝 Hive。

```
wget https://archive.apache.org/dist/hive/hive-3.1.2/apache-hive-3.1.2-bin.tar.gz
```

```
bao@master:~$ jps
5043 Jps
4200 SecondaryNameNode
4008 DataNode
4651 NodeManager
3870 NameNode
4510 ResourceManager
bao@master:~$ wget https://archive.apache.org/dist/hive/hive-3.1.2/apache-hive-3
.1.2-bin.tar.gz
--2020-02-26 20:39:31--  https://archive.apache.org/dist/hive/hive-3.1.2/apache-
hive-3.1.2-bin.tar.gz
Resolving archive.apache.org (archive.apache.org)... 163.172.17.199
Connecting to archive.apache.org (archive.apache.org)|163.172.17.199|:443...
HTTP request sent, awaiting response... 200 OK
Length: 278813748 (266M) [application/x-gzip]
Saving to: 'apache-hive-3.1.2-bin.tar.gz'

apache-hive-3.1.2-b 100%[=====] 265.90M  3.27MB/s    in 68s

2020-02-26 20:40:41 (3.92 MB/s) - 'apache-hive-3.1.2-bin.tar.gz' saved [27881374
8/278813748]

bao@master:~$
```

5. 解壓縮至 /opt 底下

```
tar zxvf apache-hive-3.1.2-bin.tar.gz
```

```
sudo mv apache-hive-3.1.2-bin /opt/hive
```

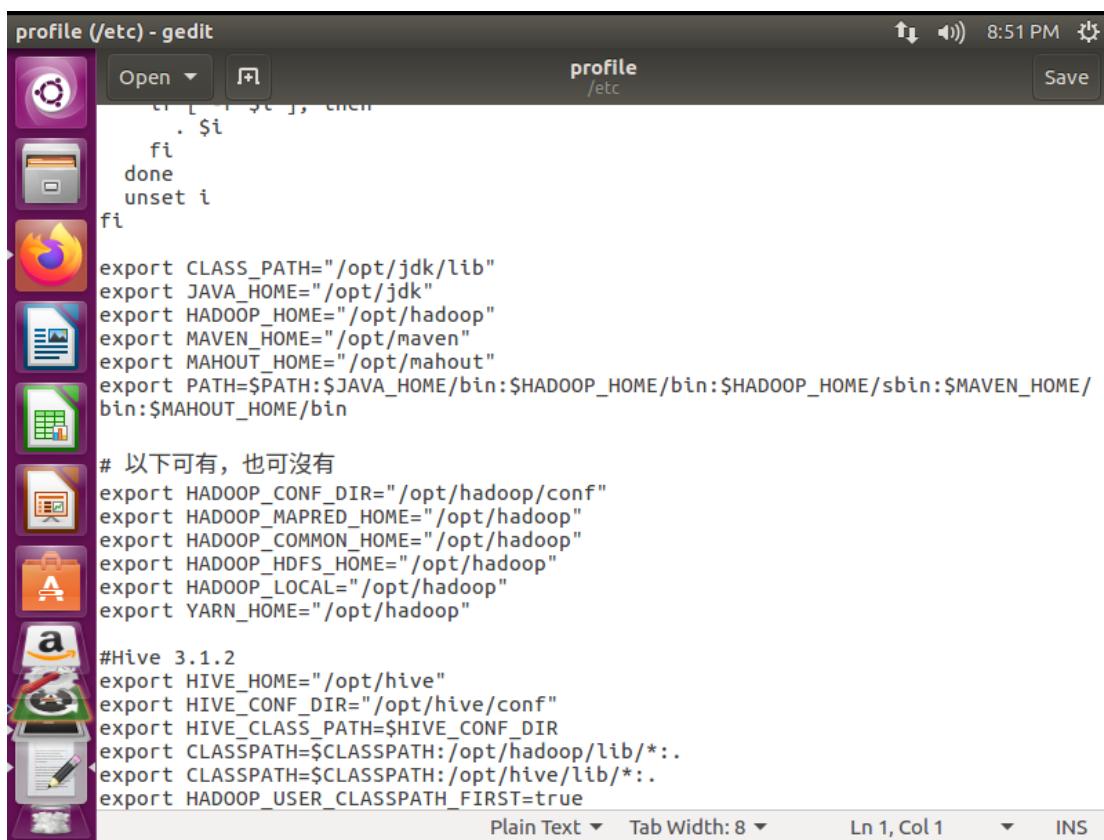
```
bao@master:~
apache-hive-3.1.2-bin/hcatalog/share/hcatalog/hive-hcatalog-core-3.1.2.jar
apache-hive-3.1.2-bin/hcatalog/share/hcatalog/hive-hcatalog-pig-adapter-3.1.2.ja
r
apache-hive-3.1.2-bin/hcatalog/share/hcatalog/hive-hcatalog-server-extensions-3.
1.2.jar
apache-hive-3.1.2-bin/hcatalog/share/webhcat/svr/lib/jersey-json-1.19.jar
apache-hive-3.1.2-bin/hcatalog/share/webhcat/svr/lib/jaxb-impl-2.2.3-1.jar
apache-hive-3.1.2-bin/hcatalog/share/webhcat/svr/lib/jackson-jaxrs-1.9.2.jar
apache-hive-3.1.2-bin/hcatalog/share/webhcat/svr/lib/jackson-xc-1.9.2.jar
apache-hive-3.1.2-bin/hcatalog/share/webhcat/svr/lib/jersey-core-1.19.jar
apache-hive-3.1.2-bin/hcatalog/share/webhcat/svr/lib/jsr311-api-1.1.1.jar
apache-hive-3.1.2-bin/hcatalog/share/webhcat/svr/lib/jersey-servlet-1.19.jar
apache-hive-3.1.2-bin/hcatalog/share/webhcat/svr/lib/hive-webhcat-3.1.2.jar
apache-hive-3.1.2-bin/hcatalog/share/webhcat/svr/lib/wadl-resourcedoc-doclet-1.4
.jar
apache-hive-3.1.2-bin/hcatalog/share/webhcat/svr/lib/xercesImpl-2.9.1.jar
apache-hive-3.1.2-bin/hcatalog/share/webhcat/svr/lib/xml-apis-1.3.04.jar
apache-hive-3.1.2-bin/hcatalog/share/webhcat/svr/lib/commons-exec-1.1.jar
apache-hive-3.1.2-bin/hcatalog/share/webhcat/svr/lib/jul-to-slf4j-1.7.10.jar
apache-hive-3.1.2-bin/hcatalog/share/webhcat/java-client/hive-webhcat-java-clien
t-3.1.2.jar
bao@master:~$ sudo mv apache-hive-3.1.2-bin /opt/hive
[sudo] password for bao:
bao@master:~$
```

## 6. 設定環境變數

**sudo gedit /etc/profile**

並將以下程式碼寫入檔案最後面，寫完右上角 save。

```
export HIVE_HOME="/opt/hive"
export HIVE_CONF_DIR="/opt/hive/conf"
export HIVE_CLASS_PATH=$HIVE_CONF_DIR
export CLASSPATH=$CLASSPATH:/opt/hadoop/lib/*
export CLASSPATH=$CLASSPATH:/opt/hive/lib/*
export HADOOP_USER_CLASSPATH_FIRST=true
export PATH=$PATH:$HIVE_HOME /bin
```



## 7. 執行環境變數設定

source /etc/profile

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

8. 使用 Hadoop HDFS 命令來創建/ tmp 和/ user/學號/ hive / warehouse 。

先利用 `ls` 查看 `hdfs` 有哪些資料夾

**hadoop fs -ls /**

```
bao@master:~  
  
(gedit:6224): Gtk-WARNING **: Calling Inhibit failed: GDBus.Error:org.freedesktop.DBus.Error.ServiceUnknown: The name org.gnome.SessionManager was not provided by any .service files  
  
** (gedit:6224): WARNING **: Set document metadata failed: Setting attribute met  
adata::gedit-spell-enabled not supported  
  
** (gedit:6224): WARNING **: Set document metadata failed: Setting attribute met  
adata::gedit-encoding not supported  
  
** (gedit:6224): WARNING **: Set document metadata failed: Setting attribute met  
adata::gedit-position not supported  
bao@master:~$ sudo gedit /etc/profile  
  
(gedit:6266): IBUS-WARNING **: The owner of /home/bao/.config/ibus/bus is not ro  
ot!  
  
** (gedit:6266): WARNING **: Set document metadata failed: Setting attribute met  
adata::gedit-position not supported  
bao@master:~$ source /etc/profile  
bao@master:~$ source /opt/hadoop/etc/hadoop/hadoop-env.sh  
bao@master:~$ hadoop fs -ls /  
bao@master:~$
```

利用 mkdir 指令創建

```
hdfs dfs -mkdir /tmp  
hdfs dfs -mkdir /user  
hdfs dfs -mkdir /user/****  
hdfs dfs -mkdir /user/****/hive  
hdfs dfs -mkdir /user/****/hive/warehouse
```

Note:\*\*\*\*填入你的學號!!!!!!

在使用 ls 指令確認創建成功

```
hadoop fs -ls /  
hadoop fs -ls /user  
hadoop fs -ls /user/****  
hadoop fs -ls /user/****/hive
```

Note:\*\*\*\*填入你的學號!!!!!!

```

bao@master:~$ source /etc/profile
bao@master:~$ source /opt/hadoop/etc/hadoop/hadoop-env.sh
bao@master:~$ hadoop fs -ls /
bao@master:~$ hdfs dfs -mkdir /tmp
bao@master:~$ hdfs dfs -mkdir /user
bao@master:~$ hdfs dfs -mkdir /user/bao
bao@master:~$ hdfs dfs -mkdir /user/bao/hive
bao@master:~$ hdfs dfs -mkdir /user/bao/hive/warehouse
bao@master:~$ hadoop fs -ls /
Found 2 items
drwxr-xr-x  - bao supergroup          0 2020-02-26 21:02 /tmp
drwxr-xr-x  - bao supergroup          0 2020-02-26 21:05 /user
bao@master:~$ hadoop fs -ls /user
Found 1 items
drwxr-xr-x  - bao supergroup          0 2020-02-26 21:05 /user/bao
bao@master:~$ hadoop fs -ls /user/bao
Found 1 items
drwxr-xr-x  - bao supergroup          0 2020-02-26 21:06 /user/bao/hive
bao@master:~$ hadoop fs -ls /user/bao/hive
Found 1 items
drwxr-xr-x  - bao supergroup          0 2020-02-26 21:06 /user/bao/hive/warehouse
se
bao@master:~$

```

也可利用利用 <http://master:9870>,點選 Utilities - Browse the filesystem  
查看剛剛創建的資料夾

The screenshot shows the HDFS Browser interface with the title "Browsing HDFS". The address bar shows "master:9870/explorer.h". The main area displays a table of file system entries under the path "/". The table has columns: Permission, Owner, Group, Size, Last Modified, Replication, and Block Size. There are two entries:

	Permission	Owner	Group	Size	Last Modified	Replication	Block Size
<input type="checkbox"/>	drwxr-xr-x	bao	supergroup	0 B	Feb 26 21:02	0	0
<input type="checkbox"/>	drwxr-xr-x	bao	supergroup	0 B	Feb 26 21:05	0	0

Below the table, it says "Showing 1 to 2 of 2 entries".

## 9. 變更檔案目錄權限

```
hdfs dfs -chmod g+w /tmp
```

```
hdfs dfs -chmod g+w /user/****/hive/warehouse
```

Note:\*\*\*\*填入你的學號!!!!!!

利用 ls 查看變更後 hdfs 資料夾權限

```
hdfs dfs -ls /
```

```
bao@master:~$ hdfs dfs -ls /
Found 2 items
drwxrwxr-x  - bao supergroup          0 2020-02-26 21:02 /tmp
drwxr-xr-x  - bao supergroup          0 2020-02-26 21:05 /user
bao@master:~$
```

## 10. 編輯 hive-env.sh

移至/opt/hive/conf 目錄下，複製該目錄下 hive-env.sh.template 檔案並重新命名為 hive-env.sh

```
cd ~
```

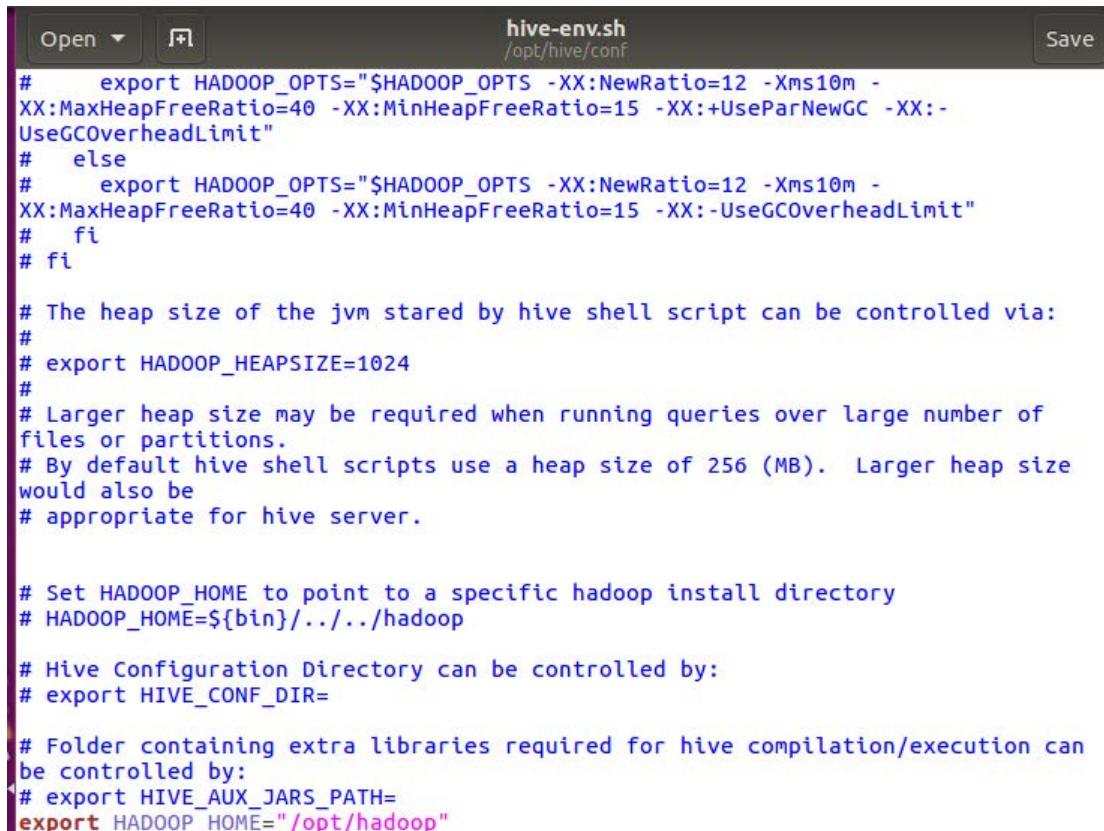
```
cd /opt/hive/conf
```

```
sudo cp hive-env.sh.template hive-env.sh
```

```
sudo gedit hive-env.sh
```

編輯檔案並在最下方增加此行並儲存：

```
export HADOOP_HOME="/opt/hadoop"
```



```
hive-env.sh
/opt/hive/conf

#      export HADOOP_OPTS="$HADOOP_OPTS -XX:NewRatio=12 -Xms10m -
#XX:MaxHeapFreeRatio=40 -XX:MinHeapFreeRatio=15 -XX:+UseParNewGC -XX:-
#UseGCOverheadLimit"
#  else
#      export HADOOP_OPTS="$HADOOP_OPTS -XX:NewRatio=12 -Xms10m -
#XX:MaxHeapFreeRatio=40 -XX:MinHeapFreeRatio=15 -XX:-UseGCOverheadLimit"
#  fi
# fi

# The heap size of the jvm stared by hive shell script can be controlled via:
#
# export HADOOP_HEAPSIZE=1024
#
# Larger heap size may be required when running queries over large number of
# files or partitions.
# By default hive shell scripts use a heap size of 256 (MB). Larger heap size
# would also be
# appropriate for hive server.

# Set HADOOP_HOME to point to a specific hadoop install directory
# HADOOP_HOME=${bin}/../../hadoop

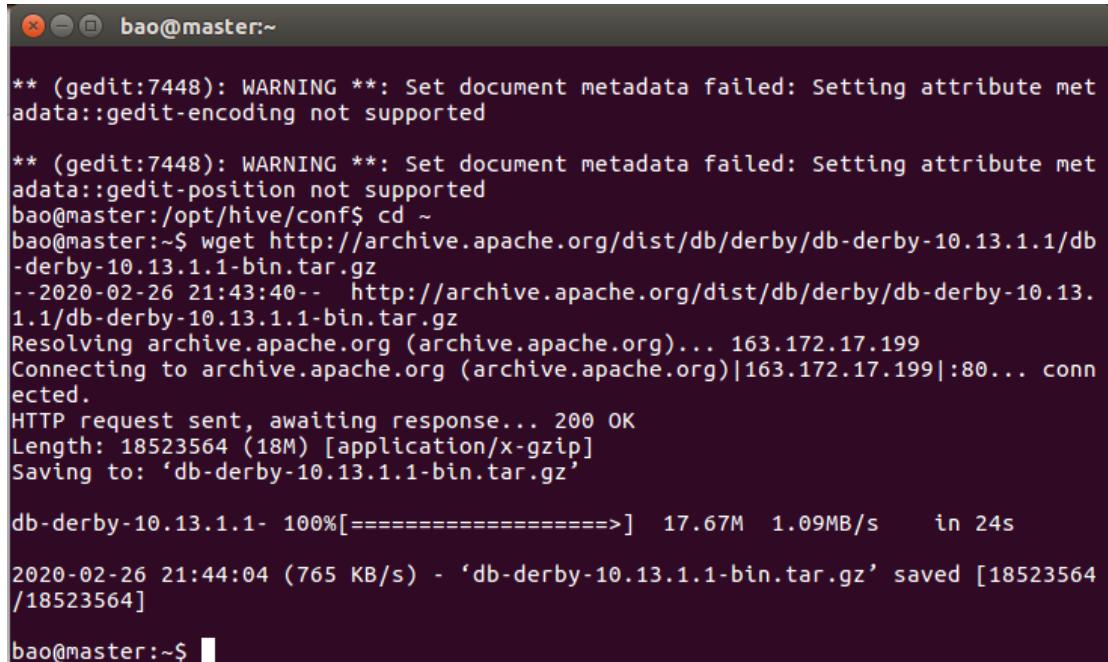
# Hive Configuration Directory can be controlled by:
# export HIVE_CONF_DIR=

# Folder containing extra libraries required for hive compilation/execution can
# be controlled by:
# export HIVE_AUX_JARS_PATH=
export HADOOP_HOME="/opt/hadoop"
```

## 11. 安裝 Apache Derby

```
cd ~
```

```
wget http://archive.apache.org/dist/db/derby/db-derby-10.13.1.1/db-derby-10.13.1.1-bin.tar.gz
```



```
bao@master:~
```

```
** (gedit:7448): WARNING **: Set document metadata failed: Setting attribute met
adata::gedit-encoding not supported

** (gedit:7448): WARNING **: Set document metadata failed: Setting attribute met
adata::gedit-position not supported
bao@master:/opt/hive/conf$ cd ~
bao@master:~$ wget http://archive.apache.org/dist/db/derby/db-derby-10.13.1.1/db-
-derby-10.13.1.1-bin.tar.gz
--2020-02-26 21:43:40-- http://archive.apache.org/dist/db/derby/db-derby-10.13.
1.1/db-derby-10.13.1.1-bin.tar.gz
Resolving archive.apache.org (archive.apache.org)... 163.172.17.199
Connecting to archive.apache.org (archive.apache.org)|163.172.17.199|:80... conn
ected.
HTTP request sent, awaiting response... 200 OK
Length: 18523564 (18M) [application/x-gzip]
Saving to: 'db-derby-10.13.1.1-bin.tar.gz'

db-derby-10.13.1.1- 100%[=====] 17.67M  1.09MB/s    in 24s
2020-02-26 21:44:04 (765 KB/s) - 'db-derby-10.13.1.1-bin.tar.gz' saved [18523564
/18523564]
```

```
bao@master:~$
```

解壓縮到/opt 下

```
sudo tar zxvf db-derby-10.13.1.1-bin.tar.gz
```

```
sudo mv db-derby-10.13.1.1-bin /opt/derby
```

配置環境變數

```
sudo gedit /etc/profile
```

將以下程式碼加在檔案最下面並且儲存

```
export DERBY_HOME="/opt/derby"
```

```
export PATH=$PATH:$DERBY_HOME/bin
```

```
export PATH=$PATH:$DERBY_HOME/bin
```

```
export CLASSPATH=$CLASSPATH:$DERBY_HOME/lib/derby.jar:$DERBY_HOME/lib/derbytools.jar
```

```

profile
/etc
Save
export CLASSPATH="/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

# 以下可有，也可沒有
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"

#Hive 3.1.2
export HIVE_HOME="/opt/hive"
export HIVE_CONF_DIR="/opt/hive/conf"
export HIVE_CLASS_PATH=$HIVE_CONF_DIR
export CLASSPATH=$CLASSPATH:/opt/hadoop/lib/*:.
export CLASSPATH=$CLASSPATH:/opt/hive/lib/*:.
export HADOOP_USER_CLASSPATH_FIRST=true

#Derby
export DERBY_HOME="/opt/derby"
export PATH=$PATH:$DERBY_HOME/bin
export CLASSPATH=$CLASSPATH:$DERBY_HOME/lib/derby.jar:$DERBY_HOME/lib/derbytools.jar

```

執行環境變數設定

```

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

```

在/opt/derby 目錄下建立一個 data 目錄用來儲存 Metastore 數據

```

cd ~
cd /opt/derby
sudo mkdir data

```

## 12. 配置 Hive Metastore

移至 /opt/hive/conf 目錄下，複製該目錄下 hive-default.xml.template 檔案並重新命名為

```

hive-site.xml
cd ~
cd /opt/hive/conf
sudo cp hive-default.xml.template hive-site.xml

```

編輯 hive-site.xml 文件

```

sudo gedit hive-site.xml

```

將<configuration> and </configuration>之間的內容清除，增加下列內容即可

```

<property>
<name>javax.jdo.option.ConnectionURL</name>
<value>jdbc:derby:;databaseName=metastore_db;create=true</value>
<description>

```

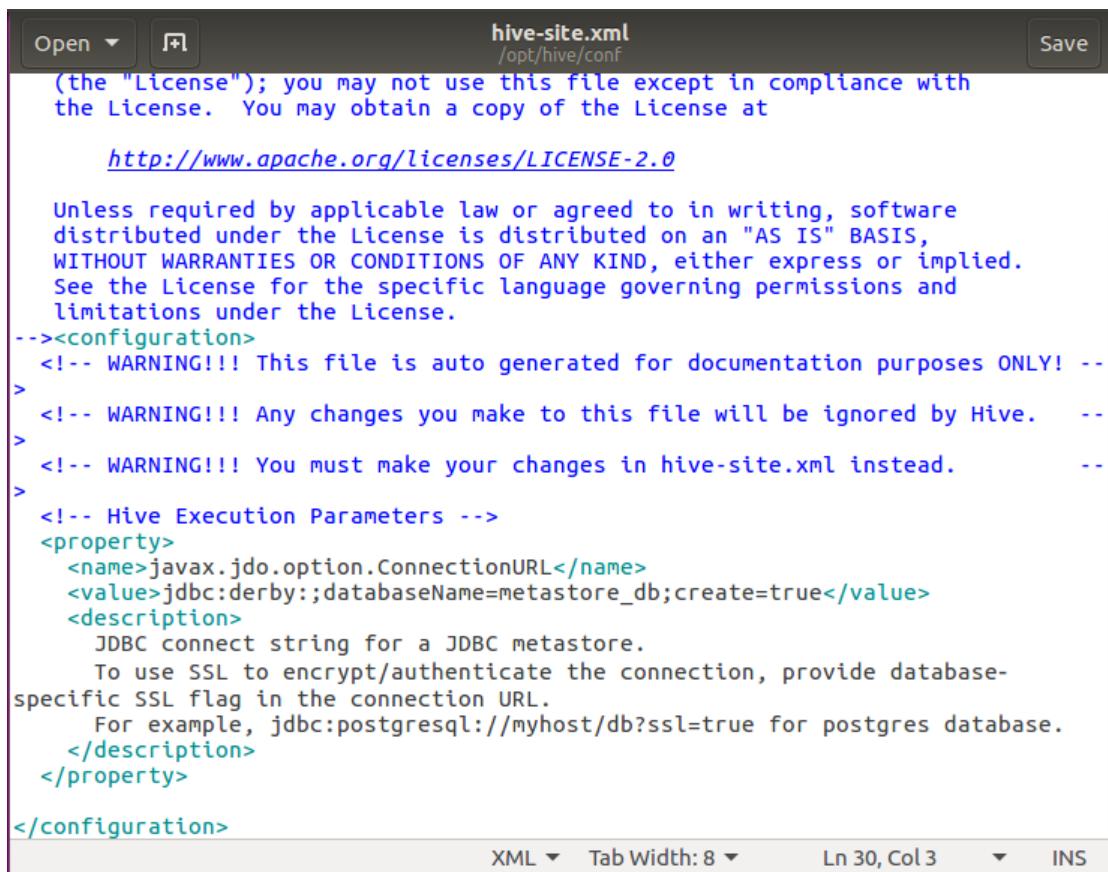
JDBC connect string for a JDBC metastore.

To use SSL to encrypt/authenticate the connection, provide database-specific SSL flag in the connection URL.

For example, `jdbc:postgresql://myhost/db?ssl=true` for postgres database.

`</description>`

`</property>`



The screenshot shows a code editor window with the title "hive-site.xml" and the path "/opt/hive/conf". The file contains the Apache License 2.0 header and configuration code for Hive. The configuration section includes a property for the JDBC connection URL, which is set to "jdbc:derby:;databaseName=metastore\_db;create=true". A detailed description for this property explains the use of JDBC for a metastore and the option to use SSL for encryption and authentication.

```
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

--><configuration>
  <!-- WARNING!!! This file is auto generated for documentation purposes ONLY! -->
  <!-- WARNING!!! Any changes you make to this file will be ignored by Hive. -->
  <!-- WARNING!!! You must make your changes in hive-site.xml instead. -->
  <!-- Hive Execution Parameters -->
<property>
  <name>javax.jdo.option.ConnectionURL</name>
  <value>jdbc:derby:;databaseName=metastore_db;create=true</value>
  <description>
    JDBC connect string for a JDBC metastore.
    To use SSL to encrypt/authenticate the connection, provide database-
    specific SSL flag in the connection URL.
    For example, jdbc:postgresql://myhost/db?ssl=true for postgres database.
  </description>
</property>

</configuration>
```

在/opt/hive/conf 目錄下，建立一個名為 jpox.properties 的文件

`cd ~`

`cd /opt/hive/conf`

`sudo gedit jpox.properties`

在該文件增加下列內容，並存檔跳出

```
javax.jdo.PersistenceManagerFactoryClass =
org.jpox.PersistenceManagerFactoryImpl
org.jpox.autoCreateSchema = false
org.jpox.validateTables = false
org.jpox.validateColumns = false
org.jpox.validateConstraints = false
org.jpox.storeManagerType = rdbms
org.jpox.autoCreateSchema = true
```

```
org.jpox.autoStartMechanismMode = checked
org.jpox.transactionIsolation = read_committed
javax.jdo.option.DetachAllOnCommit = true
javax.jdo.option.NontransactionalRead = true
javax.jdo.option.ConnectionDriverName = org.apache.derby.jdbc.ClientDriver
javax.jdo.option.ConnectionURL = jdbc:derby://hadoop1:1527/metastore_db;create
= true
javax.jdo.option.ConnectionUserName = APP
javax.jdo.option.ConnectionPassword = mine
```

The screenshot shows a text editor window titled "jpox.properties" located at "/opt/hive/conf". The window has standard operating system controls (Open, Save, etc.) at the top. The main content area contains the configuration code shown above. At the bottom of the window, there are buttons for "Plain Text", "Tab Width: 8", "Ln 1, Col 1", and "INS".

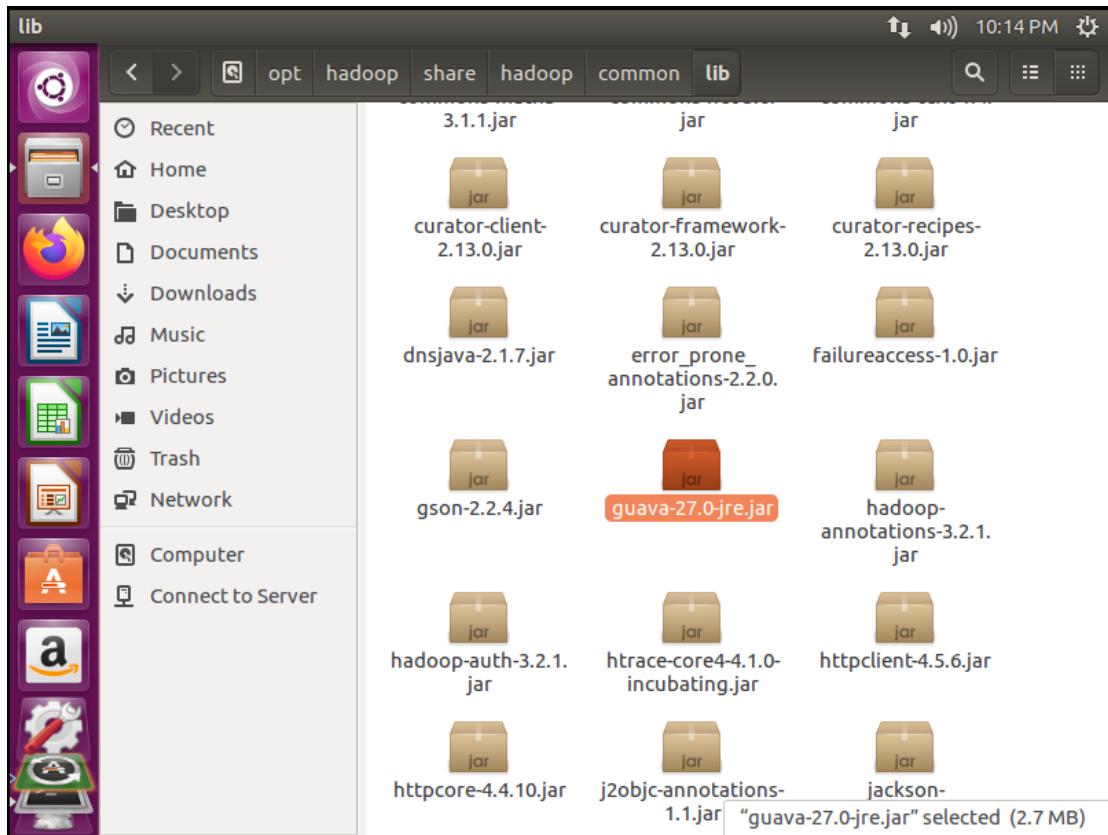
```
javax.jdo.PersistenceManagerFactoryClass =
org.jpox.PersistenceManagerFactoryImpl
org.jpox.autoCreateSchema = false
org.jpox.validateTables = false
org.jpox.validateColumns = false
org.jpox.validateConstraints = false
org.jpox.storeManagerType = rdbms
org.jpox.autoCreateSchema = true
org.jpox.autoStartMechanismMode = checked
org.jpox.transactionIsolation = read_committed
javax.jdo.option.DetachAllOnCommit = true
javax.jdo.option.NontransactionalRead = true
javax.jdo.option.ConnectionDriverName = org.apache.derby.jdbc.ClientDriver
javax.jdo.option.ConnectionURL = jdbc:derby://hadoop1:1527/metastore_db;create
= true
javax.jdo.option.ConnectionUserName = APP
javax.jdo.option.ConnectionPassword = mine
```

設置 Hive 資料夾的權限

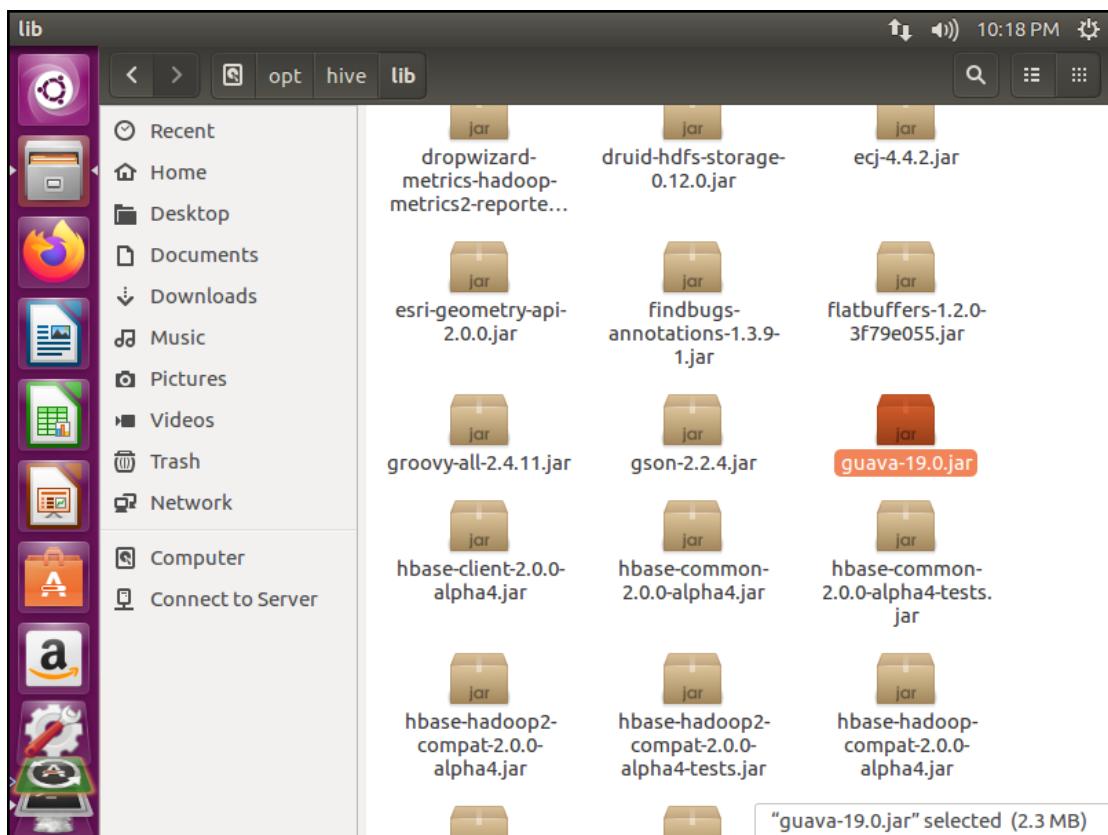
```
cd ~
cd /opt
sudo chown -R **** hive
Note:****填入 ubuntu 使用者帳號
```

同步 Hadoop 和 hive 的 guava 版本(兩者取較高的版本)

下圖為 Hadoop 的 guava 所在位置



下圖為 Hive 的 guava 所在位置



刪除 Hive 的 guava-19.0.jar

複製 Hadoop 的 guava-27.0-jre.jar 貼至 Hive 的 lib 資料夾裡

Metastore schema initialization

回到 Terminal。從 Hive 2.1 開始，我們需要運行 schematool 命令作為初始化步驟，在這邊使用 derby as db type。

```
cd ~
```

```
cd /opt/hive/bin
```

```
./schematool -dbType derby -initSchema
```

```
bao@master:/opt/hive/bin$ ./schematool -dbType derby -initSchema
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/opt/hive/lib/log4j-slf4j-impl-2.10.0.jar!/org
/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/opt/hadoop/share/hadoop/common/lib/slf4j-log4
j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
Metastore connection URL:      jdbc:derby:;databaseName=metastore_db;create=tr
ue
Metastore Connection Driver :  org.apache.derby.jdbc.EmbeddedDriver
Metastore connection User:    APP
Starting metastore schema initialization to 3.1.0
Initialization script hive-schema-3.1.0.derby.sql
```

```
Initialization script completed
schemaTool completed
bao@master:/opt/hive/bin$
```

至 /opt/hive/bin/目錄下直接輸入 ./hive (就可以進入 Hive 的互動式查詢介面)。

```
./hive
```

顯示所有資料表:

```
show tables;
```

```
bao@master:/opt/hive/bin

Initialization script completed
schemaTool completed
bao@master:/opt/hive/bin$ ./hive
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/opt/hive/lib/log4j-slf4j-impl-2.10.0.jar!/org
/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/opt/hadoop/share/hadoop/common/lib/slf4j-log4
j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
Hive Session ID = 146e3758-dab2-4be5-bc04-a96603ac28ba

Logging initialized using configuration in jar:file:/opt/hive/lib/hive-common-3.
1.2.jar!/hive-log4j2.properties Async: true
Hive Session ID = 562cabbf-d193-4205-a7f3-80030bd088b7
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versio
ns. Consider using a different execution engine (i.e. spark, tez) or using Hive
1.X releases.
hive> show tables;
OK
Time taken: 0.518 seconds
hive>
```

## Hive 練習-建立資料表

建立一個 emp 資料表，透過 CREATE TABLE 語句創建一個名為 emp 資料表。emp

表中的字段和數據類型如下：

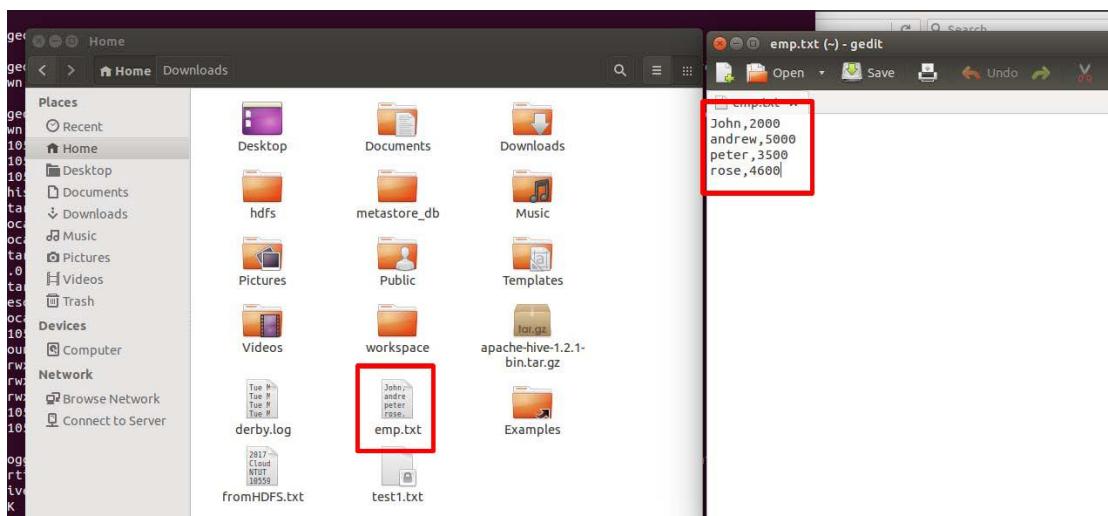
r.No	字段名稱	數據類型
1	Name	String
2	Esal	int

create table emp(ename string,esal int) row format delimited fields terminated by ',' stored as textfile;

```
hive> create table emp(ename string,esal int) row format delimited fields termin
ated by ',' stored as textfile;
OK
Time taken: 0.551 seconds
hive> █
```

在本機端目錄 Home 下建立 emp.txt 檔，檔案內容資料如下：

```
John,2000
andrew,5000
peter,3500
rose,4600
```



將 emp.txt 檔案匯入資料到 Hive 資料表。

load data local inpath '/home/\*\*\*\*/emp.txt' into table emp;

Note:\*\*\*\*填入 Ubuntu 使用者帳戶

```
hive> load data local inpath '/home/bao/emp.txt' into table emp;
Loading data to table default.emp
OK
Time taken: 1.134 seconds
hive> █
```

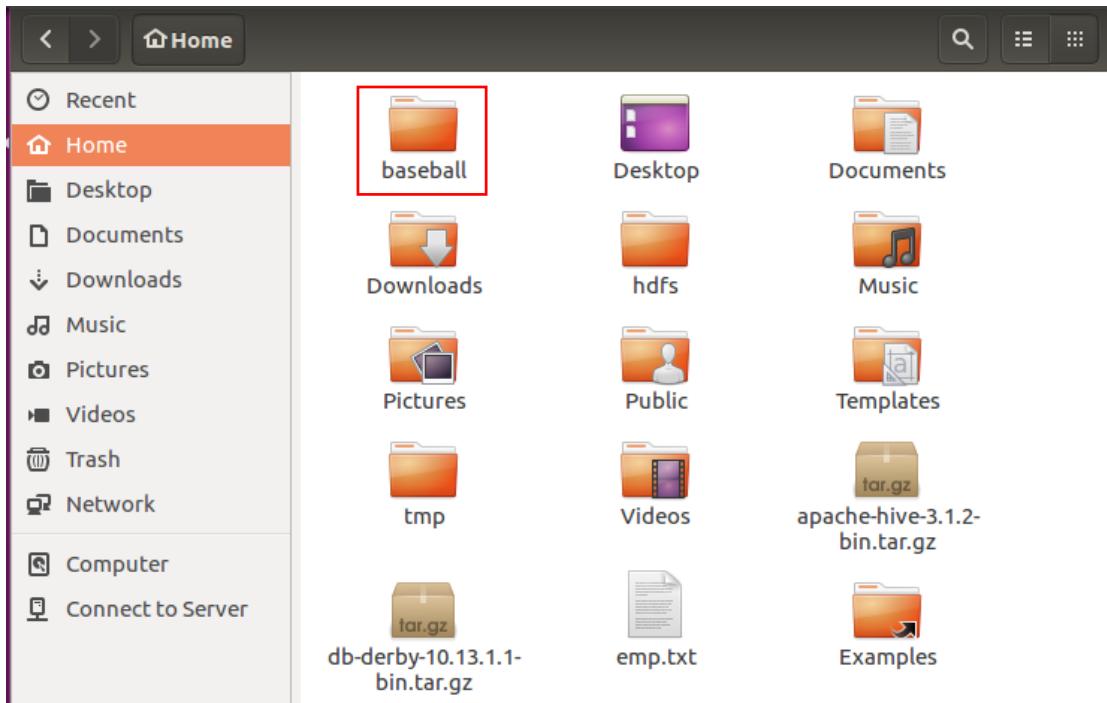
檔案匯入完畢後可透過 SELECT 來進行資料的檢索。

```
Select * from emp;
```

```
Time taken: 1.134 seconds
hive> Select * from emp;
OK
John    2000
andrew  5000
peter   3500
rose    4600
Time taken: 1.641 seconds, Fetched: 4 row(s)
hive> █
```

## Hive 練習-將 CSV 資料匯出到 Hive

於本機端建立一個資料夾 /baseball。



下載 2012 年的球賽統計資料，並解壓縮至 baseball 資料夾中。

cd ~

cd baseball

wget <http://seanlahman.com/files/database/lahman2012-csv.zip>

unzip lahman2012-csv.zip

```
bao@master:~/baseball
inflating: Appearances.csv
inflating: AwardsManagers.csv
inflating: AwardsPlayers.csv
inflating: AwardsShareManagers.csv
inflating: AwardsSharePlayers.csv
inflating: Batting.csv
inflating: BattingPost.csv
inflating: Fielding.csv
inflating: FieldingOF.csv
inflating: FieldingPost.csv
inflating: HallOfFame.csv
inflating: Managers.csv
inflating: ManagersHalf.csv
inflating: Pitching.csv
inflating: PitchingPost.csv
inflating: readme 2012.txt
inflating: Salaries.csv
inflating: Schools.csv
inflating: SchoolsPlayers.csv
inflating: SeriesPost.csv
inflating: Teams.csv
inflating: TeamsFranchises.csv
inflating: Master.csv
bao@master:~/baseball$
```

在 HDFS 上建立一個名為 baseball 的目錄，並將 /baseball 資料夾中所有 CSV 檔案上傳到該目錄中

(Hadoop 記得啟動 詳見第一頁)

```
hdfs dfs -mkdir /baseball
```

```
hdfs dfs -put *.csv /baseball
```

```
hdfs dfs -ls /baseball
```

```
bao@master:~/baseball
SV
-rw-r--r-- 2 bao supergroup 573945 2020-02-26 23:05 /baseball/FieldingPost.csv
-rw-r--r-- 2 bao supergroup 175990 2020-02-26 23:05 /baseball/HallOfFame.csv
SV
-rw-r--r-- 2 bao supergroup 130719 2020-02-26 23:05 /baseball/Managers.csv
-rw-r--r-- 2 bao supergroup 3662 2020-02-26 23:05 /baseball/ManagersHalf.csv
-rw-r--r-- 2 bao supergroup 3049250 2020-02-26 23:05 /baseball/Master.csv
-rw-r--r-- 2 bao supergroup 3602473 2020-02-26 23:05 /baseball/Pitching.csv
-rw-r--r-- 2 bao supergroup 381812 2020-02-26 23:05 /baseball/PitchingPost.csv
-rw-r--r-- 2 bao supergroup 700024 2020-02-26 23:05 /baseball/Salaries.csv
-rw-r--r-- 2 bao supergroup 42933 2020-02-26 23:05 /baseball/Schools.csv
-rw-r--r-- 2 bao supergroup 180758 2020-02-26 23:05 /baseball/SchoolsPlayes.csv
-rw-r--r-- 2 bao supergroup 8369 2020-02-26 23:05 /baseball/SeriesPost.csv
SV
-rw-r--r-- 2 bao supergroup 550032 2020-02-26 23:05 /baseball/Teams.csv
-rw-r--r-- 2 bao supergroup 3238 2020-02-26 23:05 /baseball/TeamsFranchises.csv
-rw-r--r-- 2 bao supergroup 1609 2020-02-26 23:05 /baseball/TeamsHalf.csv
V
bao@master:~/baseball$
```

啟動 hive

```
cd ~
```

```
cd /opt/hive/bin
```

```
./hive
```

```
bao@master:/opt/hive/bin
-rw-r--r-- 2 bao supergroup 3238 2020-02-26 23:05 /baseball/TeamsFranchises.csv
-rw-r--r-- 2 bao supergroup 1609 2020-02-26 23:05 /baseball/TeamsHalf.csv
V
bao@master:~/baseball$ cd ~
bao@master:~$ cd /opt/hive/bin
bao@master:/opt/hive/bin$ ./hive
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/opt/hive/lib/log4j-slf4j-impl-2.10.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/opt/hadoop/share/hadoop/common/lib/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
Hive Session ID = 2b620a74-8b75-4337-8e10-506acb87bce1

Logging initialized using configuration in jar:file:/opt/hive/lib/hive-common-3.1.2.jar!/hive-log4j2.properties Async: true

Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
hive>
>
```

建立資料庫

```
create database baseball_****;
```

Note:\*\*\*\*為學號

```
hive> create database baseball_bao ;
OK
Time taken: 0.043 seconds
hive> █
```

建立 Hive 資料表，選定某一個想要分析的 CSV 資料，例如：Master.csv

Master 的資料型態如下表所示：

欄位	範例	資料型態
lahmanID	1	INT
playerID	aaronha01	STRING
managerID	NULL	INT
hofID	aaronha01h	STRING
birthYear	1934	INT
birthMonth	2	INT
birthDay	5	INT
birthCountry	USA	STRING
birthState	AL	STRING
birthCity	Mobile	STRING
deathYear	NULL	INT
deathMonth	NULL	INT
deathDay	NULL	INT
deathCountry	NULL	STRING
...略		

在 baseball\_\*\*\*\* 中建立 Master 資料表

```
create table baseball_****.Master
( lahmanID INT, playerID STRING, managerID INT, hofID STRING, birthYear INT,
birthMonth INT, birthDay INT, birthCountry STRING, birthState STRING, birthCity
STRING, deathYear INT, deathMonth INT, deathDay INT, deathCountry STRING,
deathState STRING, deathCity STRING, nameFirst STRING, nameLast STRING,
nameNote STRING, nameGiven STRING, nameNick STRING, weight INT, height INT,
```

```
bats STRING, throws STRING, debut STRING, finalGame STRING, college STRING,  
lahman40ID STRING, lahman45ID STRING, retroID STRING, holtzID STRING, bbrefID  
STRING ) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',';
```

Note:\*\*\*\*為學號

```
hive> create table baseball_bao.Master  
    > (Display all 633 possibilities? (y or n)  
    > (ID INT, playerID STRING, managerID INT, hofID STRING, birthYear INT, birthMonth INT, birthDay INT, birthCountry STRING, birthState STRING, birthCity STRING, deathYear INT, deathMonth INT, deathDay INT, deathCountry STRING, deathState STRING, deathCity STRING, nameFirst STRING, nameLast STRING, nameNote STRING, nameGiven STRING, nameNick STRING, weight INT, height INT, bats STRING, throws STRING, debut STRING, finalGame STRING, college STRING, lahman40ID STRING, lahman45ID STRING, retroID STRING, holtzID STRING, bbrefID STRING ) ROW FORMAT DELIMITED  
    > FIELDS TERMINATED BY ',' ;  
OK  
Time taken: 0.095 seconds  
hive>
```

CSV 檔案匯入資料到 Hive 資料表

```
LOAD DATA LOCAL INPATH '/home/****/baseball/Master.csv' INTO TABLE  
baseball_XXXX.Master;
```

Note:\*\*\*\*為 ubuntu 使用者帳戶,XXXX 為學號

檢查匯出結果

```
SHOW DATABASES;
```

```
hive> LOAD DATA LOCAL INPATH '/home/bao/baseball/Master.csv' INTO TABLE  
    > baseball_bao.Master;  
Loading data to table baseball_bao.master  
OK  
Time taken: 0.24 seconds  
hive> SHOW DATABASES;  
OK  
baseball_bao  
default  
Time taken: 0.021 seconds, Fetched: 2 row(s)  
hive> █
```

切換預設的資料庫，變成剛剛產生的 baseball 資料庫中

```
USE baseball_****; Note:****為學號
```

查詢目前的資料庫有哪幾個資料表

```
SHOW TABLES;
```

```
hive> SHOW TABLES;  
OK  
master  
Time taken: 0.026 seconds, Fetched: 1 row(s)  
hive> █
```

檢查一下剛剛建立的 baseball.master 資料表，內容是否正常

```
SELECT * FROM Master;
```

nthony	John	190	71	R	R	NULL	NULL	NULL	USA	MO	S	N
19414	mckenfr01	NULL	NULL	NULL	NULL	Frank	McKenna	McKenna				N
ULL	NULL	NULL										
ULL	NULL											
19415	mckenpa01	NULL	NULL	NULL	NULL	Patrick	McKenna	McKenna				N
ULL	NULL	NULL										
ULL	NULL											
19416	sulliwi01	NULL	NULL	NULL	NULL	NULL	NULL	NULL	USA	MO	S	
t. Louis	NULL	NULL	NULL	NULL	NULL				William	Sullivan		
NULL	NULL											
19417	gilgahu01	NULL	NULL	1852	Hugh	NULL	Gilgan	Ireland				N
ULL	NULL	NULL										
ULL	NULL											
19418	crossjo01	NULL	NULL	1858	1	6	Joe	Cross	IL	C		N
hicago	NULL	NULL	NULL									
ULL	NULL											
19419	snydech03	NULL	NULL	1890	8	20	Chubby	USA	NY	B		
uffalo	NULL	NULL	NULL									
ULL	NULL											
19420	ruperja99	NULL	ruperja99h	1867	8	5	USA	Jacob	N			
Y	New York	NULL	NULL	NULL	NULL							R
uppert				NULL	NULL							
Time taken:	0.134	seconds,	Fetched:	18126	row(s)							
hive>												

查詢預查詢之欄位資料

```
select lahmanID,playerID,birthYear,birthMonth,birthDay,birthState from Master limit 10;
```

hicago	NULL	NULL	NULL						Joe	Cross		N
ULL	NULL											
19419	snydech03	NULL	NULL	1890	8	20	USA	NY	B			
uffalo	NULL	NULL	NULL									
ULL	NULL											
19420	ruperja99	NULL	ruperja99h	1867	8	5	USA	Jacob	N			
Y	New York	NULL	NULL	NULL	NULL							R
uppert				NULL	NULL							
Time taken:	0.107	seconds,	Fetched:	18126	row(s)							
hive> select lahmanID,playerID,birthYear,birthMonth,birthDay,birthState from Master limit 10;												
OK												
NULL	playerID	NULL	NULL	NULL								
1	aaronha01	1934	2	5								
2	aaronto01	1939	8	5								
3	aasedo01	1954	9	8								
4	abadan01	1972	8	25								
5	abadijo01	1854	11	4								
6	abbated01	1877	4	15								
7	abbeybe01	1869	11	29								
8	abbeych01	1866	10	14								
9	abbotda01	1862	3	16								
Time taken:	0.093	seconds,	Fetched:	10	row(s)							
hive>												

建立球隊打擊資料表，並匯入資料

```
create table baseball_****.Batting
( playerID STRING, yearID INT, stint INT, teamID STRING, lgID STRING, G INT, G_batting
INT, AB INT, R INT, H INT, twoB INT, threeB INT, HR INT,
RBI INT, SB INT, CS INT, BB INT, SO INT, IBB INT, HBP INT, SH INT, SF INT, GIDP INT,
G_old INT ) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',';
```

Note:\*\*\*\*為學號

```

hive> create table baseball_bao.Batting
      > ( playerID STRING, yearID INT, stint INT, teamID STRING, lgID STRING, G IN
T, G_batting INT, AB INT, R INT, H INT, twoB INT, threeB INT, HR INT,
      > RBI INT, SB INT, CS INT, BB INT, SO INT, IBB INT, HBP INT, SH INT, SF INT,
      GIDP INT, G_old INT ) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' ;
OK
Time taken: 0.069 seconds

```

匯入打擊資料

LOAD DATA INPATH "/baseball/Batting.csv" OVERWRITE INTO TABLE

baseball\_\*\*\*\*.Batting;

Note:\*\*\*\*為學號

```

hive> LOAD DATA INPATH "/baseball/Batting.csv" OVERWRITE INTO TABLE baseball_bao
.Batting;
Loading data to table baseball_bao.batting
OK
Time taken: 0.155 seconds
hive> select * from Batting limit 3;
OK
playerID      NULL    NULL    teamID   lgID    NULL    NULL    NULL    NULL    NULL    N
ULL      NULL    NULL    NULL    NULL    NULL    NULL    NULL    NULL    NULL    N
ULL      NULL    NULL    NULL    NULL    NULL    NULL    NULL    NULL    NULL    N
aardsda01     2004     1      SFN     NL      11      11      0       0       0       0
0          0       0       0       0       0       0       0       0       0       0
0          0       11      0       0       0       0       0       0       0       0
aardsda01     2006     1      CHN     NL      45      43      2       0       0       0
0          0       0       0       0       0       0       0       0       0       1
0          0       45      0       0       0       0       0       0       0       0
Time taken: 0.115 seconds, Fetched: 3 row(s)
hive>

```

JOIN 跨表查詢

SELECT A.PlayerID, B.teamID, B.AB, B.R, B.H, B.twoB, B.threeB, B.HR, B.RBI FROM  
Master A JOIN BATTING B ON A.playerID = B.playerID ;

```

bao@master:/opt/hive/bin
zuvelpa01    ATL      5      0      0      0      0      0      0      0
zuvelpa01    ATL     25     2      5      1      0      0      0      1
zuvelpa01    ATL    190    16     48     8      1      0      0      4
zuvelpa01    NYA     48     2      4      1      0      0      0      2
zuvelpa01    NYA     34     2      6      0      0      0      0      0
zuvelpa01    CLE    130     9     30     5      1      0      0      7
zuvelpa01    CLE     58    10     16     2      0      2      0      6
zuvelpa01    KCA      0     0      0      0      0      0      0      0
zuverge01    CLE      0     0      0      0      0      0      0      0
zuverge01    CLE      0     1      0      0      0      0      0      0
zuverge01    CIN      2     1      1      0      0      0      0      0
zuverge01    DET     64     1      8      1      0      0      0      3
zuverge01    DET      4     0      0      0      0      0      0      0
zuverge01    BAL     23     1      5      1      0      0      0      0
zuverge01    BAL     17     0      2      0      0      0      0      2
zuverge01    BAL     23     1      3      0      0      0      0      0
zuverge01    BAL      9     0      2      0      1      0      0      2
zuverge01    BAL      0     0      0      0      0      0      0      0
zwilldu01    CHA     87     7     16     5      0      0      0      5
zwilldu01    CHF    592    91    185    38      8     16     95
zwilldu01    CHF    548    65    157    32      7     13     94
zwilldu01    CHN     53     4      6      1      0      1      8
Time taken: 30.197 seconds, Fetched: 96610 row(s)
hive>

```

```
source /opt/hadoop/etc/hadoop/hadoop-env.sh  
start-all.sh
```

```
sudo apt-get install xsltproc  
wget http://gis.taiwan.net.tw/XMLReleaseALL_public/hotel_C_f.xml  
cat hotel_C_f.xml
```

```
gedit hotel.xslt
```

```
<?xml version="1.0"?>  
<xsl:stylesheet version="1.0"  
    xmlns:xsl="http://www.w3.org/1999/XSL/Transform" >  
    <xsl:output method="text" indent="no"/>  
    <xsl:template match="/">  
        <xsl:for-each select="XML_Head/Infos/Info">  
            <xsl:sort select="Zipcode"/>  
            <xsl:value-of select="concat(Zipcode,',',Name,',',Add,',',Tel)"/>,  
        </xsl:for-each>  
    </xsl:template>  
</xsl:stylesheet>
```

```
xsltproc hotel.xslt hotel_C_f.xml>hotel.txt  
more hotel.txt
```

空格替換成逗號

```
sed 's/\ //g'< hotel.txt >hotel.csv
```

```
hdfs dfs -mkdir /etl
```

```
hdfs dfs -mkdir /etl/hotel
```

```
hdfs dfs -put hotel.csv /etl/hotel
```

```
gedit hotel.sql
```

```
CREATE EXTERNAL TABLE hotel(
    zip STRING,
    name STRING,
    add STRING,
    tel STRING
)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION '/etl/hotel';
```

```
cd /opt/hive/bin
```

```
hive -S -f /home/kjy/tmp/hive/hotel.sql
```

```
hive -S -e "select * from hotel"
```

```
hive -S -e "select id, a.namee, b.name, b.add, b.tel from customer a LEFT OUTER JOIN
hotel b ON trim(a.zip) = trim(b.zip)"
```

```
firefox http://download.post.gov.tw/post/download/Zip32_utf8_10501_1.csv
```

```
iconv -f utf16 -t utf8 Zip32_utf8_10501_1.csv>a.csv
```

```
hdfs dfs -ls /
hdfs dfs -put a.csv /etl
```

```
cd ~/tmp
wget http://data.fda.gov.tw/cacheData/35_2.csv -O dragstore.csv
```

```
pig -x local 2>/dev/null
```

```
load a = '/home/kjy/tmp/dragstore.csv';
load a = '/home/kjy/Downloads/dragstore.csv';
dump a;
```

```
store a into 'hdfs://ubuntu:8020/etl/dragstore.csv' USING PigStorage(',');
```