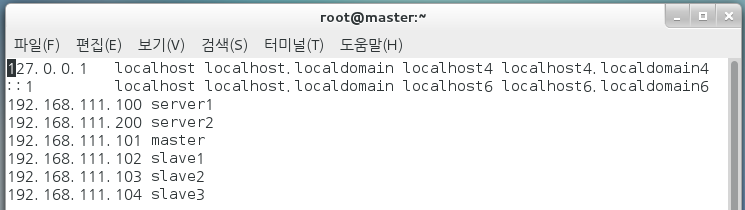
**<hadoop 설치>**

**1. hosts 설정**



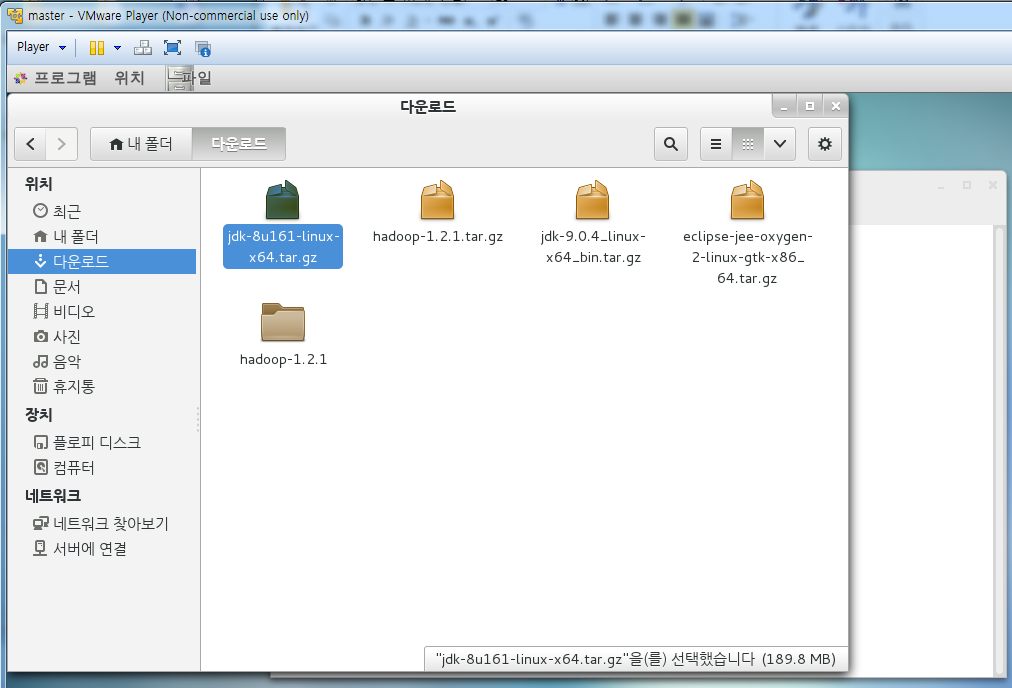


hosts 설정을 통해 각 호스트 명과 그에 해당하는 IP주소를 정의한다. (리눅스에서만 유효하며, window 에서는 C:\Windows\System32\drivers\etc\hosts 설정)

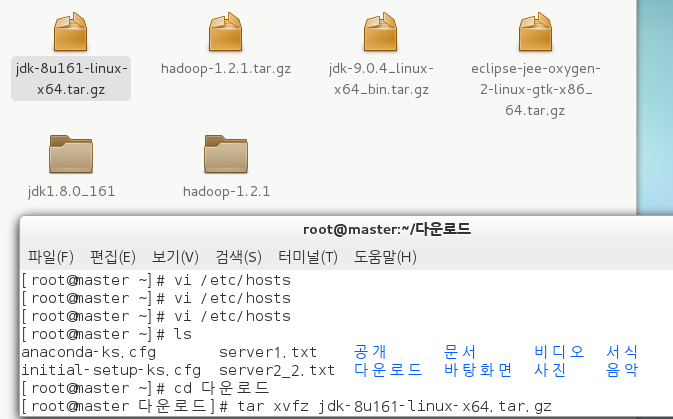
IP주소로 이동하지 않고 호스트 명으로 편하게 이동하기 위함. -> 해킹에도 사용되는 부분

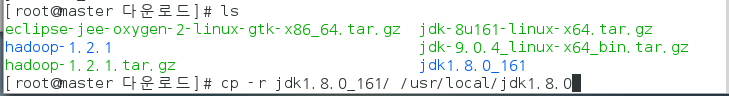
현재 호스트네임은 vi/etc/hostname 에서 master로 바꿔줬으며, 그 이름에 맞는 ip주소를 적어줬다.

**2. 자바(jdk)설치하기**



다운로드 폴더에 jdk8버전.gz를 옮겨주고 압축을 푼다.(압축 푼 후에 local로 옮겨도 되고 local로 가서 압축을 풀어도 된다.)8버전으로 할 것. 9버전은 하이브에서 오류남.

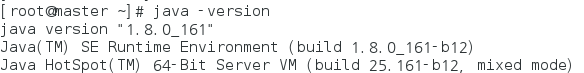




압축 풀린 jdk폴더를 local로 옮겨준다.

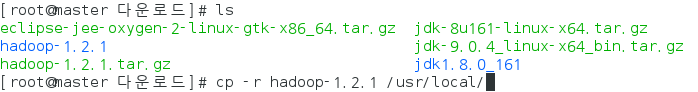


/usr/bin 에 자바 소프트링크 설정해주기.



자바 설치 잘 되었는지 확인하기.

**3. 하둡 설치하기**

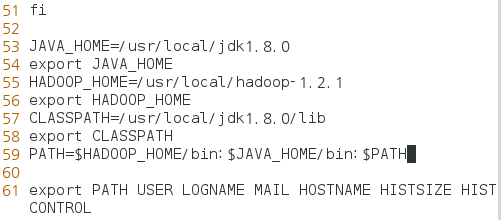


하둡 압축풀고 local로 복사해준다.

**4. 환경설정해주기**

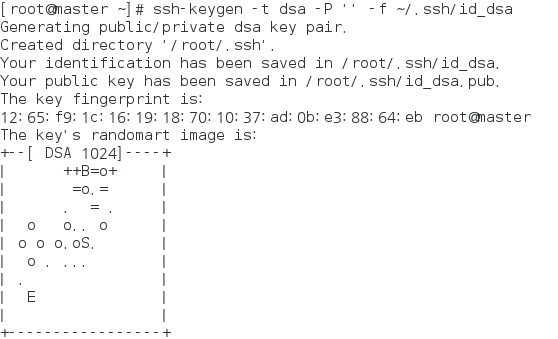
vi /etc/profile 실행

52라인에 아래와 같이 추가 (:set nu 로 줄 넘버 띄움)

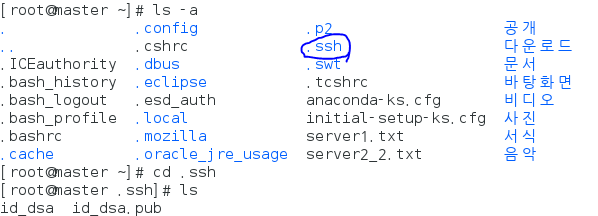


. /etc/profile 으로 설정된 것 재실행.

**5. SSH설정(다른 컴퓨터 들어갈 때 자동 허가 하도록 키를 만들어 나누는 것**



키 생성.(P 대문자)

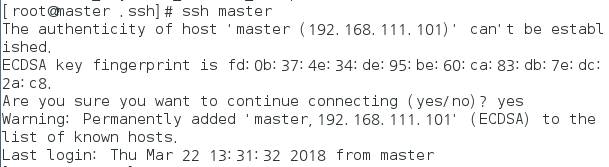


여기 있는 pub키를 내가 접속하고자 하는 PC에 복사해 주면 다음 접속 시 패스워드 물어보지 않는다.



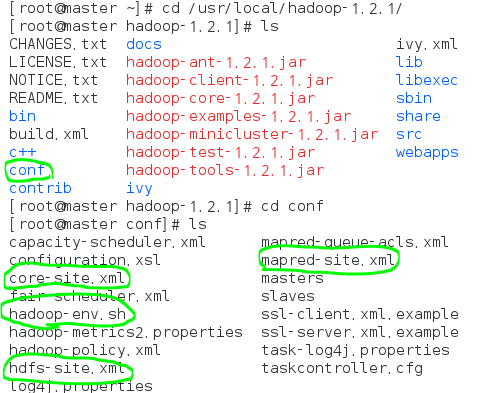
현재는 master 에서 master로 접속하는 가상 분산모드를 할 것이기 때문에 나에게 복사해줌.

다른 컴퓨터로 복사는 하둡교재 p.52참고

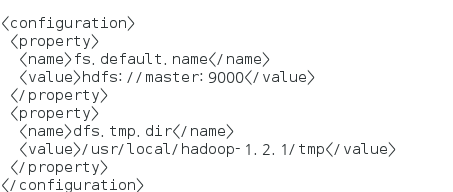


Master로 로그인시 패스워드 입력하지 않는 것 볼 수 있다. (원래는 password 입력해야함) 나올 땐 exit로 나온다.

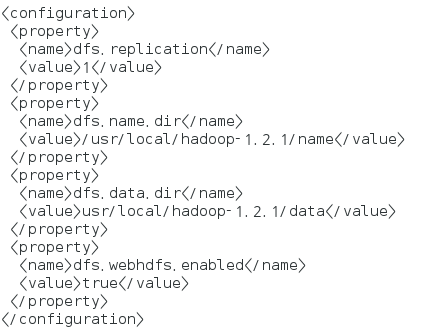
**6. configuration 파일 설정해주기.**



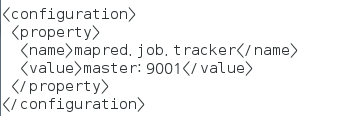
위 네 개 파일 설정을 변경해준다.



core-site.xml 위와 같이 수정



hdfs-site.xml 수정



Mapred-site.xml 수정(잡트래커의 수행 현황을 볼 수 있는 사이트 설정.)

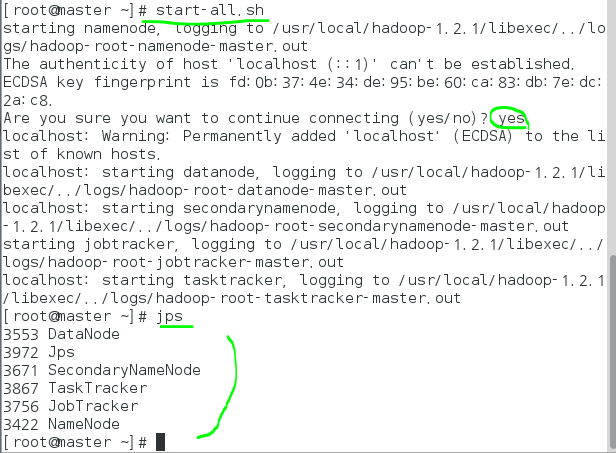


hadoop-env.sh 수정. 네 개 수정 모두 끝났으면 재부팅 실시!

**7. 하둡 포맷 후 제대로 작동 하는지 확인 하기.,**

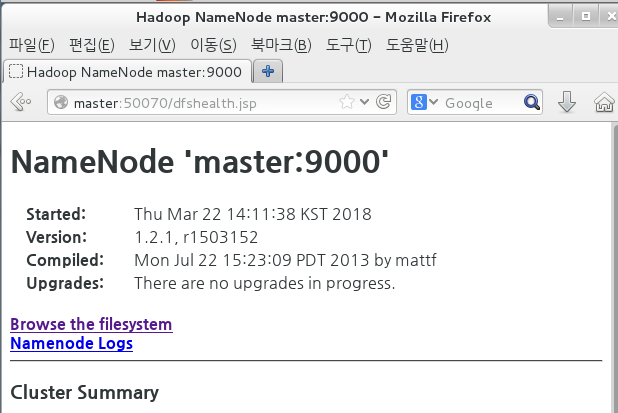
hadoop 쳐보고 명령어 잘 듣는지 확인.

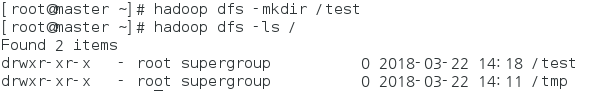
hadoop namenode –format 실행하여 포맷 후 start-all.sh 로 하둡 실행하여 jps 확인하기.



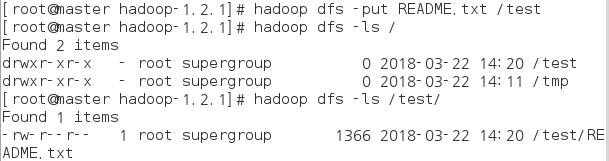
위와 같이 프로세스가 jps 포함해서 6개 나오면 정상.

파이어 폭스에서 master:50070 입력하여 확인하자.(윈도우에서 하려면 윈도우의 hosts 파일을 설정해주거나 192.168.111.101(master ip)로 접속해야함.

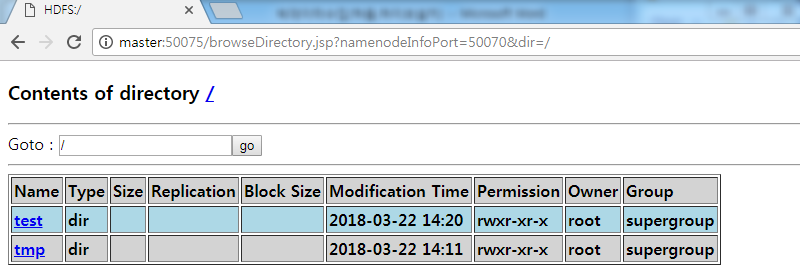


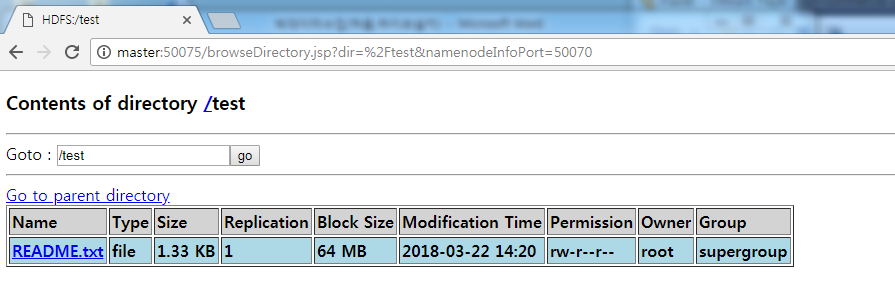


하둡에 test 폴더을 만들어 파일을 넣어보자.



하둡 폴더에 있던 README.txt 파일을 하둡에 복사했다.



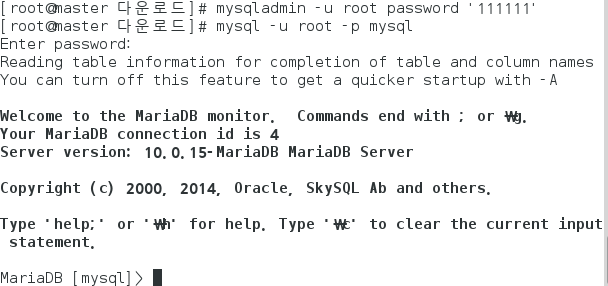


웹으로도 확인 가능하다.

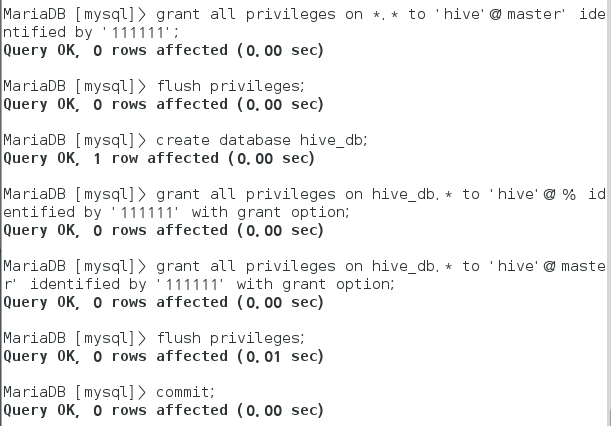
**<hive 설치하기>**

**1. 하둡의 데이터 구조를 정의할 수 있는 mariaDB 설치하기.**

리눅스 책 555~57p 따라서 마리아디비 깔기, firewall 설정까지 한다.

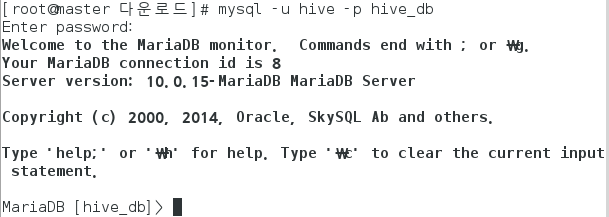


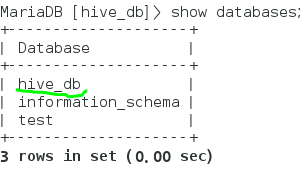
루트 계정 패스워드 111111로 변경하고 접속한다.



위의 명령어를 모두 수행하고 접속 권한을 설정한다.(궁금한건 리눅스 565페이지)

Master 외에 localhost도 추가해 주자.





hive 계정으로 디비 접속 되는지 확인하기.

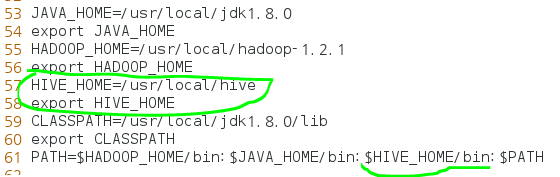
**2. hive 설치하기**

<http://hive.apache.org/downloads.html> 에서 하이브 패키지(~bin.tar.gz) 다운로드 하기.

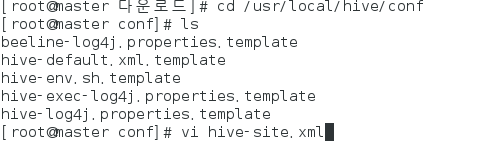




hive패키지 압축 풀고 hive이 름으로 local에 복사



vi /etc/profile 위와 같이 내용 추가후 재실행( . /etc/profile)



hive의 conf로 가서 hive-site.xml을 아래 내용으로 추가해준다.

<?xml version="1.0"?>

<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<configuration>

    <property>

        <name>hive.metastore.local</name>

        <value>true</value>

        <description>controls whether to connect to remove metastore server or open a new metastore server in Hive Client JVM</description>

    </property>

    <property>

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

        <value>jdbc:mariadb://localhost:3306/hive\_db?createDatabaseIfNotExist=true</value>

        <description>JDBC connect string for a JDBC metastore</description>

    </property>

    <property>

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

        <value>org.mariadb.jdbc.Driver</value>

        <description>Driver class name for a JDBC metastore</description>

    </property>

    <property>

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

        <value>hive</value>

        <description>username to use against metastore database</description>

    </property>

    <property>

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

        <value>111111</value>

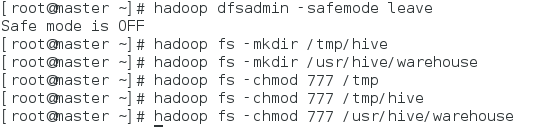
        <description>password to use against metastore database</description>

    </property>

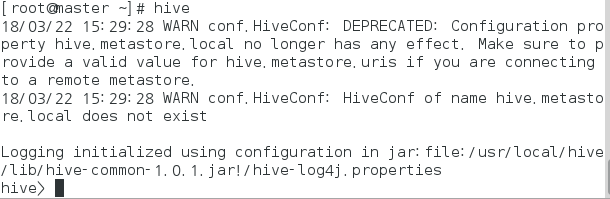
</configuration>



mariaDB JDBC API를 받아준다. 하이브 lib 안으로 옮겨준다.



하이브가 하둡에서 활동하는 디렉토리를 만들어줌-> 하이브와 하둡 연결



하이브 실행

재 부팅 후 다음 단계로.

**3. 하이브에 데이터 넣고 자바로 꺼내기.**

다시 start-all.sh(hadoop을 작동 시키고) 해주고 hive 접속 후 테이블을 만든다.

CREATE TABLE airline\_delay(

Year INT,

MONTH INT,

DayofMonth INT,

DayofWeek INT,

DepTime INT,

CRSDepTime INT,

ArrTime INT,

CRSArrTime INT,

UniqueCarrier STRING,

FlightNum INT,

TailNum STRING,

ActualElapsedTime INT,

CRSElapsedTime INT,

AirTime INT,

ArrDelay INT,

DepDelay INT,

Origin STRING,

Dest STRING,

Distance INT,

TaxiIn INT,

TaxiOut INT,

Cancelled INT,

CancellationCode STRING

COMMENT 'A = carrier, B = weather, C = NAS, D = security',

Diverted INT COMMENT '1 = yes, 0 = no',

CarrierDelay STRING,

WeatherDelay STRING,

NASDelay STRING,

SecurityDelay STRING,

LateAircraftDelay STRING)

COMMENT 'TEST DATA'

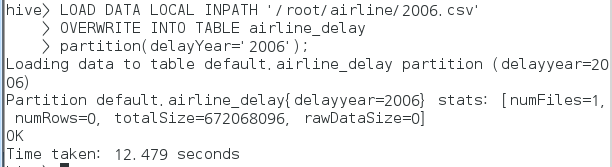
PARTITIONED BY (delayYear INT)

ROW FORMAT DELIMITED

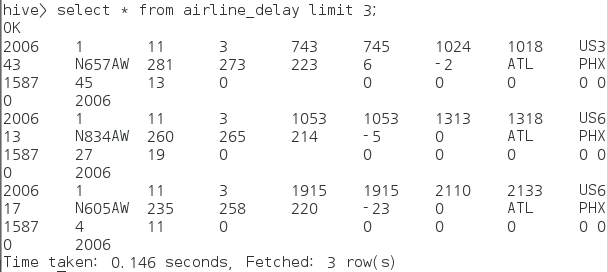
    FIELDS TERMINATED BY ','

    LINES TERMINATED BY '\n'

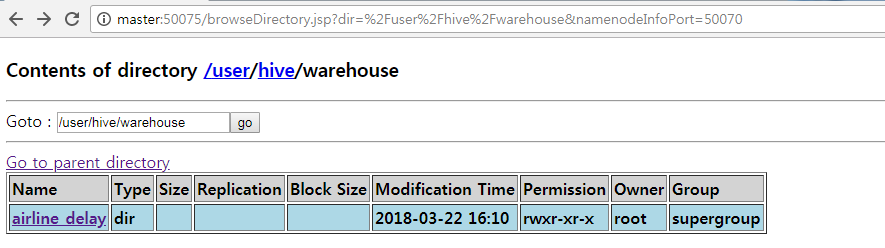
    STORED AS TEXTFILE;

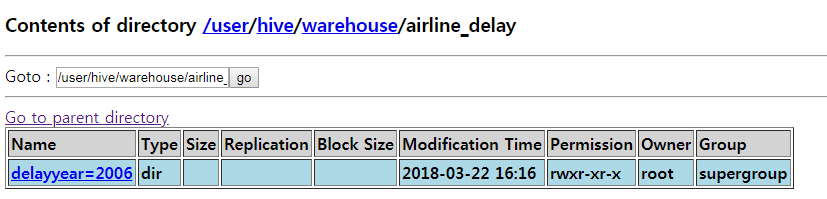


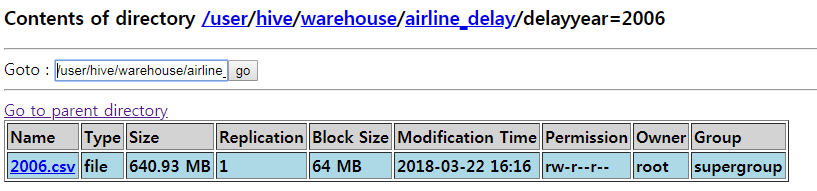
2006.csv 받아놓은 것 하이브 로드하기

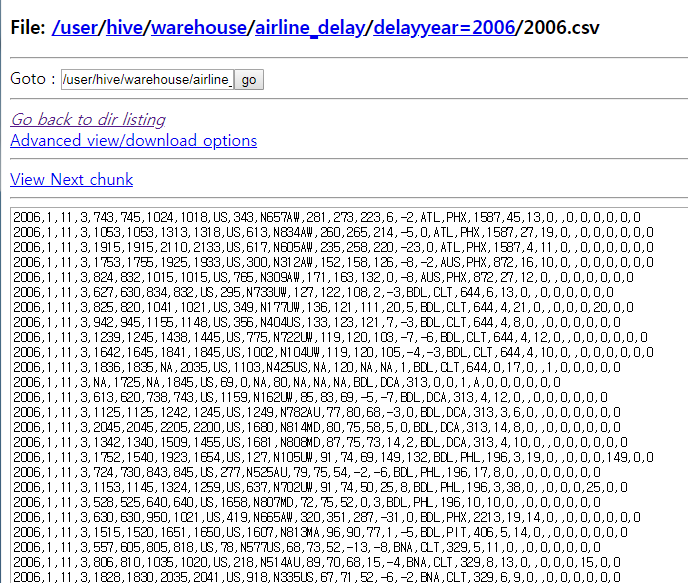


데이터 꺼내서 확인해보기.







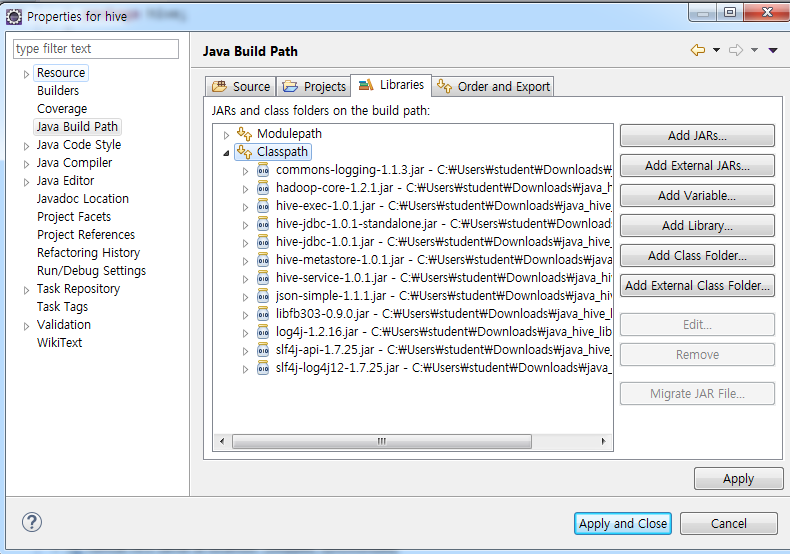
웹 관리화면에서도 확인 가능하다.

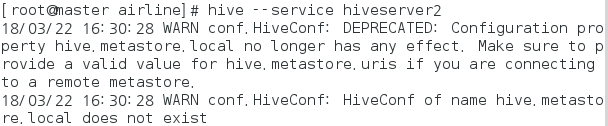
**4. 자바 어플리케이션 연동하기.**

이클립스에서 자바 프로젝트를 하나 만든다.

java\_hive 라이브러리를 다운받는다.

프로젝트의 property 설정에 가서 libraries 탭에서 add External JARs를 선택하여 다운받은 라이브러리를 추가한다.





하이브 서비스를 실행해 놓는다.

이클립스로 이동해서 아래와 같은 소스코드를 작성한다.

package hive;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.Statement;

public class HiveTest2 {

public static void main(String[] args) throws Exception {

Class.forName("org.apache.hive.jdbc.HiveDriver"); //hive driver loading

Connection conn = DriverManager.getConnection("jdbc:hive2://192.168.111.101:10000/default", "root", "111111");

Statement stmt = conn.createStatement();

ResultSet rs = stmt.executeQuery("select \* from airline\_delay limit 10");

while (rs.next()) {

System.out.println(rs.getInt(1));

System.out.println(rs.getInt(2));

System.out.println(rs.getInt(3));

}

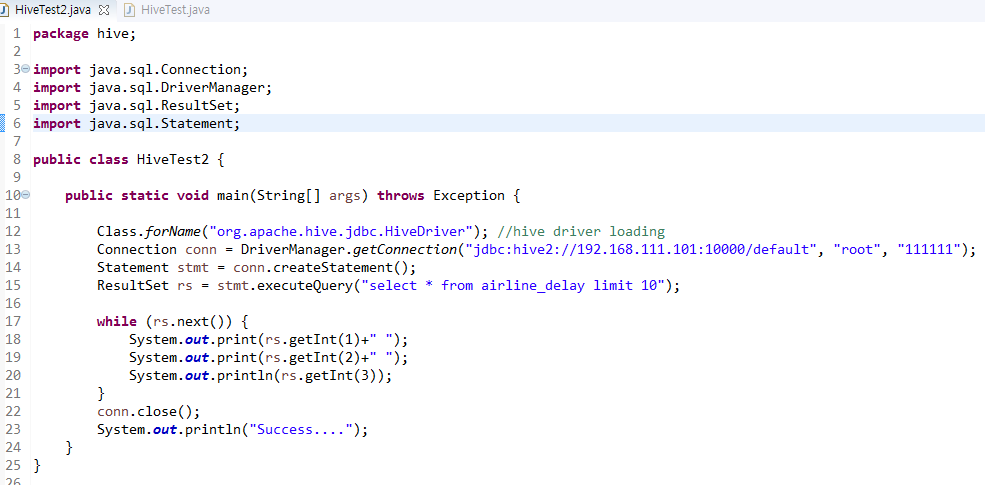
conn.close();

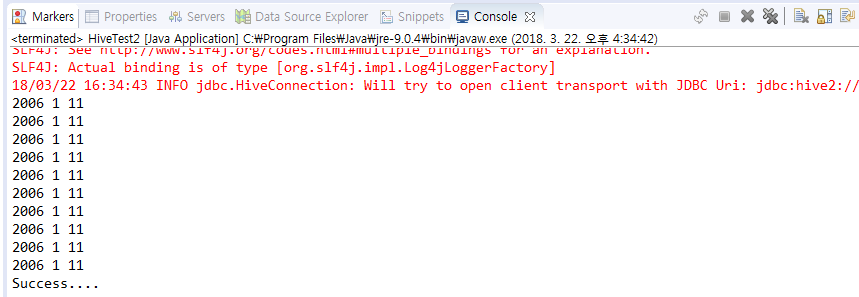
System.out.println("Success....");

}

}

jdbc 드라이버 로딩하고 계정 접속 했던 것 처럼 하이브로 접속한다.





하이브로부터 데이터를 정상적으로 가져왔다.