**How to set up DB and run Sqoop with Movielens Data**

( 이삭엔지니어링 인턴 김형근 [hkkim@isaac-eng.com](mailto:hkkim@isaac-eng.com) )  
( 테스트 환경 : CDH 5.15.1 DIA )

**Index**

1. **MySQL Test (10.100.1.116)**
2. **Movielens.sql 파일 다운받기**
3. **“movielens” DB생성과 “hadoop” 사용자 생성**
4. **Movielens.sql문 실행**
5. **Sqoop 진행 ( DIA utility1에서 진행)**
6. **최신영화 골라내기**
7. **Oracle (10.100.3.152)**
8. **MySQL Test (MySQL은 10.100.1.116에 설치되어 있음.)  
   (1) movielens.sql 파일 다운받기**

$ wget <https://s3.amazonaws.com/bigdata-hipic/movielens.sql>

**(2) “movielens” DB생성과 “hadoop” 사용자 생성**

$ mysql -u root -p

MariaDB [none] > CREATE DATABASE movielens;

MariaDB [none] > CREATE USER ‘hadoop’@’%’ IDENTIFIED BY ‘cisbigdata’;

MariaDB [none] > GRANT SELECT, RELOAD, PROCESS, REFERENCES, INDEX, SHOW DATABASES, EXECUTE, SHOW VIEW, EVENT , TRIGGER ON \*.\* TO ‘hadoop’@’%’ WITH GRANT OPTION;

MariaDB [none] > exit

( username : hadoop, password : cisbigdata)

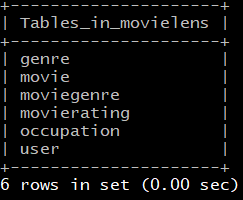
**(3) movielens.sql문 실행**

$ mysql -u hadoop -p

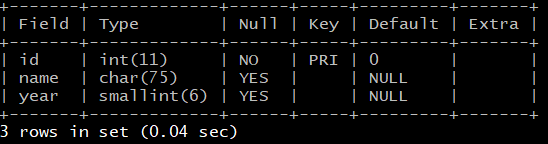
MariaDB [none] > use movielens;

MariaDB [movielens] > source /home/hkkim/movielens.sql;

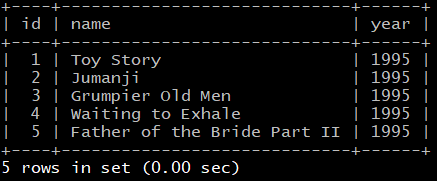
MariaDB [movielens] > show tables;



MariaDB [movielens] > describe movie;



MariaDB [movielens] > select \* from movie limit 5;



**(4) Sqoop 진행 ( DIA utility1에서 진행)**

$ hive

hive > create database movielensanalysis;

hive > quit;

// mysql의 movielens DB에 있던 모든 table을 hive로 import하는 sqoop문이다

$ sqoop import --connect jdbc:mysql://10.100.1.116/movielens --table **movie** --fields-terminated-by ‘\t’ --username hadoop --password cisbigdata   
--hive-import --hive-table movieanalysis.**movie**

$ sqoop import --connect jdbc:mysql://10.100.1.116/movielens --table **user**   
--fields-terminated-by ‘\t’ --username hadoop --password cisbigdata   
--hive-import --hive-table movieanalysis.**users**

$ sqoop import --connect jdbc:mysql://10.100.1.116/movielens --table **genre**   
--fields-terminated-by ‘\t’ --username hadoop --password cisbigdata   
--hive-import --hive-table movieanalysis.**genre**

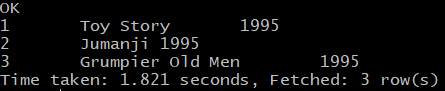
$ sqoop import --connect jdbc:mysql://10.100.1.116/movielens --table **moviegenre**   
--fields-terminated-by ‘\t’ --username hadoop --password cisbigdata   
--hive-import --hive-table movieanalysis.**moviegenre**

$ sqoop import --connect jdbc:mysql://10.100.1.116/movielens --table **movierating**   
--fields-terminated-by ‘\t’ --username hadoop --password cisbigdata   
--hive-import --hive-table movieanalysis.**movierating**

$ sqoop import --connect jdbc:mysql://10.100.1.116/movielens --table **occupation**   
--fields-terminated-by ‘\t’ --username hadoop --password cisbigdata   
--hive-import --hive-table movieanalysis.**occupation**

$ hive

hive > use movielensanalysis;

hive > select \* from movie limit 3;  


**(5) 최신 영화 골라내기**

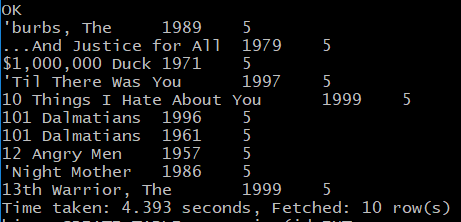
hive > CREATE EXTERNAL TABLE user\_rating (userid INT, numratings INT, avgrating FLOAT);

hive > INSERT OVERWRITE TABLE user\_rating SELECT userid, COUNT(userid), AVG(rating) FROM movierating GROUP BY userid;

hive > SELECT \* FROM movie SORT BY year DESC LIMIT 1;



hive > SELECT DISTINCT name, year, rating FROM movie LEFT OUTER JOIN movierating ON movie.id = movierating.movieid WHERE rating > 4.5 LIMIT 10;



hive > CREATE TABLE newmovie (id INT,

name STRING,

year INT,

numratings INT,

avgrating FLOAT );

hive > INSERT OVERWRITE TABLE newmovie

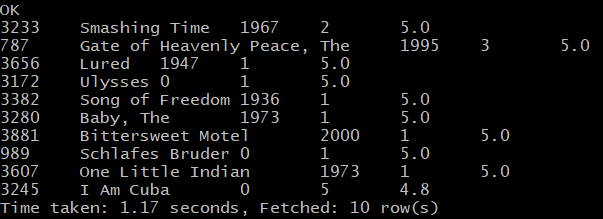
SELECT m.id, m.name, m.year, COUNT(1), AVG(mr.rating)

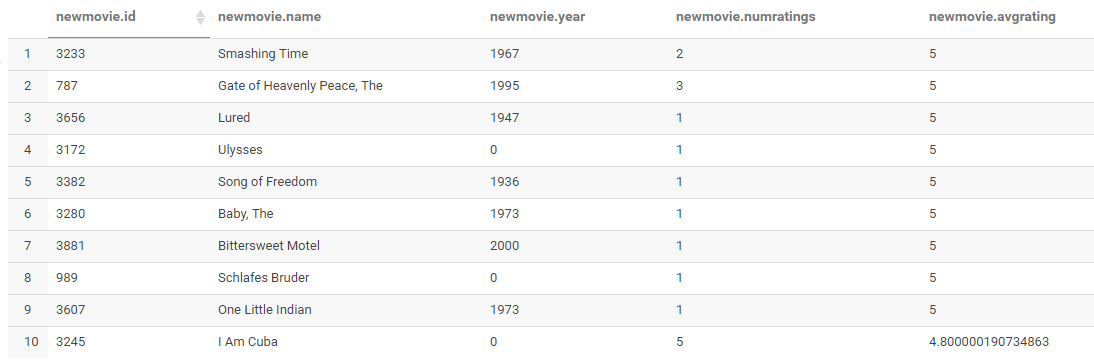
FROM movie m, movierating mr

WHERE m.id = mr.movieid

GROUP BY m.id, m.name, m.year;

hive > SELECT \* FROM newmovie ORDER BY avgrating DESC LIMIT 10;





1. **Oracle (미해결, Sqoop 마지막단계에서 Import Error 발생.)**

// Oracle에 MySQL의 movielens DB에 있던 table을 복사한 후, hive로 import하는 sqoop

$ sqoop import --connect jdbc:oracle:thin:@10.100.3.152:1521:DWD --table MOVIE --fields-terminated-by ‘\t’ --username selectonly --password 'ise1212' --hive-import --hive-table movieanalysis\_oracle\_hkkim.movie

$ sqoop import --connect jdbc:oracle:thin:@10.100.3.152:1521:DWD --table USER\_ --fields-terminated-by ‘\t’ --username selectonly --password 'ise1212' --hive-import --hive-table movieanalysis\_oracle\_hkkim.user --m 1

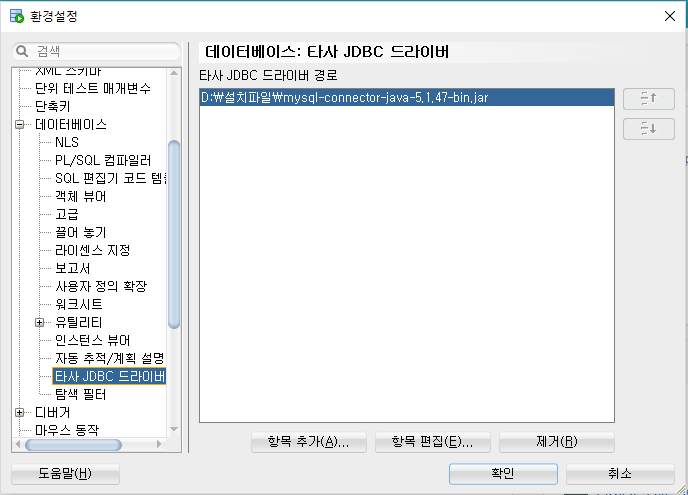
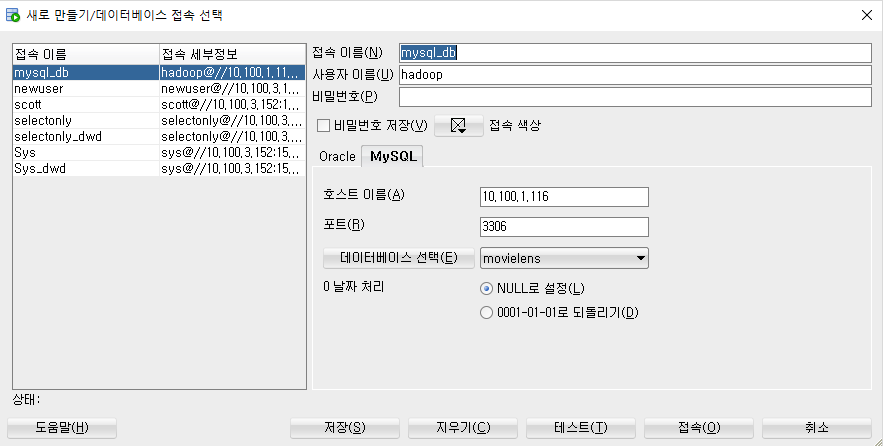
$ sqoop import --connect jdbc:oracle:thin:@10.100.3.152:1521:DWD --table GENRE --fields-terminated-by ‘\t’ --username selectonly --password 'ise1212' --hive-import --hive-table movieanalysis\_oracle\_hkkim.genre

$ sqoop import --connect jdbc:oracle:thin:@10.100.3.152:1521:DWD --table MOVIEGENRE --fields-terminated-by ‘\t’ --username selectonly --password 'ise1212' --hive-import --hive-table movieanalysis\_oracle\_hkkim.moviegenre

$ sqoop import --connect jdbc:oracle:thin:@10.100.3.152:1521:DWD --table MOVIERATING --fields-terminated-by ‘\t’ --username selectonly --password 'ise1212' --hive-import --hive-table movieanalysis\_oracle\_hkkim.movierating

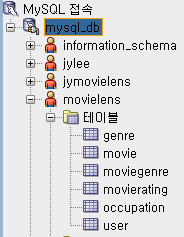
$ sqoop import --connect jdbc:oracle:thin:@10.100.3.152:1521:DWD --table OCCUPATION --fields-terminated-by ‘\t’ --username selectonly --password 'ise1212' --hive-import --hive-table movieanalysis\_oracle\_hkkim.occupation

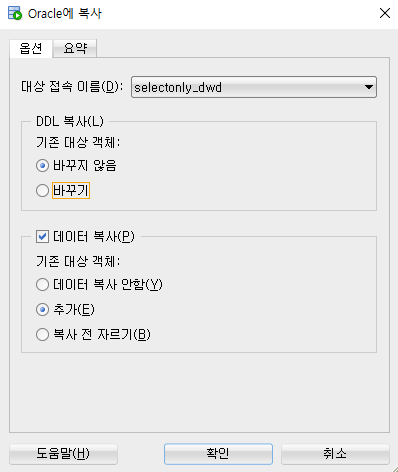
+ MySQL에 있는 DB를 SQL Developer에서 연결하는 방법

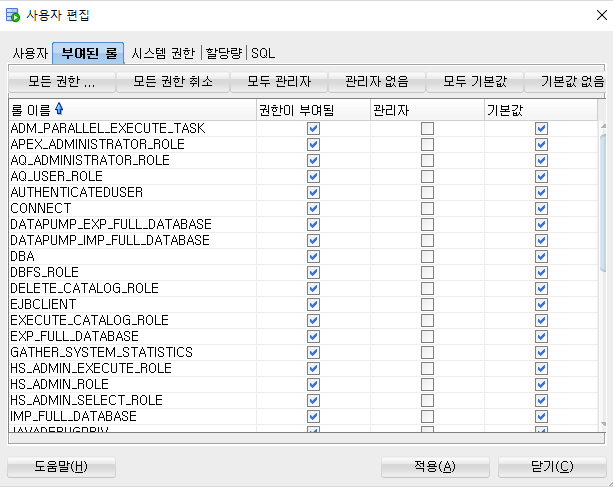
1. “도구” -> “환경설정” -> “데이터베이스” -> “타사 JDBC 드라이버” -> mysql JDBC 드라이버 셋팅  
   
2. Mysql 연결하기  
   

**호스트 이름** : ex) 10.100.1.116  
**포트** : 3306  
**데이터베이스 선택** : (연결하고 싶은 DB 선택)

1. 테이블을 Oracle로 복사 (원하는 테이블 선택 후, 오른쪽 마우스 클릭)





**\*유의점** : 연결하려는 대상에게 충분한 권한을 주어야한다. (그렇지 않으면 오류발생)   
하지만 구체적으로 어떤 걸 넣어야 할지 몰랐던 관계로  
**부여된 롤** : “모두 기본값”, **시스템권한** : “모두 관리자” 으로 설정했다.  


**시스템권한** 의 경우, 몇몇 권한은 넣을 수 없다는 메시지가 나올텐데 무시하고, 체크된것이라도 있다면 된다.  
