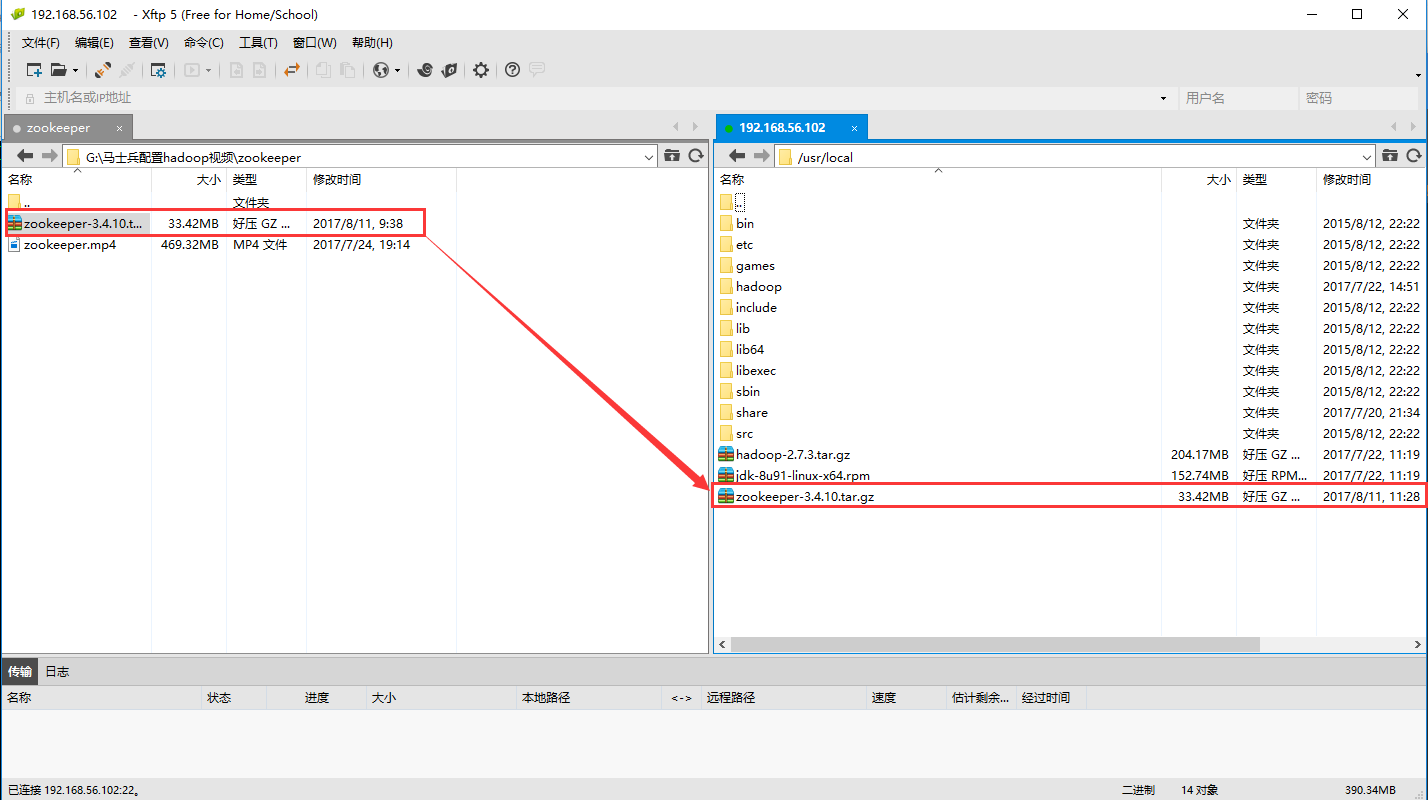
**大数据平台部分实践作业5#**

# 通过命令和WebUI验证启动是否成功

## 1.1、Zookeeper的配置与验证

1、用Xshell和Xftp将下载好的zookeeper复制到master的usr/local/目录下。

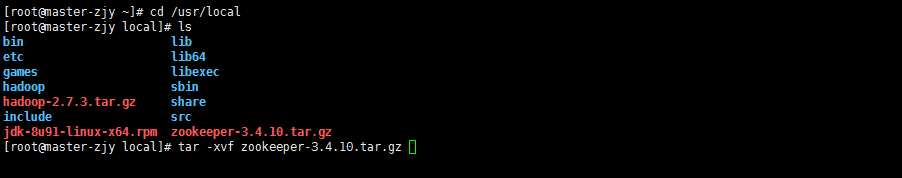


2、在/etc/local目录下解压zookeeper

cd /usr/local

tar -xvf zookeeper-3.4.10.tar.gz

mv zookeeper-3.4.10 zookeeper



3、配置zoo.cfg文档指定集群机器名字、IP、更改临时文件

cd /usr/local/zookeeper/conf

cp zoo\_sample.cfg zoo.cfg

vim zoo.cfg

dataDir=/var/zjy/zk

server.1=master:2888:3888

server.2=slave1:2888:3888

server.3=slave2:2888:3888

server.4=slave3:2888:3888

（

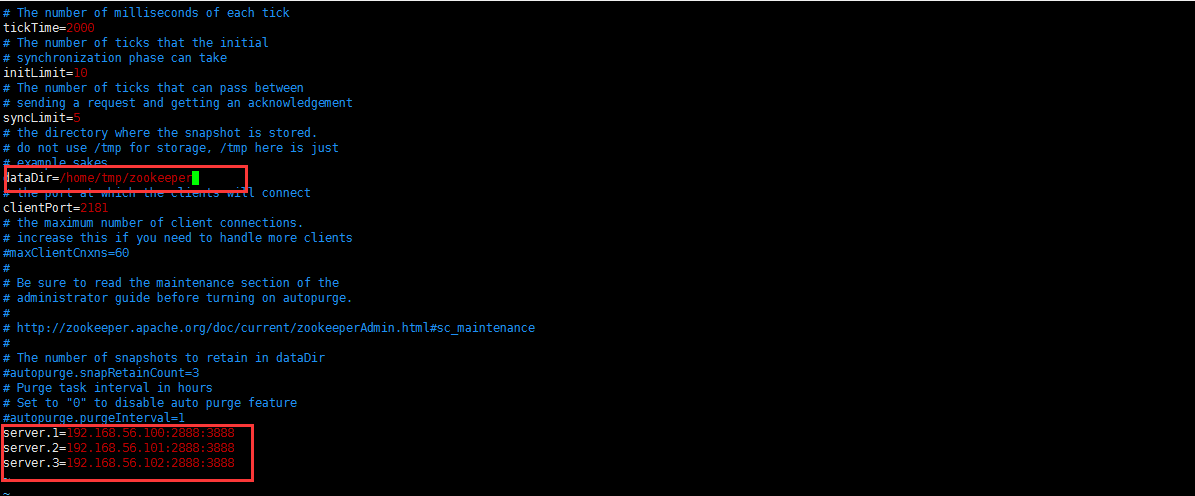
或者：

server.1=slave1:2888:3888

server.2=slave2:2888:3888

server.3=slave3:2888:3888

）



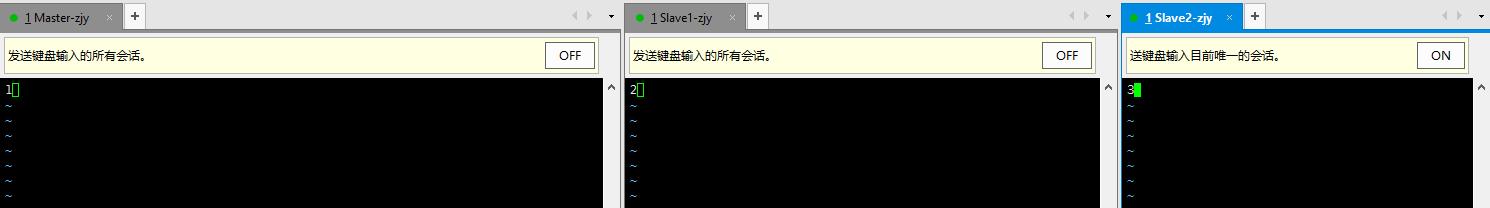
4、在/var/zjy/zk建立目录（与之前配置文件中dataDir一致），新建myid文件并且依次写入“1”、“2”、“3”

cd /var

mkdir zjy

mkdir zk

vim myid



5、配置各个组件的环境变量

cd /etc

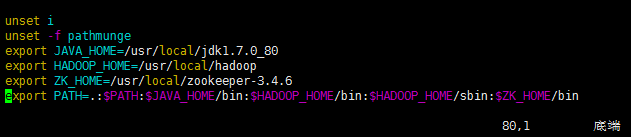
vim profile

export JAVA\_HOME=/usr/local/jdk1.7.0\_80

export HADOOP\_HOME=/usr/local/hadoop

export ZK\_HOME=/usr/local/zookeeper-3.4.6

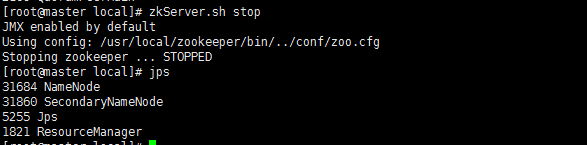
PATH=.:$PATH:$JAVA\_HOME/bin:$HADOOP\_HOME/bin:$HADOOP\_HOME/sbin:$ZK\_HOME/bin



1. 启动与关闭(注意在master/slave1/2/3/上依次启动或者关闭)

zkServer.sh start

zkServer.sh stop

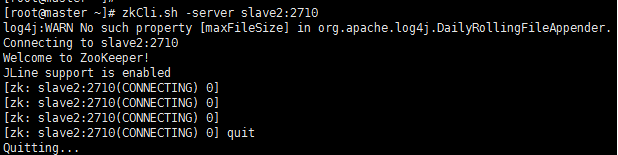


1. 验证状态和通讯

zkServer.sh start

一个leader3个follower

zkCli.sh -server slave2:2710 quit



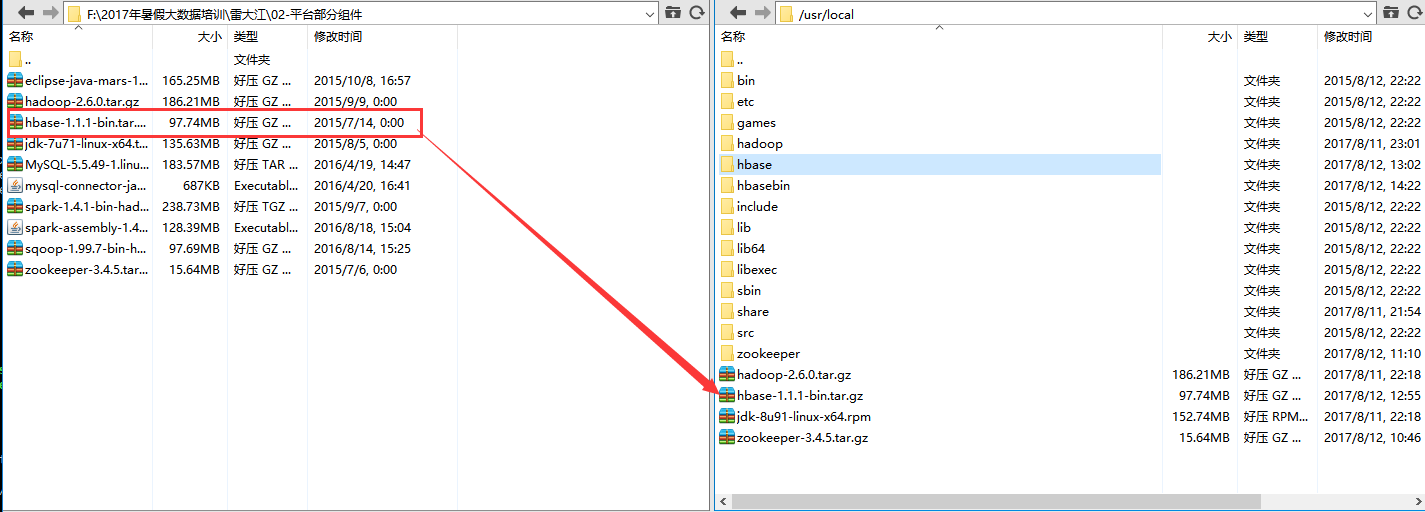
## **1.2、Hbase的**配置与验证（16010）

1、用Xshell和Xftp将下载好的Hbase复制到master的usr/local/目录下。

cd /usr/local

tar -xvf hbase-1.1.1-bin.tar.gz

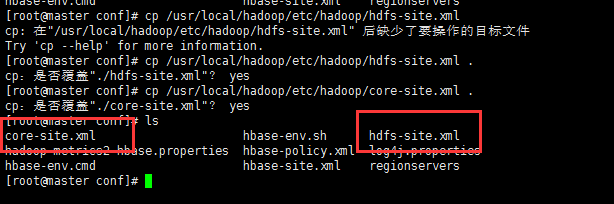
mv hbase-1.1.1 hbase



2、将Hadoop/etc/hadoop下的配置文件hfdfs-sie.xml和core-site.xml复制到Hbase的conf目录下 。

cp /usr/local/hadoop/etc/hadoop/core-site.xml .

cp /usr/local/hadoop/etc/hadoop/hdfs-site.xml .

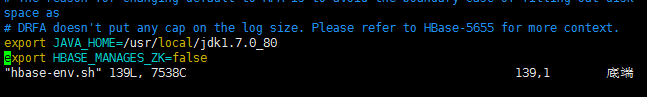


3、配置hbase-env.sh文件

vim hbase-env.sh

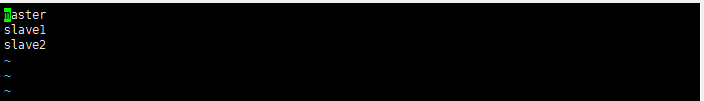
export JAVA\_HOME=/usr/java/jdk1.7.0\_80

export HBASE\_MANAGES\_ZK=false



4、在regionservers文件里写入所有的slave机器

vim regionservers



5、配置hbase-site.xml文件

vim hbase-site.xml

<configuration>

<property>

<name>hbase.rootdir</name>

<value>hdfs://master:9000/hbase</value>

</property>

<property>

<name>hbase.cluster.distributed</name>

<value>true</value>

</property>

<property>

<name>hbase.zookeeper.quorum</name>

<value>slave1,slave2</value>

</property>

<property>

<name>hbase.zookeeper.property.dataDir</name>

<value>/home/tmp2/hbase/data</value>

</property>

<property>

<name>hbase.master</name>

<value>hdfs://master:60000</value>

</property>

<property>

<name>hbase.zookeeper.property.clientPort</name>

<value>2181</value>

</property>

<property>

<name>hbase.master.info.port</name>

<value>16010</value>

</property>

<property>

<name>hbase.regionserver.info.port</name>

<value>16030</value>

</property>

</configuration>

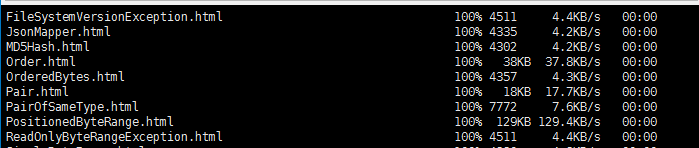
5、将Hbase文件夹传送到各个slave节点上

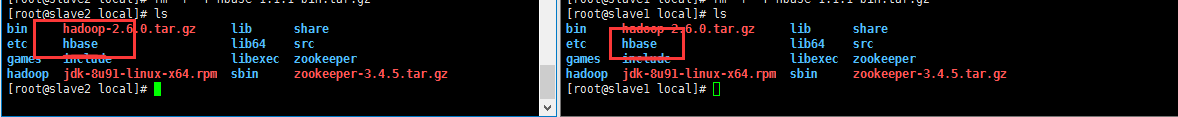
cd /usr/local

scp -r hbase slave1:/usr/local/

scp -r hbase slave2:/usr/local/

scp -r hbase slave3:/usr/local/





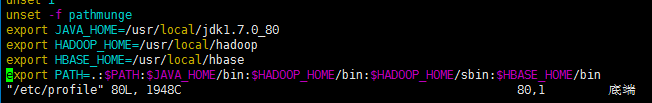
6、配置Hbase环境变量，Master和两台slave

vim /etc/profile

export HBASE\_HOME=/usr/local/hbase

export PATH=$PATH:$HBASE\_HOME/bin

重启source /etc/profile

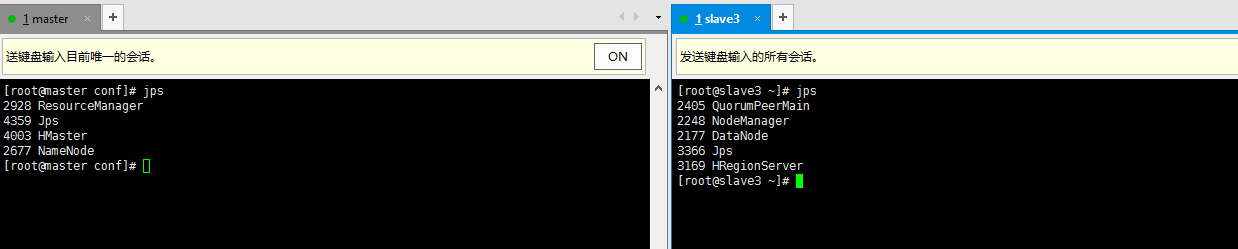


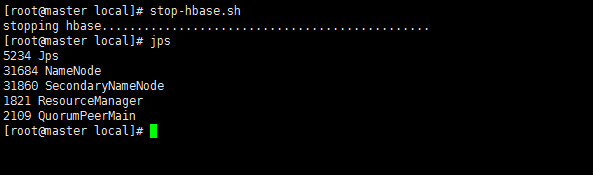
7、启动与关闭Hbase

start-hbase.sh

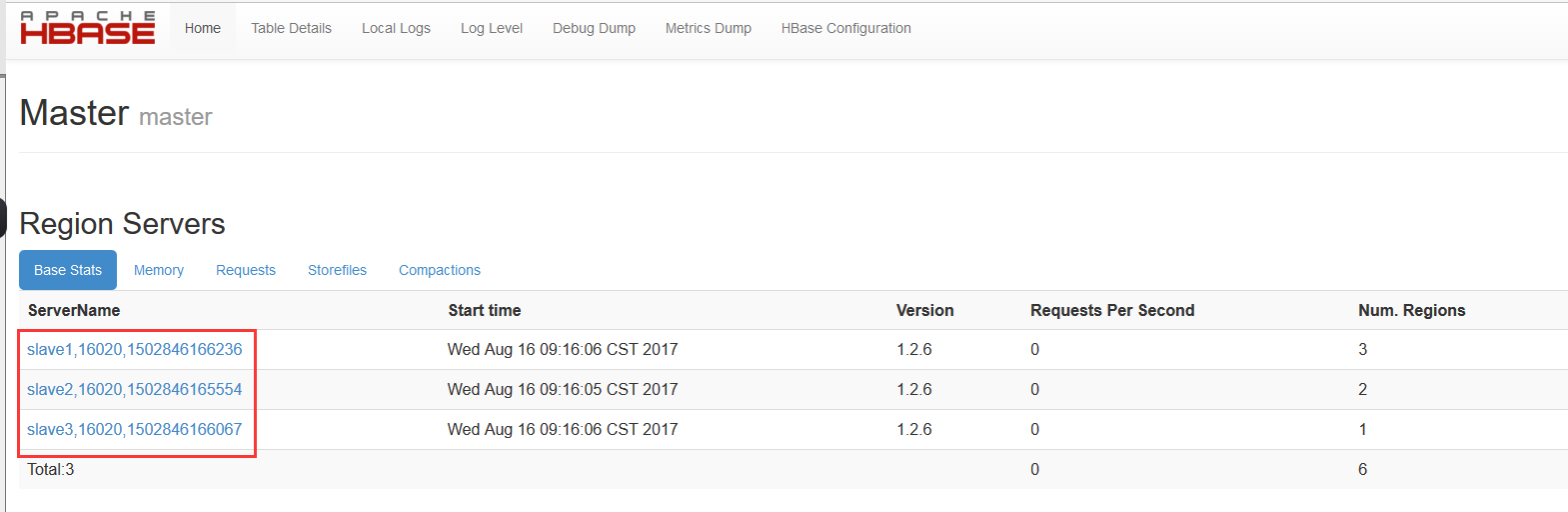
stop-hbase.sh







8、web端口查看192.168.56.100:16010



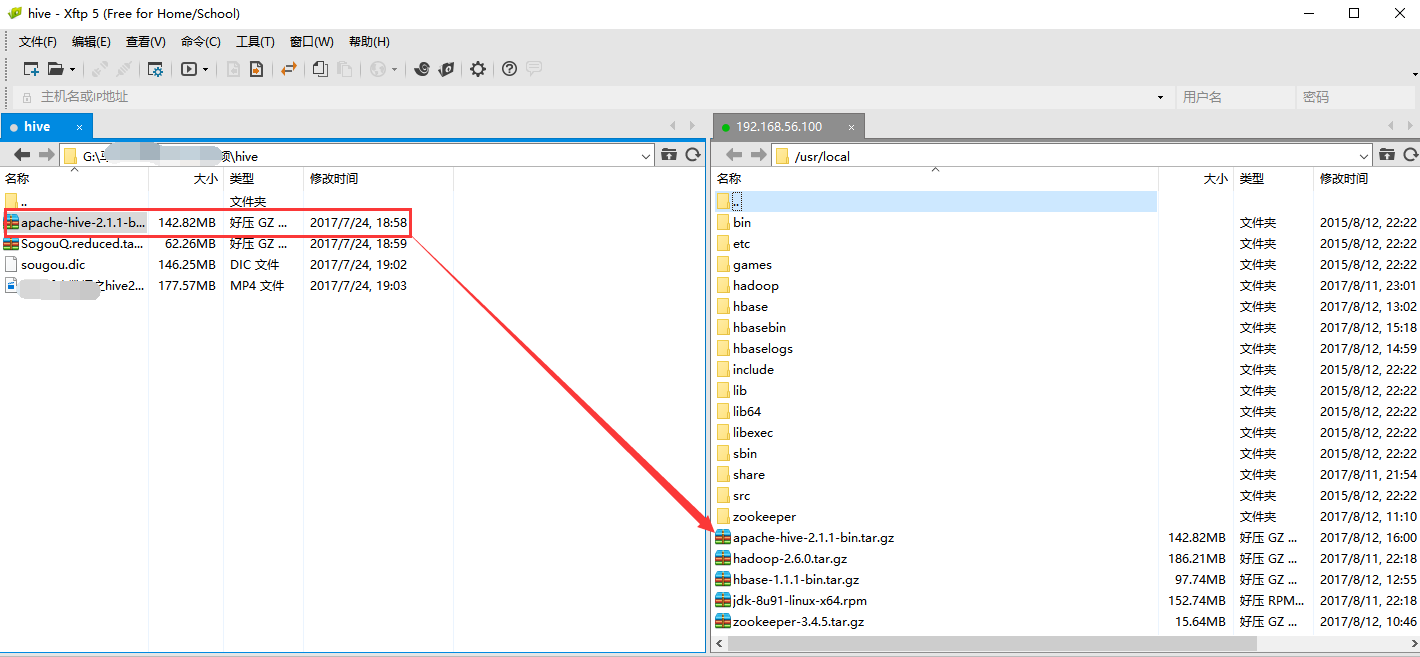
## **1.3、Hive的**配置与验证

1、用Xshell和Xftp将下载好的Hive复制到master的usr/local/目录下。

cd /usr/local

tar -xvf apache-hive-2.1.1-bin.tar.gz

mv apache-hive-2.1.1-bin hive



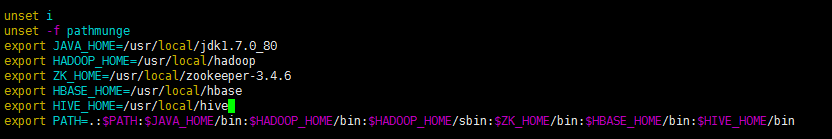
2、更改Hive配置环境(三台同时)

vim /etc/profile

export HIVE\_HOME=/usr/local/hive

export PATH=$PATH:$HIVE\_HOME/bin

source /etc/profile



3 、配置hive-site.xml文件

cd /usr/local/hive/conf

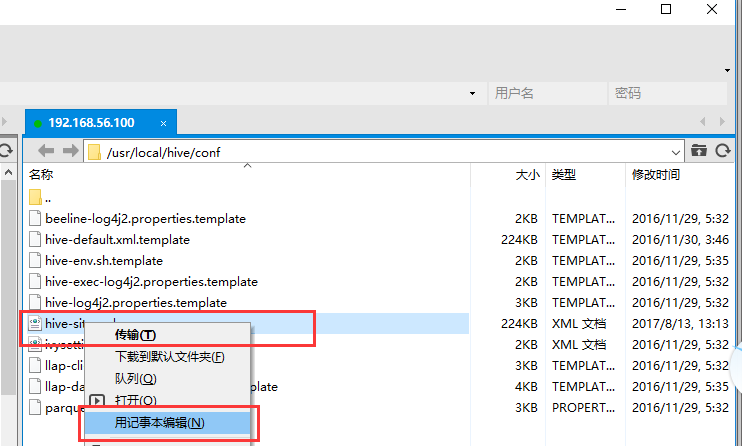
cp hive-default.xml.template hive-site.xml

修改hive-site.xml，可在xftp下修改，如图所示。

修改hive.metastore.schema.verification，设定为false

创建/usr/local/hive/tmp目录，并替换${system:java.io.tmpdir}为该目录

替换${system:user.name}为root

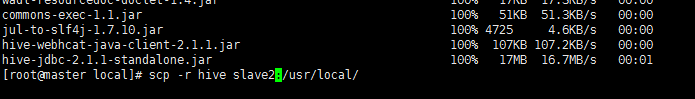


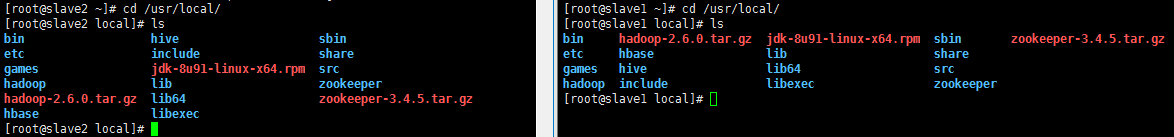
4、将Hbase文件夹传送到各个slave节点上

cd /usr/local

scp -r hive slave1:/usr/local/

scp -r hive slave2:/usr/local/





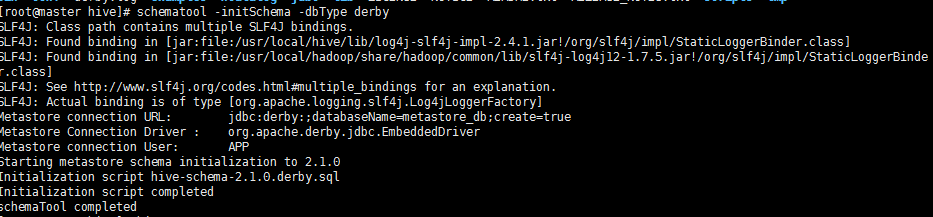
5、格式化derby数据库

schematool -initSchema -dbType derby

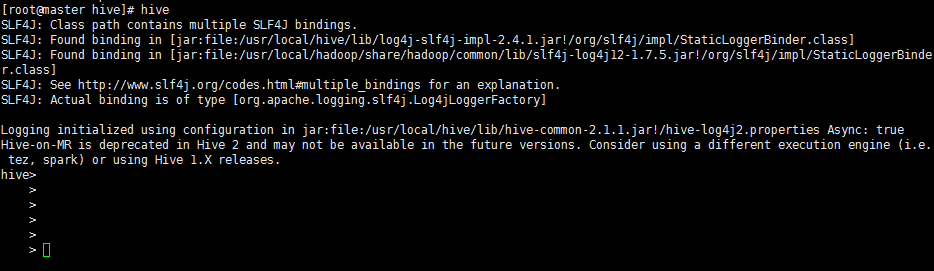
会在当前目录下简历metastore\_db的数据库。

注意！！！下次执行hive时应该还在同一目录，默认到当前目录下寻找metastore。

遇到问题，把metastore\_db删掉，重新执行命令。



5启动与关闭Hive



注：将mysql的jbdc驱动放到hive的lib目录下，并且修改hive-site.xml文件，将hive的默认数据库换成mysql。

1. 首先进入/usr/local/hive目录下将metastore\_db删除。
2. 修改hive-site.xml文件。

<property>

<name>javax.jdo.option.ConnectionURL</name>

<value>jdbc:mysql://192.168.56.100:3306/hive?createDatabaseIfNotExist=true</value>

<property>

<property>

<name>javax.jdo.option.ConnectionDriverName</name>

<value>com.mysql.jdbc.Driver</value>

<property>

<property>

<name>javax.jdo.option.ConnectionUserName</name>

<value>root</value>

<property>

<property>

<name>javax.jdo.option.ConnectionPassword</name>

<value>123456</value>

<property>

1. 关闭重启mysql

service mysqld restart

4、在hive下格式化mysql数据库

schematool -initSchema -dbType mysql

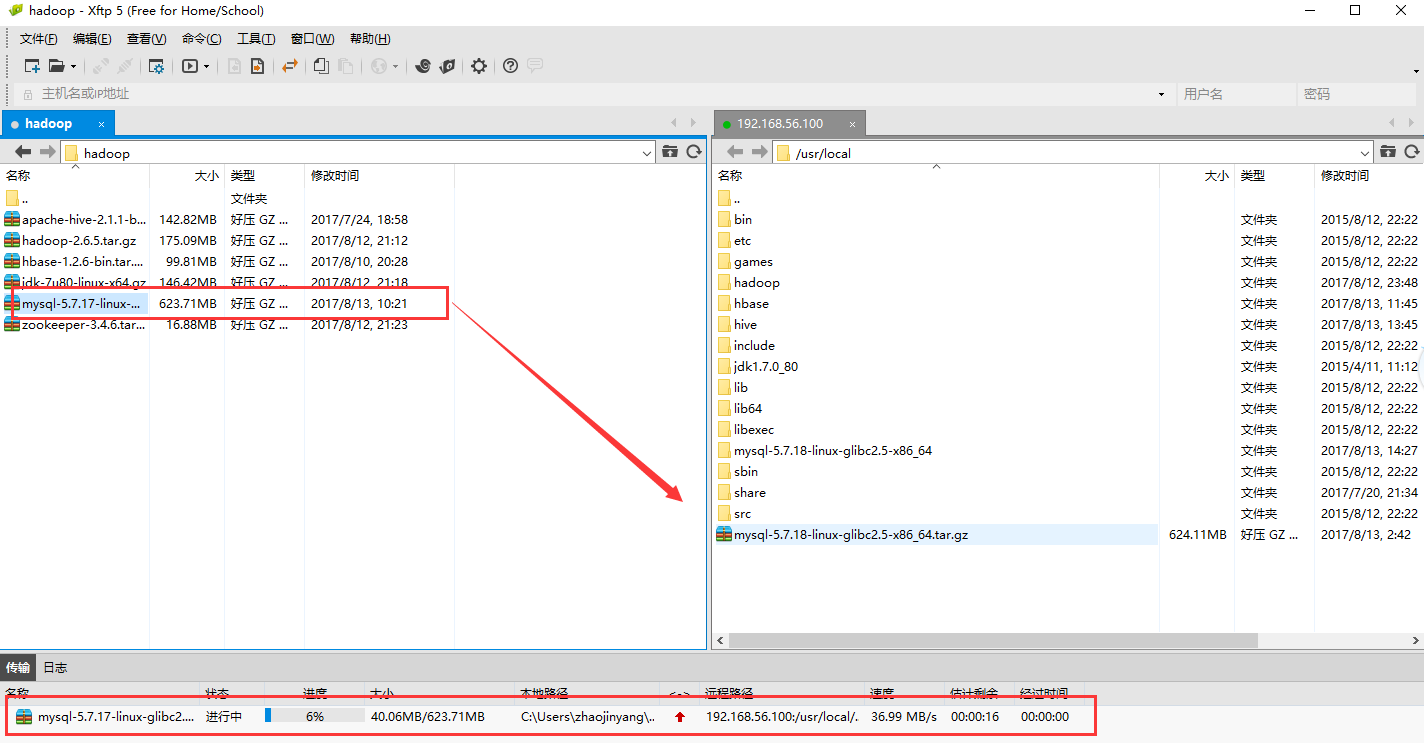


5、 在hive下输入hive如果进入hive则成功。

## 1.4、MySql配置与验证(3306)

1、用Xshell和Xftp将下载好的MySql复制到master的usr/local/目录下。

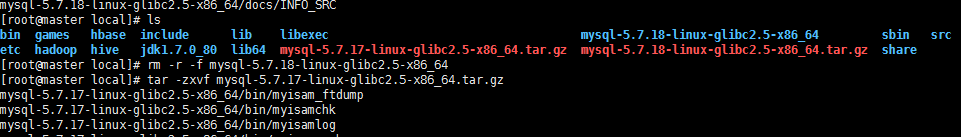
cd /usr/local



2、解压安装包并将解压包里的内容拷贝到mysql的安装目录/home/mysql

tar -zxvf mysql-5.7.17-linux-glibc2.5-x86\_64.tar.gz

mv mysql-5.7.17-linux-glibc2.5-x86\_64 mysql



3、创建data目录

mkdir /usr/local/mysql/data



4、创建mysql的用户组/用户

groupadd mysql

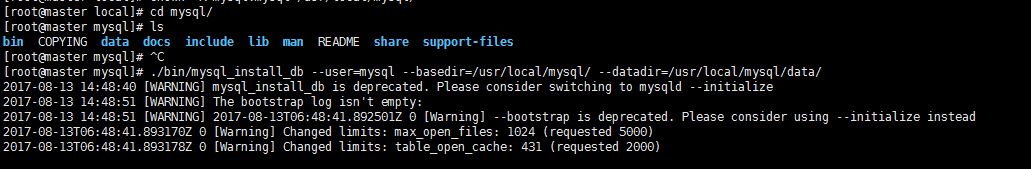
useradd -g mysql mysql

chown -R mysql.mysql /usr/local/mysql/

cd mysql/

5、初始化mysql数据库

./bin/mysql\_install\_db --user=mysql --basedir=/usr/local/mysql/ --datadir=/usr/local/mysql/data/

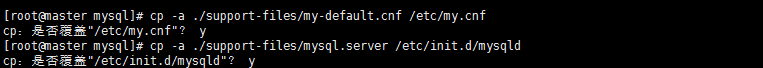


6、复制配置文件到 /etc/my.cnf

cp -a ./support-files/my-default.cnf /etc/my.cnf (选择y)

7、mysql的服务脚本放到系统服务中(选择y)

cp -a ./support-files/mysql.server /etc/init.d/mysqld



8、mysql的服务脚本放到系统服务中

cp -a ./support-files/mysql.server /etc/init.d/mysqld

9、进入vim /etc/my.cnf修改下面

basedir = /usr/local/mysql

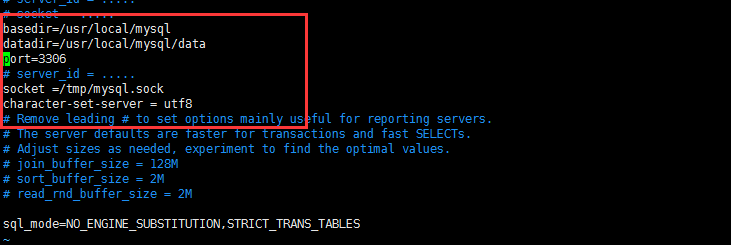
datadir = /usr/local/mysql/data

port = 3306

# server\_id = .....

socket = /tmp/mysql.sock

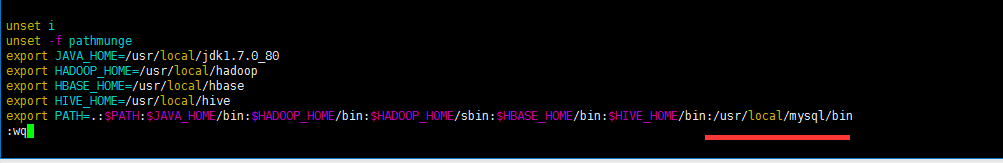
character-set-server = utf8



1. 配置mysql环境变量

vim /etc/profile

. /etc/profile



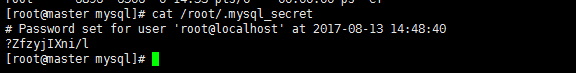
10、启动mysql

service mysqld start



11、查看mysql,记住密码，后边改密码需要

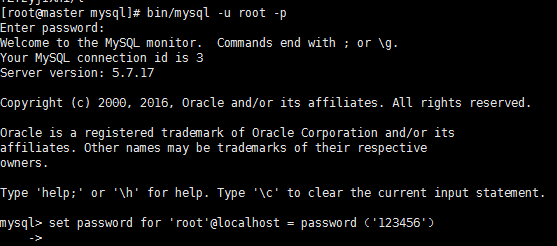
cat /root/.mysql\_secret



12、登录mysql

bin/mysql -u root -p

set password for 'root'@localhost = password（'123456'）;(自己手输入)



退出：exit()

## **1.5、Scala**配置与验证

1、用Xshell和Xftp将下载好的Scala复制到master的usr/local/目录下。

cd /usr/local解压并文件名

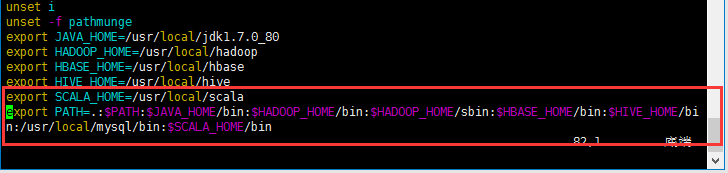
tar -xvf scala-2.10.6.tgz

mv scala-2.10.6 scala

2、配置Scale环境变量

vim /etc/profile

. /etc/profile



3、将Scala文件夹传送到各个slave节点上

cd /usr/local

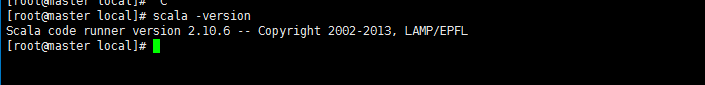
scp -r hive slave1:/usr/local/

scp -r hive slave2:/usr/local/

scp -r hive slave3:/usr/local/

4、如果打印出如下版本信息，则说明安装成功

scala -version



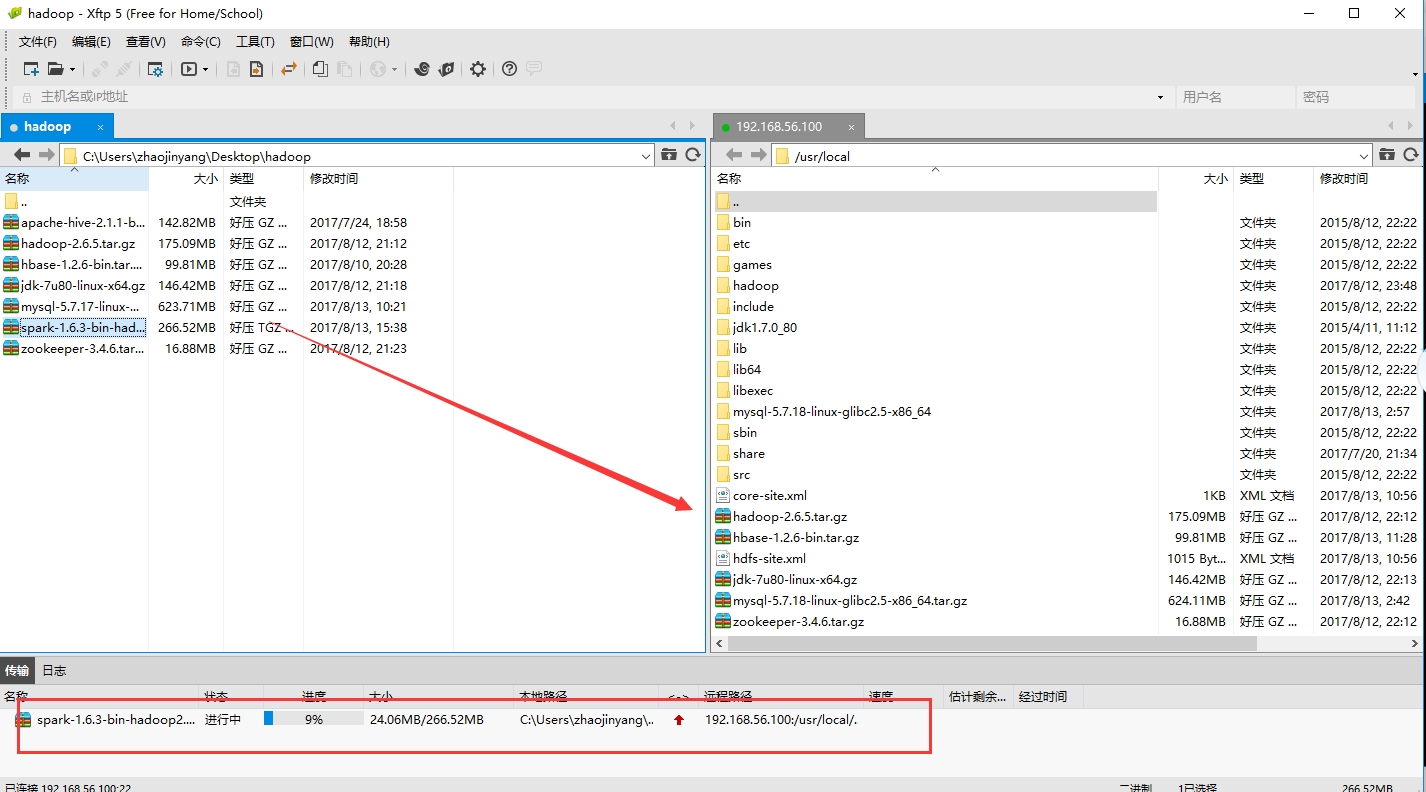
## **1.6、Spark的**配置与验证（4040,7077）

1、用Xshell和Xftp将下载好的Scala复制到master的usr/local/目录下。

cd /usr/local解压并文件名

tar -xvf spark-1.6.3-bin-hadoop2.6.tgz

mv spark-1.6.3-bin-hadoop2.6 spark



2、配置spark-env.sh文件

cd /usr/local/spark/conf

vim spark-env.sh

export JAVA\_HOME=/usr/local/jdk1.7.0\_80

export HADOOP\_HOME=/usr/local/hadoop

export SCALA\_HOME=/usr/local/scala

export SPARK\_HOME=/usr/local/spark

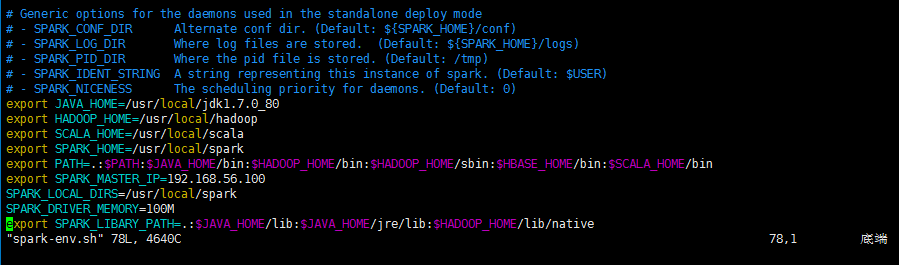
export PATH=.:$PATH:$JAVA\_HOME/bin:$HADOOP\_HOME/bin:$HADOOP\_HOME/sbin:$HBASE\_HOME/bin:$SCALA\_HOME/bin

export SPARK\_MASTER\_IP=192.168.56.100

SPARK\_LOCAL\_DIRS=/usr/local/spark

SPARK\_DRIVER\_MEMORY=100M

export SPARK\_LIBARY\_PATH=.:$JAVA\_HOME/lib:$JAVA\_HOME/jre/lib:$HADOOP\_HOME/lib/native



4、将Spark文件夹传送到各个slave节点上

cd /usr/local

scp -r spark slave1:/usr/local/

scp -r spark slave2:/usr/local/

scp -r spark slave3:/usr/local/

4、配置各个组件的环境变量

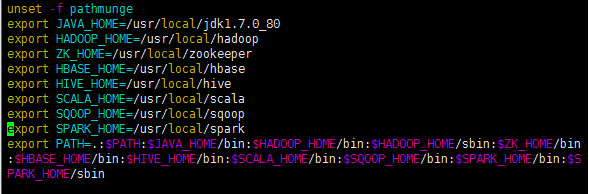
cd /etc

vim profile

export SPARK\_HOME=/usr/local/spark

PATH=.:$PATH:$JAVA\_HOME/bin:$HADOOP\_HOME/bin:$HADOOP\_HOME/sbin:$ZK\_HOME/bin:$HBASE\_HOME/bin:$HIVE\_HOME/bin:$SCALA\_HOME/bin:$SQOOP\_HOME/bin:$SPARK\_HOME/bin:$SPARK\_HOME/sbin

最后记得重新启动: . /etc/profile



5、从master节点进入到/usr/local/spark/sbin启动Spark

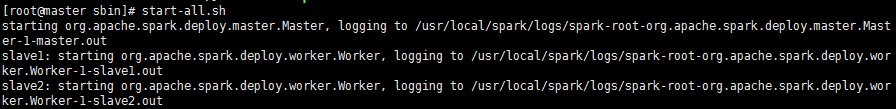
cd /usr/local/spark/sbin

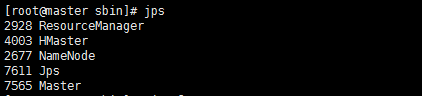
开启：

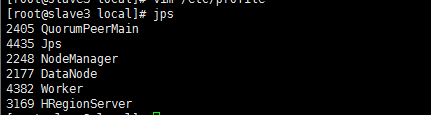
start-all.sh

关闭：

stop-all.sh







## **1.7、Sqoop的**配置与验证

1、用Xshell和Xftp将下载好的Sqoop复制到master的usr/local/目录下。

cd /usr/local解压并更改文件名

tar -xvf sqoop-1.4.3.bin\_\_hadoop-1.0.0.tar.gz

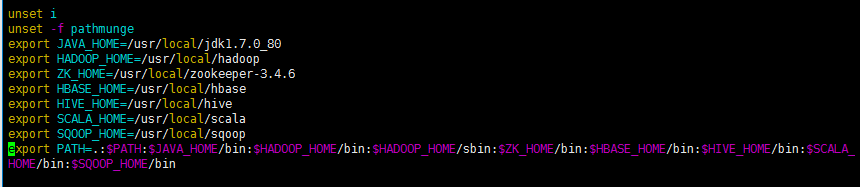
mv sqoop-1.4.3.bin\_\_hadoop-1.0.0 sqoop

2、配置Sqoop环境变量

vim /etc/profile

export SQOOP\_HOME=/usr/local/sqoop

. /etc/profile



3、把mysql的jdbc驱动mysql-connector-java-5.1.10.jar复制到sqoop项目的lib目录下。

4、配置sqoop-env.sh文件

cd /usr/local/sqoop/conf

cp sqoop-env-template.sh sqoop-env.sh

vim sqoop-env.sh

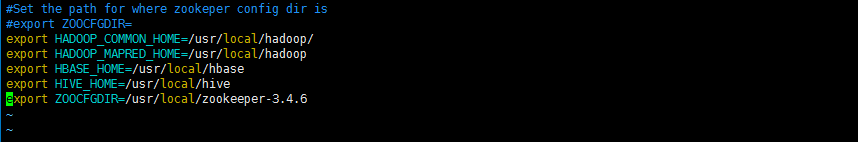
export HADOOP\_COMMON\_HOME=/usr/local/hadoop/

export HADOOP\_MAPRED\_HOME=/usr/local/hadoop

export HBASE\_HOME=/usr/local/hbase

export HIVE\_HOME=/usr/local/hive

export ZOOCFGDIR=/usr/local/zookeeper



5、将Sqoop文件夹传送到各个slave节点上

cd /usr/local

scp -r spark slave1:/usr/local/

scp -r spark slave2:/usr/local/

scp -r spark slave3:/usr/local/

1. 在slave下验证Sqoop

sqoop version

## **1.8、Hadoop集群启动原则**

启动：

（1）、先启动hdfs集群（在master上运行start-dfs.sh）

（2）、再启动Yarn集群（在master上运行start-yarn.sh）

（3）、 然后启动zookeeper集群（在slave每个机器上运行zkServer.sh start）

（4）、最后启动hbase（在master机器上运行start-hbase.sh）

停止：

（1）、先停止hbase（在master上stop-hbase.sh）

（2）、再停止zookeeper集群（在slave依次运行zkServer.sh stop）

（3）、然后停止Yarn集群（在master上运行stop-yarn.sh）

（3）、最后停止hdfs集群（在master上运行stop-dfs.sh）

hbase-daemons.sh stop regionserver

从master节点进入到/usr/local/spark/sbin启动Spark

开启：

start-all.sh

关闭：

stop-all.sh

Hive、Scala、Sqoop查看状态

# **ETL测试小数据集**

## **2.1、把数据从mysql导入到hdfs中**

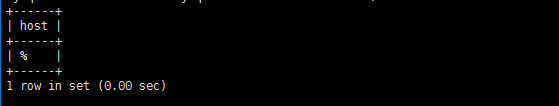
1、登录mysql设置host权限问题。

（1）、以权限用户root登录

mysql -u root -p

（2）、查看mysql库中的user表的host值（即可进行连接访问的主机/IP名称）

select host from mysql.user where user='root';



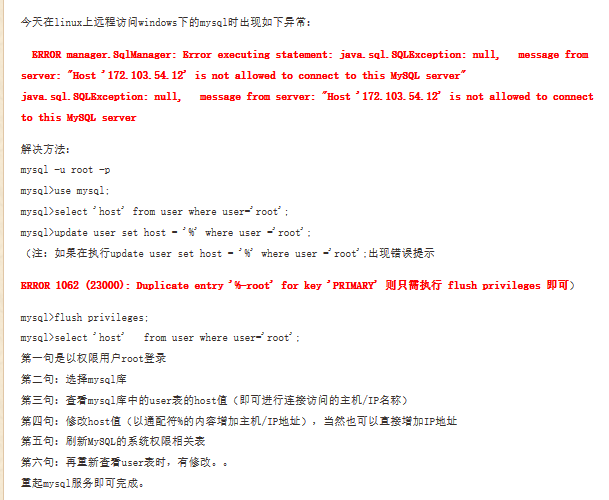
1. 、修改host值（以通配符%的内容增加主机/IP地址），当然也可以直接增加IP地址

update user set host = '%' where user ='root';

（4）、刷新MySQL的系统权限相关表

flush privileges;

1. 、重起mysql服务即可完成。
2. 、具体的可见下图



2、创建数据库、表

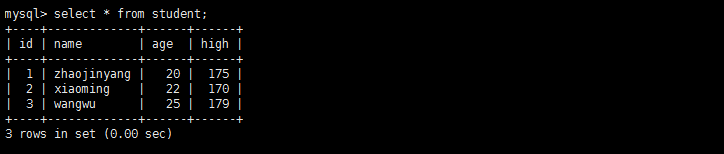
关于mysql使用具体看

http://www.runoob.com/mysql/mysql-drop-database.html

show databases;

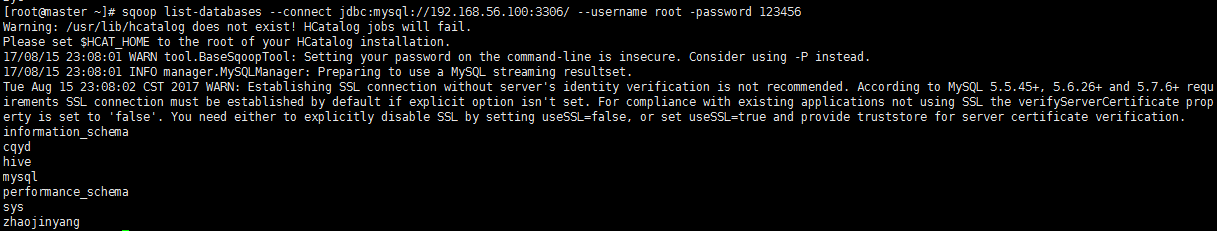
use cqyd;

select \* from student;



3、列出mysql数据库中的所有数据库

 sqoop list-databases --connect jdbc:mysql://192.168.56.100:3306/ --username root -password 123456



4、连接mysql并列出数据库中的表

sqoop list-tables --connect jdbc:mysql://localhost:3306/zhaojinyang --username root --password 123456

5、把数据从mysql导入到hdfs中

sqoop import --connect jdbc:mysql://192.168.56.100:3306/cqyd --username root --password 123456 --table student --target-dir /sqoop -m 1

sqoop ##sqoop命令

import ##表示导入

--connect jdbc:mysql://ip:3306/库名 ##告诉jdbc，连接mysql的url

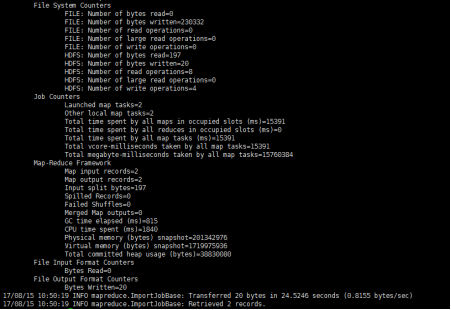
--username root ##连接mysql的用户名

--password 123456 ##连接mysql的密码

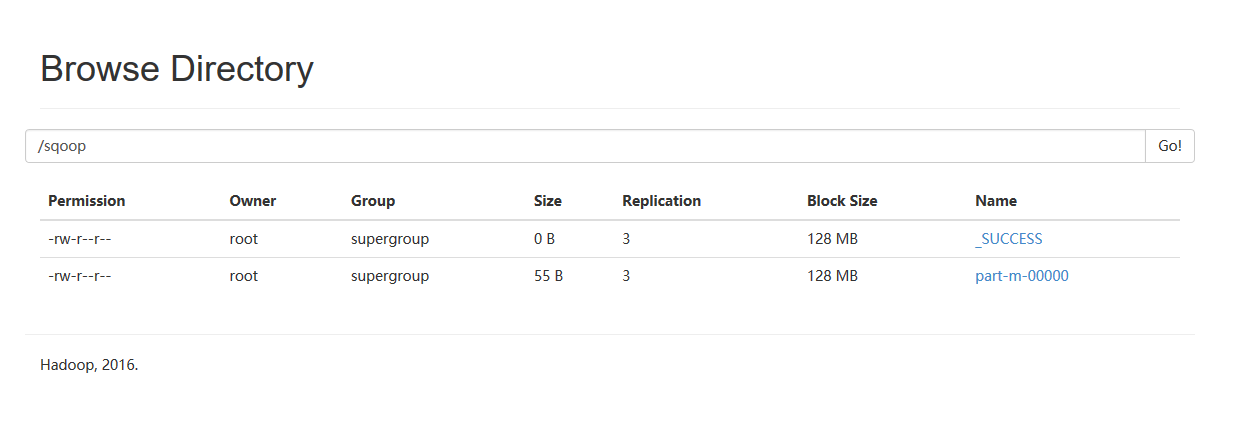
--table 表名 ##从mysql导出的表名称

--target-dir /sqoop ##指定输出文件中的行的字段分隔符

-m 1 ##复制过程使用1个map作业



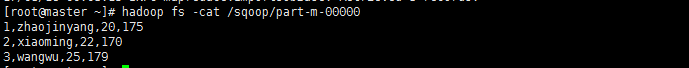
7、通过web页面观察上传结果



8、通过hadoop平台查看结果

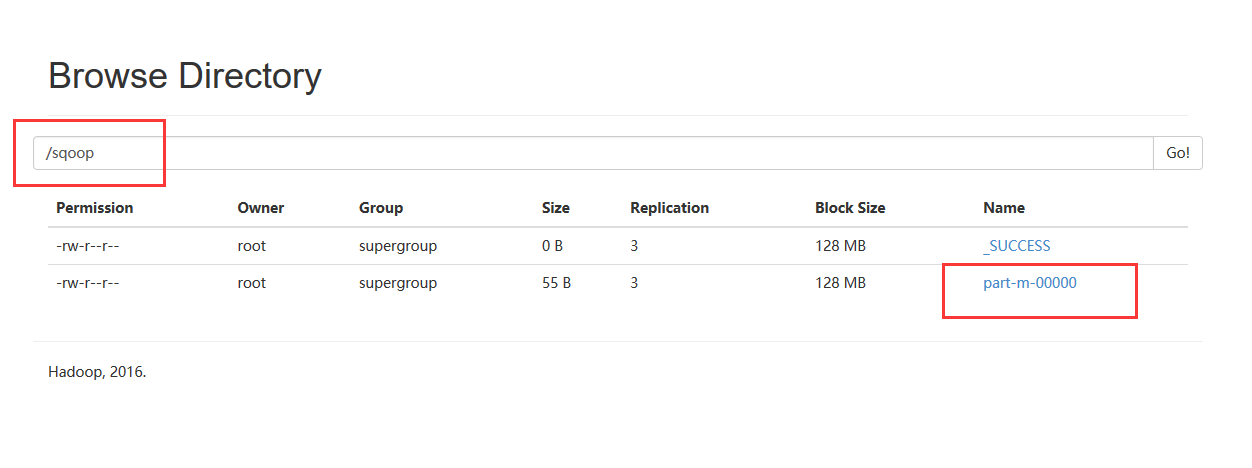
hadoop fs -cat /sqoop1/part-m-00000

hadoop fs -get /sqoop1 ./

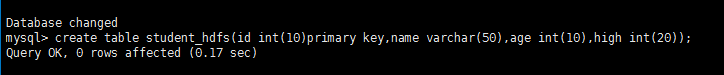


## **2.2、把数据从hdfs导出到mysql中**

1. 将传入到hdfs上的/sqoop/part-m-00000导入到mysql的student\_hdfs空表中，通过web页面观察/sqoop/part-m-00000文件。



1. 在mysql下建立空表student\_hdfs

create table student\_hdfs(id int(10)primary key,name varchar(50),age int(10),high int(20));

1. 通过sqoop把数据从hdfs导出到mysql中

sqoop export --connect jdbc:mysql://192.168.56.100:3306/cqyd --username root --password 123456 --table student\_hdfs --export-dir '/sqoop/part-m-00000' --driver com.mysql.jdbc.Driver

参数说明：

sqoop export ##表示数据从hdfs复制到mysql中

--connect jdbc:mysql://192.168.56.100:3306/zhaojinyang

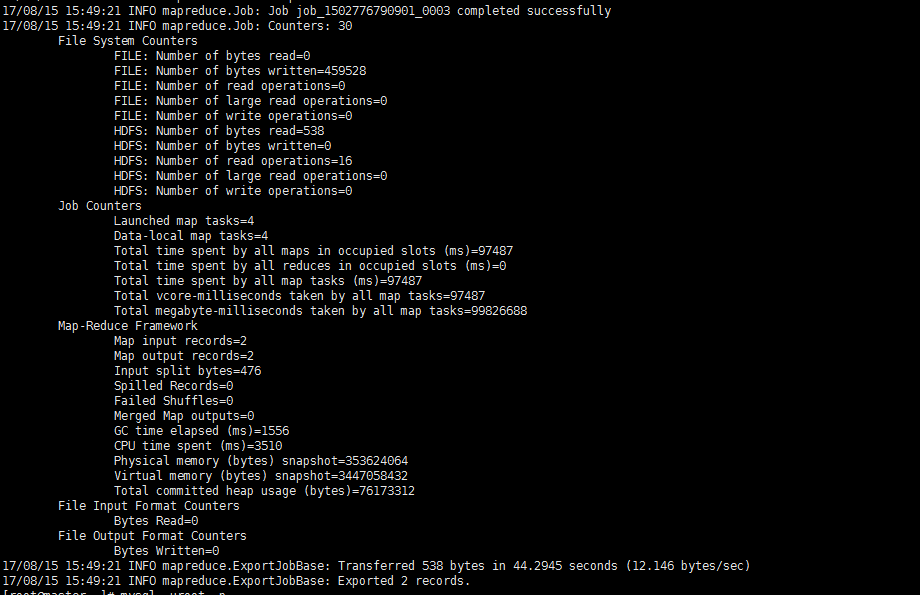
--username root

--password 123456

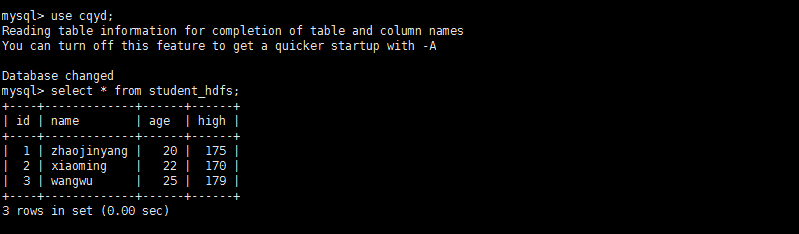
--table yangyang ##mysq中表，即将被导入的表名称

--export-dir '/sqoop1/part-m-00000'  ##hdfs中被导出的文件

--driver com.mysql.jdbc.Driver ##驱动



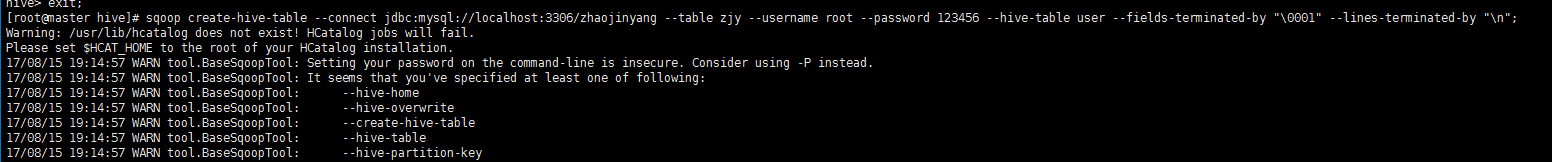
1. 进入mysql中观察结果。



## **2.3、将数据从关系数据库结构mysql导入文件到hive表中**

1. 将mysql里中的zjy表的结构导入到hive中。

sqoop create-hive-table --connect jdbc:mysql://localhost:3306/cqyd --table student --username root --password 123456 --hive-table student\_hive --fields-terminated-by "\t" ;



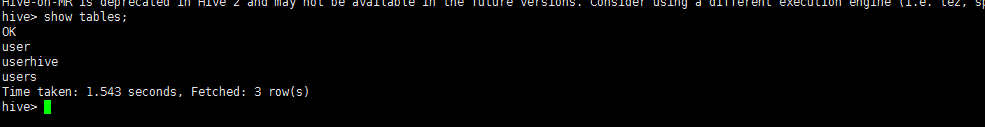
参数说明：

--fields-terminated-by "\t"  是设置每列之间的分隔符， 而sqoop的默认行内分隔符为"，"   
--lines-terminated-by "\n"  设置的是每行之间的分隔符，此处为换行符，也是默认的分隔符；

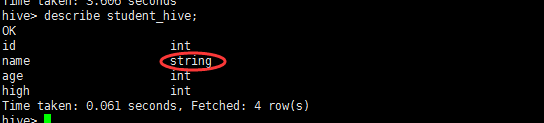
注意：只是复制表的结构，表中的内容没有复制

1. 进入hive查看结构。

show tables;



describe student\_hive;



## **2.4、将数据从关系数据库数据导入文件到hive表中**

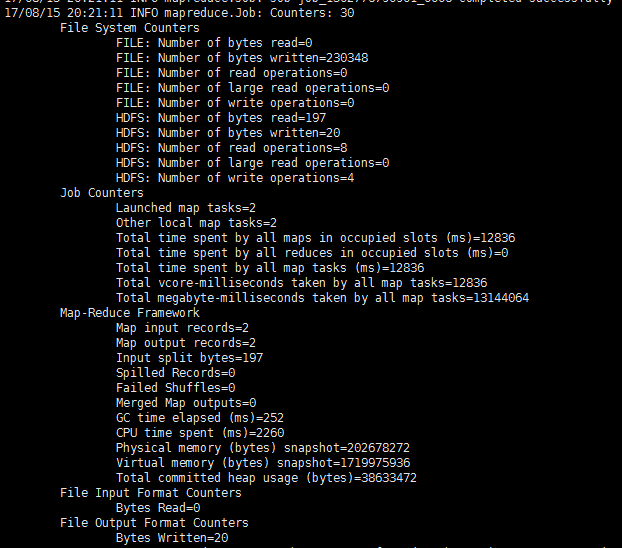
1. 将mysql里中的zjy表的内容导入到hive中。

sqoop import --connect jdbc:mysql://192.168.56.100:3306/cqyd --table student --username root --password 123456 --hive-import --hive-table student\_hive -m 1 --fields-terminated-by "\t";

参数说明：

-m 1 表示由两个map作业执行；

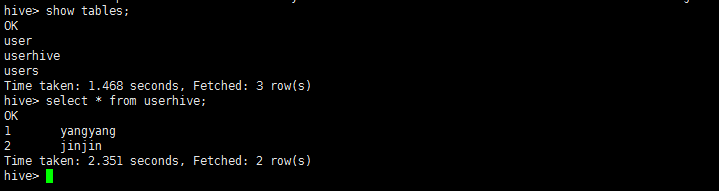
--fields-terminated-by "\t"  需同创建hive表时保持一致；



2、进入hive查看student\_hive。

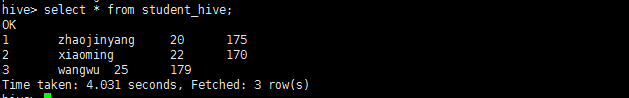
show tables;

select \* from student\_hive;



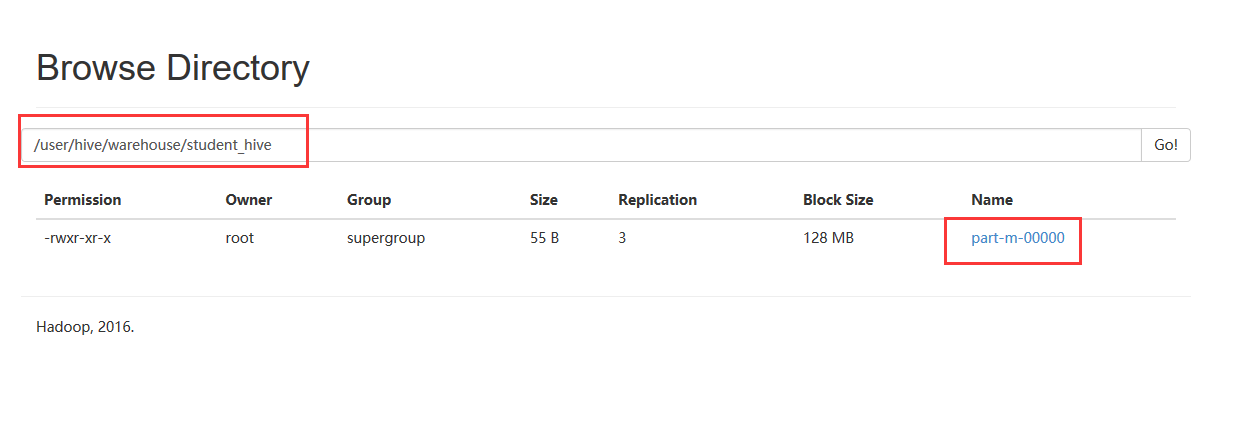
3、在hdfs上观察结果。

hadoop fs -cat /user/hive/warehouse/student\_hive/part-m-00000



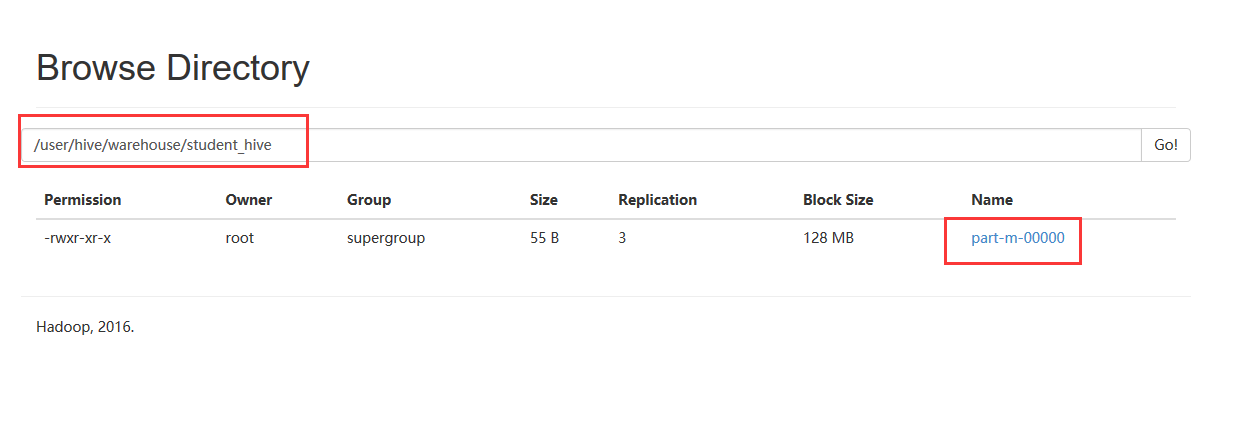
4、从mysql导入到hive中，在web观察。

将mysql里中的student表的内容导入到hive中表保存在/user/hive/warehouse/student\_hive/part-m-00000

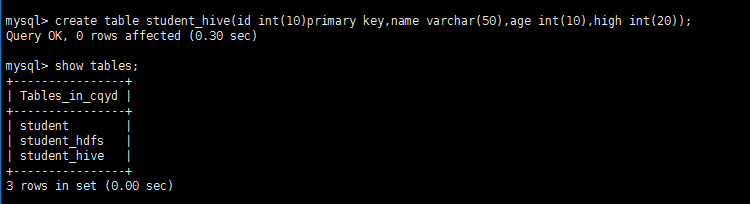


## **2.5、把数据从hive导出到mysql中**

1. 将传入到hive上的/user/hive/warehouse/student\_hive/part-m-00000中导入到mysql的yangyang空表中，通过web页面观察/user/hive/warehouse/student\_hive/part-m-00000文件。



2、在mysql下建立空表student\_hive

create table student\_hive(id int(10)primary key,name varchar(50),age int(10),high int(20));

3、通过sqoop把数据从hive导出到mysql中

sqoop export --connect jdbc:mysql://192.168.56.100:3306/cqyd --username root --password 123456 --table student\_hive --export-dir '/user/hive/warehouse/student\_hive/part-m-00000' --input-fields-terminated-by "\t"

参数解释：

sqoop export ##表示数据从hdfs复制到mysql中

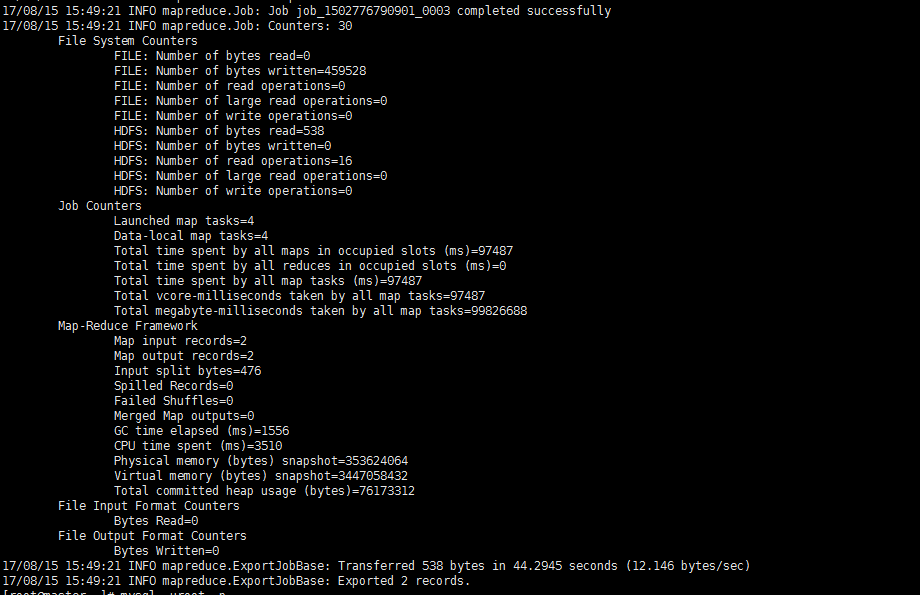
--connect jdbc:mysql://192.168.56.100:3306/zhaojinyang

--username root

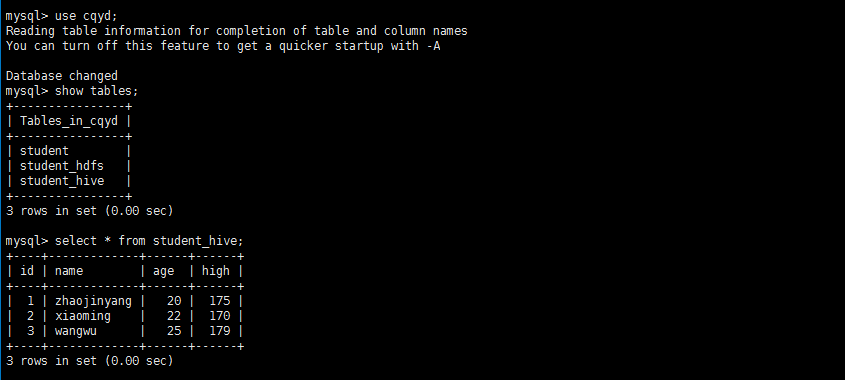
--password 123456

--table yangyang ##mysq中表，即将被导入的表名称

--export-dir '/user/hive/warehouse/beijing/part-m-00000'  ##hdfs中被导出的文件



4、进入mysql中观察结果。



## **2.6、把数据从mysql导出到hbase中**

1. 首先在hbase中建立表student-hbase2

create ‘student-hbase2’,’row’

1. 把数据从mysql导出到hbase中

sqoop import --connect jdbc:mysql://192.168.56.100:3306/cqyd --table student --hbase-table student-hbase2 --column-family row --hbase-row-key id --hbase-create-table --username root --password 123456

参数解释：

--connect jdbc:mysql://192.168.56.100:3306/cqyd 表示远程或者本地 Mysql 服务的URI，3306是Mysql默认监听端口，cqyd是数据库，若是其他数据库，如Oracle,只需修改URI即可。

--table student 表示导出cqyd数据库的student表。

--hbase-table student-hbase2 表示在HBase中建立表student-hbase2。

--column-family row 表示在表A中建立列族row。

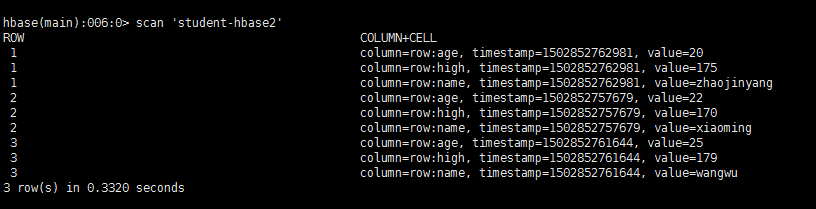
--hbase-row-key id 表示表A的row-key是student表的id字段。

--hbase-create-table 表示在HBase中建立表。

--username root 表示使用用户root连接Mysql。

IMG_256

3、在hbase下观察结果。



4、mys和hbase表转换互看

mysql中的表

|  |  |  |
| --- | --- | --- |
| Id | name | age |
|  |  |  |
|  |  |  |

Hbase中的表

|  |  |  |
| --- | --- | --- |
| Row key | row | |
|  | name | id |
|  |  |  |
|  |  |  |

## **2.7、hdfs数据加载到hive**

1、进入hive创建表结构student

create database cqyd;

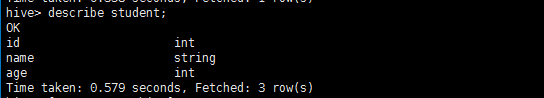
show databases;

use cqyd;

create table student(id string,name String,age string);

显示表结构

describe student;



1. 将student数据放入到hadoop/hive目录下。

hadoop fs -put hdfs\_hive.txt /hive

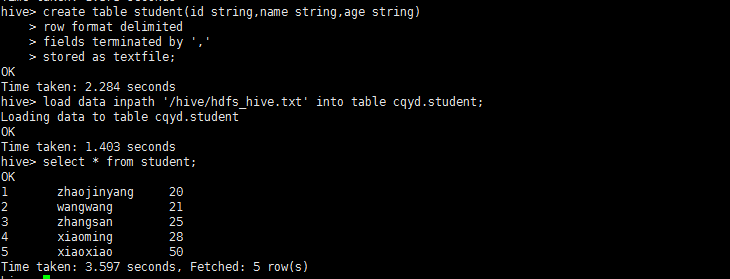


3、将hdfs的hive目录下hdfs\_hive.txt加载到hive中，是数据的转移不是复制（内表）

load data inpath '/hive/hdfs\_hive.txt' into table cqyd.student;

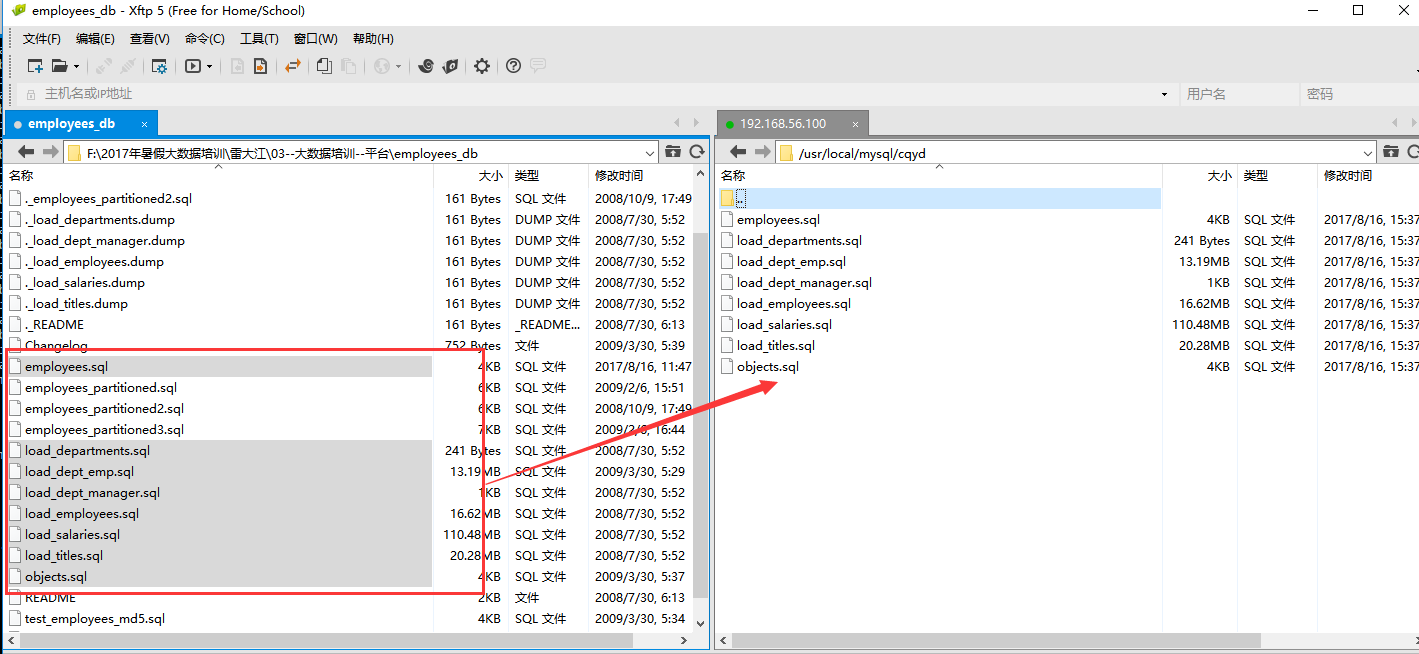
4、使用hive命令查看student表的内容

select \* from student;



# **完成数据ETL案例**

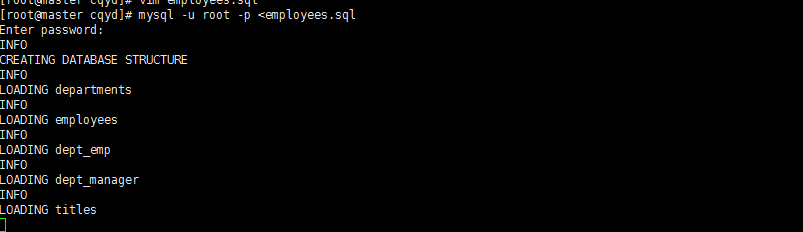
## **3.1、将employees\_db文件传到mysql下的cqyd文件夹下**



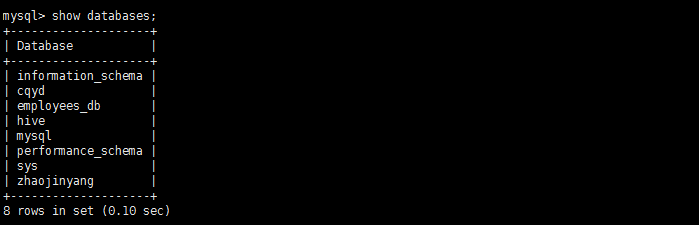
## **3.2、运行生成数据库**

在cqyd目录下执行

mysql -u root -p <employees.sql



进入mysql观察结果。

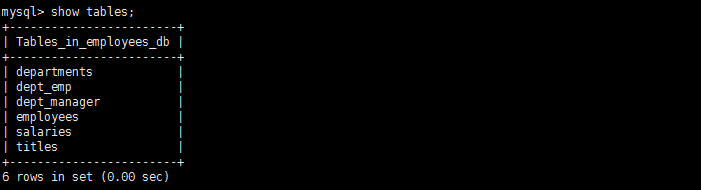


## **3.3、数据迁移**

### 3.3.1、把数据从mysql导入到hdfs中

1. 把数据从mysql导入到hdfs中

将mysql中employees\_db中dept\_emp上传到hdfs的sqoop文件中



sqoop import --connect jdbc:mysql://192.168.56.100:3306/employees\_db

--username root --password 123456 --table dept\_emp --target-dir /sqoop -m 1

sqoop ##sqoop命令

import ##表示导入

--connect jdbc:mysql://ip:3306/库名 ##告诉jdbc，连接mysql的url

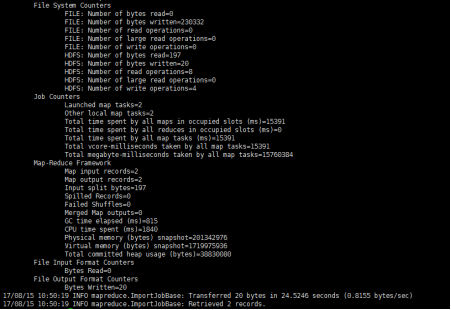
--username root ##连接mysql的用户名

--password 123456 ##连接mysql的密码

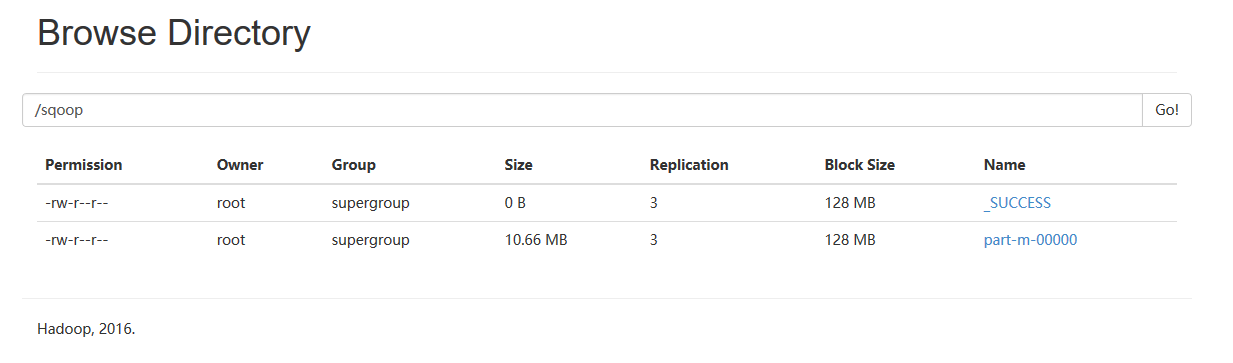
--table 表名 ##从mysql导出的表名称

--target-dir /sqoop ##指定输出文件中的行的字段分隔符

-m 1 ##复制过程使用1个map作业



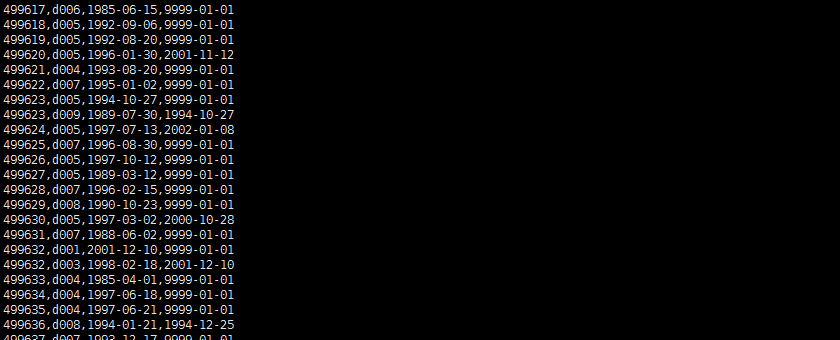
7、通过web页面观察上传结果



8、通过hadoop平台查看结果

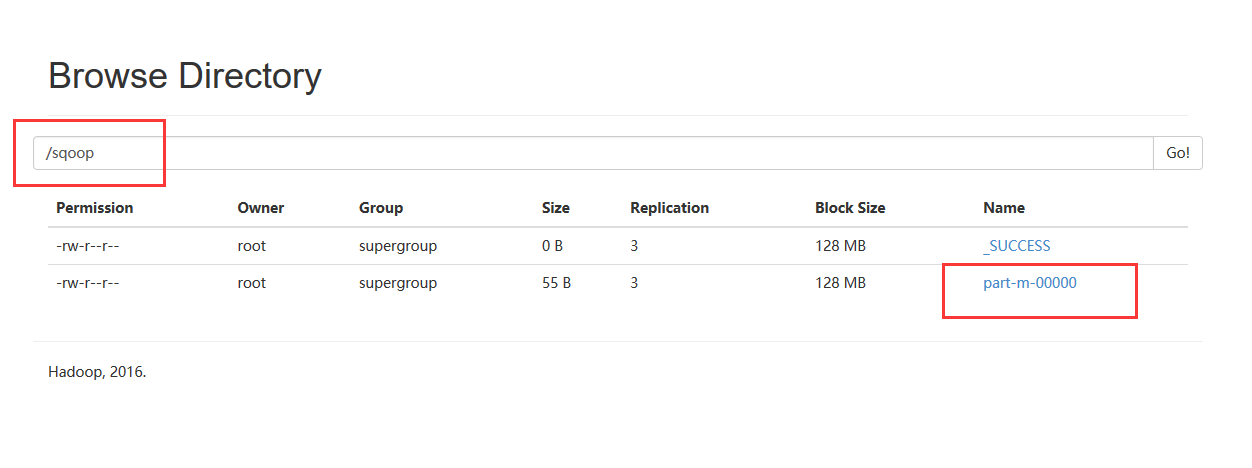
hadoop fs -cat /sqoop/part-m-00000

hadoop fs -get /sqoop ./



### **3.3.2、把数据从hdfs导出到mysql中**

1、将传入到hdfs上的/sqoop/part-m-00000导入到mysql的student\_hdfs空表中，通过web页面观察/sqoop/part-m-00000文件。



2、在mysql下建立空表student\_hdfs

create table employees\_hdfs(emp\_no int(10),dept\_no varchar(50),from\_date DATE,to\_date DATE);

3、通过sqoop把数据从hdfs导出到mysql中

sqoop export --connect jdbc:mysql://192.168.56.100:3306/cqyd --username root --password 123456 --table employees\_hdfs --export-dir '/sqoop/part-m-00000' --driver com.mysql.jdbc.Driver

参数说明：

sqoop export ##表示数据从hdfs复制到mysql中

--connect jdbc:mysql://192.168.56.100:3306/zhaojinyang

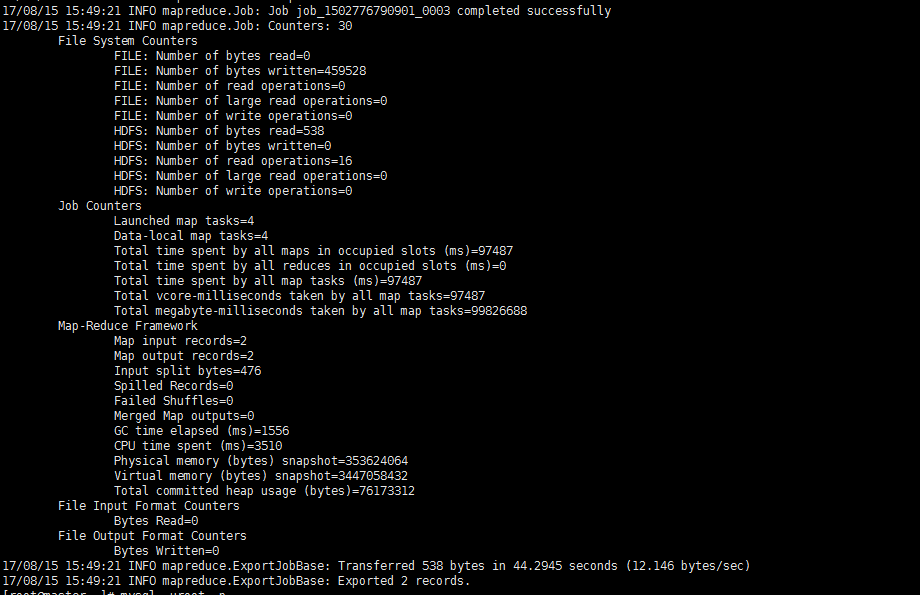
--username root

--password 123456

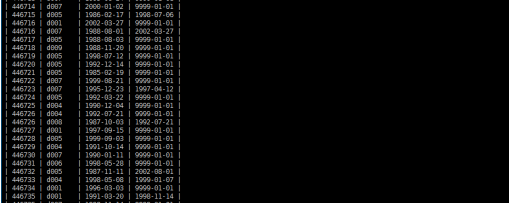
--table employees\_hdfs ##mysq中表，即将被导入的表名称

--export-dir '/sqoop/part-m-00000'  ##hdfs中被导出的文件

--driver com.mysql.jdbc.Driver ##驱动



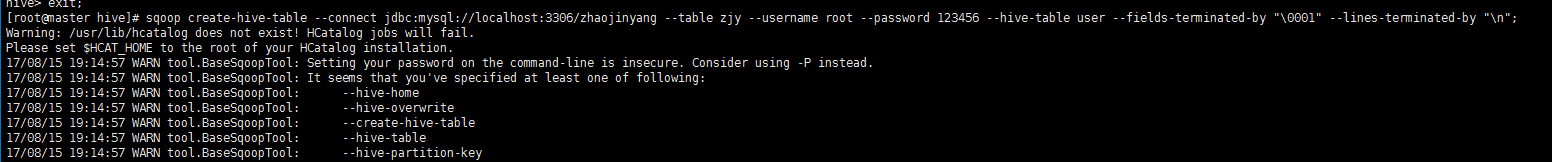
4、进入mysql中观察结果。



### **3.3.3、将数据从关系数据库结构mysql导入文件到hive表中**

1. 将mysql里中的zjy表的结构导入到hive中。

sqoop create-hive-table --connect jdbc:mysql://localhost:3306/employees\_db --table dept\_emp --username root --password 123456 --hive-table employees\_hive --fields-terminated-by "\t" ;



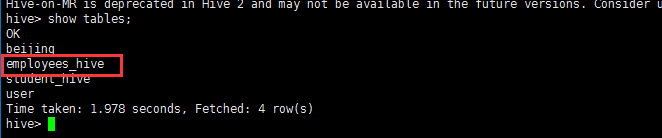
参数说明：

--fields-terminated-by "\0001"  是设置每列之间的分隔符，"\0001"是ASCII码中的1，它也是hive的默认行内分隔符， 而sqoop的默认行内分隔符为"，"   
--lines-terminated-by "\n"  设置的是每行之间的分隔符，此处为换行符，也是默认的分隔符；

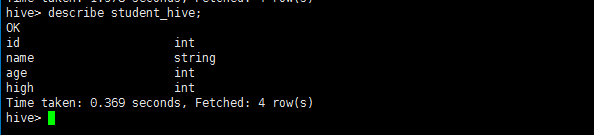
注意：只是复制表的结构，表中的内容没有复制

1. 进入hive查看结构。

show tables;



describe student\_hive;



### **3.3.4、将数据从关系数据库数据导入文件到hive表中**

1、将mysql里中的zjy表的内容导入到hive中。

sqoop import --connect jdbc:mysql://192.168.56.100:3306/employees\_db --table dept\_emp --username root --password 123456 --hive-import --hive-table employees\_hive -m 1 --fields-terminated-by "\t";

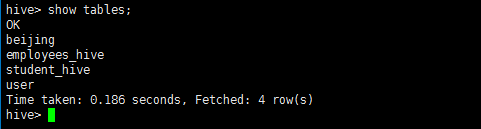
参数说明：

-m 1 表示由两个map作业执行；

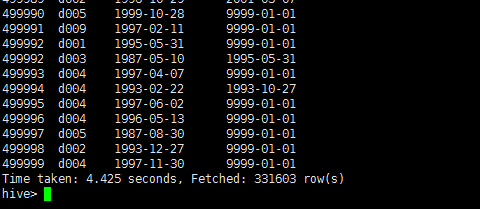
--fields-terminated-by "\t"  需同创建hive表时保持一致；

2、进入hive查看employees\_hive。

show tables;

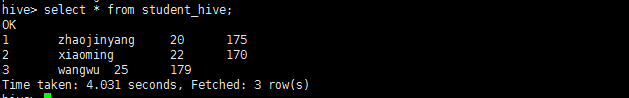


select \* from employees\_hive;



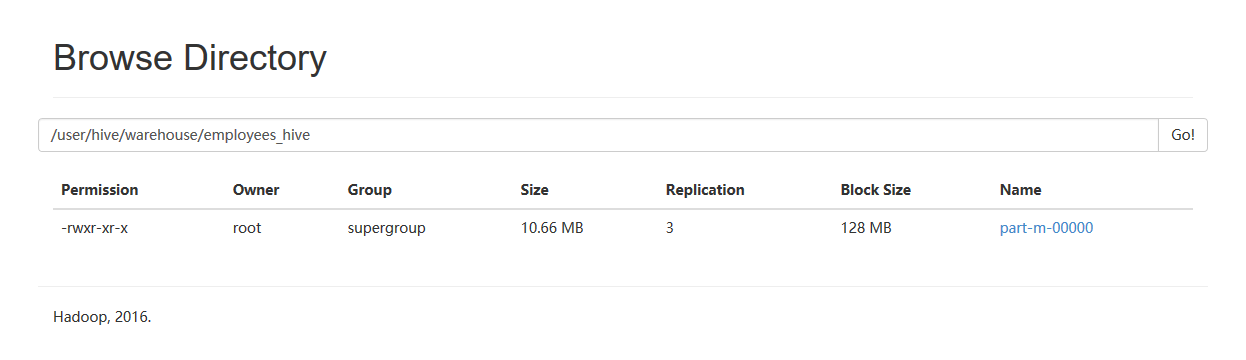
3、在hdfs上观察结果。

hadoop fs -cat /user/hive/warehouse/employees\_hive/part-m-00000



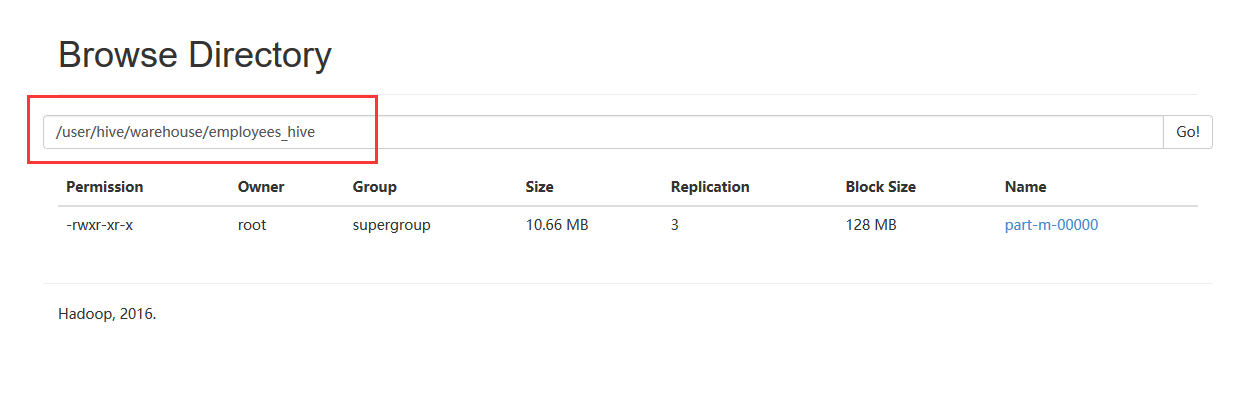
4、从mysql导入到hive中，在web观察。

将mysql里中的employees表的内容导入到hive中表保存在/user/hive/warehouse/employees\_hive/part-m-00000



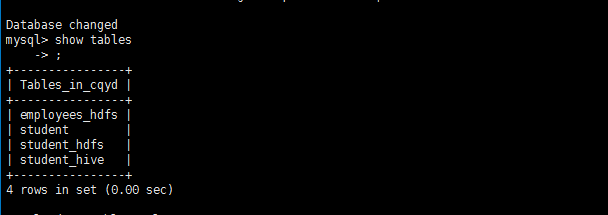
### **3.3.5、把数据从hive导出到mysql中**

1、将传入到hive上的/user/hive/warehouse/employees\_hive/part-m-00000中导入到mysql的employees\_hive空表中，通过web页面观察/user/hive/warehouse/employees\_hive/part-m-00000文件。



2、在mysql下建立空表employees\_hive

create table employees\_hdfs(emp\_no int(10),dept\_no varchar(50),from\_date DATE,to\_date DATE);



3、通过sqoop把数据从hive导出到mysql中

sqoop export --connect jdbc:mysql://192.168.56.100:3306/cqyd --username root --password 123456 --table employees\_hive --export-dir '/user/hive/warehouse/employees\_hive/part-m-00000' --input-fields-terminated-by "\t"

参数解释：

sqoop export ##表示数据从hdfs复制到mysql中

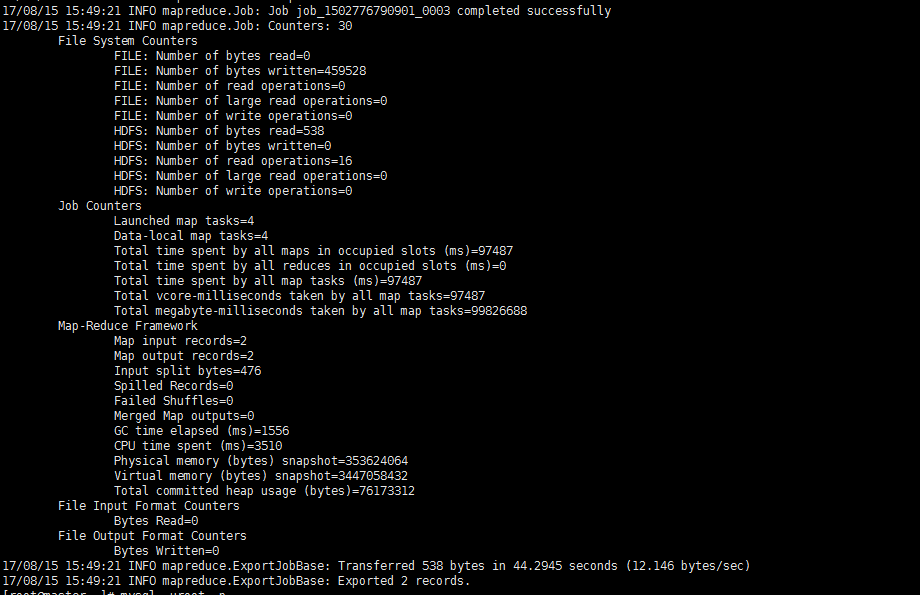
--connect jdbc:mysql://192.168.56.100:3306/zhaojinyang

--username root

--password 123456

--table employees\_hive ##mysq中表，即将被导入的表名称

--export-dir /user/hive/warehouse/employees\_hive/part-m-00000'  ##hdfs中被导出的文件



1. 进入mysql中观察结果。

select \* from employees\_hive;

