How to set up DB and run Sqoop with Movielens Data

(이삭엔지니어링 인턴 김형근 hkkim@isaac-eng.com)

(테스트 환경 : CDH 5.15.1 DIA)

Index

1. 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;

```
+---------------------+
| Tables_in_movielens |
+---------------------+
| genre               |
| movie               |
| moviegenre          |
| movierating         |
| occupation          |
| user                |
+---------------------+
6 rows in set (0.00 sec)
```

MariaDB [movielens] > describe movie;

```
+-------+-------------+------+-----+---------+-------+
| Field | Type        | Null | Key | Default | Extra |
+-------+-------------+------+-----+---------+-------+
| id    | int(11)     | NO   | PRI | 0       |       |
| name  | char(75)    | YES  |     | NULL    |       |
| year  | smallint(6) | YES  |     | NULL    |       |
+-------+-------------+------+-----+---------+-------+
3 rows in set (0.04 sec)
```

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

```
+----+------------------------------+------+
| id | name                         | year |
+----+------------------------------+------+
|  1 | Toy Story                    | 1995 |
|  2 | Jumanji                      | 1995 |
|  3 | Grumpier Old Men             | 1995 |
|  4 | Waiting to Exhale            | 1995 |
|  5 | Father of the Bride Part II | 1995 |
+----+------------------------------+------+
5 rows in set (0.00 sec)
```

(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;

```
OK
1       Toy Story       1995
2       Jumanji 1995
3       Grumpier Old Men        1995
Time taken: 1.821 seconds, Fetched: 3 row(s)
```

(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;

```
OK
3190    Supernova       2000
Time taken: 27.331 seconds, Fetched: 1 row(s)
```

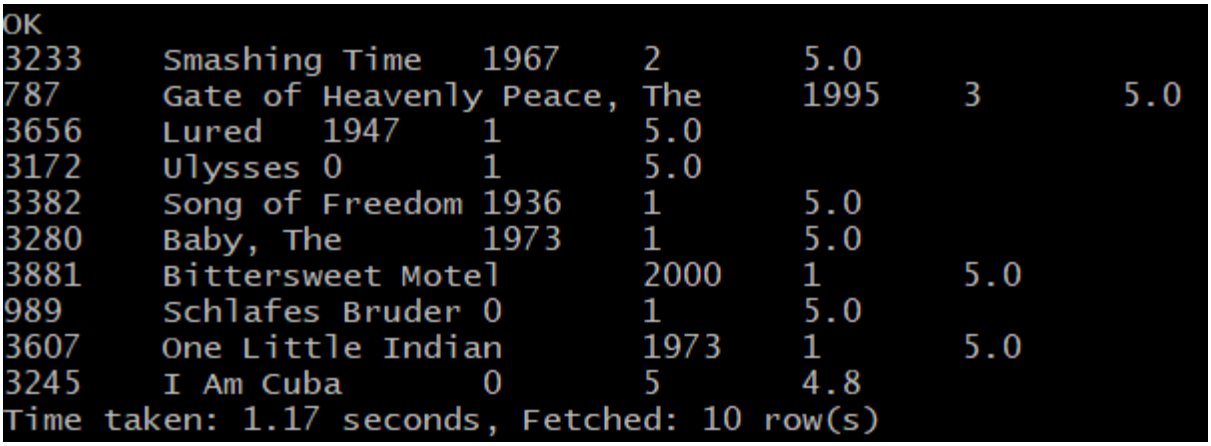
hive > SELECT DISTINCT name, year, rating FROM movie LEFT OUTER JOIN movierating ON movie.id = movierating.movieid WHERE rating > 4.5 LIMIT 10;

```
OK
'burbs, The     1989    5
...And Justice for All  1979    5
$1,000,000 Duck 1971    5
'Til There Was You      1997    5
10 Things I Hate About You      1999    5
101 Dalmatians  1996    5
101 Dalmatians  1961    5
12 Angry Men    1957    5
'Night Mother   1986    5
13th Warrior, The       1999    5
Time taken: 4.393 seconds, Fetched: 10 row(s)
```

```
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;
```



```
OK
3233    Smashing Time   1967    2       5.0
787     Gate of Heavenly Peace, The     1995    3       5.0
3656    Lured   1947    1       5.0
3172    Ulysses 0       1       5.0
3382    Song of Freedom 1936    1       5.0
3280    Baby, The       1973    1       5.0
3881    Bittersweet Motel       2000    1       5.0
989     Schlafes Bruder 0       1       5.0
3607    One Little Indian       1973    1       5.0
3245    I Am Cuba       0       5       4.8
Time taken: 1.17 seconds, Fetched: 10 row(s)
```

	newmovie.id	newmovie.name	newmovie.year	newmovie.numratings	newmovie.avgrating
1	3233	Smashing Time	1967	2	5
2	787	Gate of Heavenly Peace, The	1995	3	5
3	3656	Lured	1947	1	5
4	3172	Ulysses	0	1	5
5	3382	Song of Freedom	1936	1	5
6	3280	Baby, The	1973	1	5
7	3881	Bittersweet Motel	2000	1	5
8	989	Schlafes Bruder	0	1	5
9	3607	One Little Indian	1973	1	5
10	3245	I Am Cuba	0	5	4.800000190734863

2. Oracle (미해결, Sqoop 마지막단계에서 Import Error 발생.)

```
19/02/28 17:20:28 ERROR tool.ImportTool: Import failed: Character 8216 is an out
-of-range delimiter
```

// 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

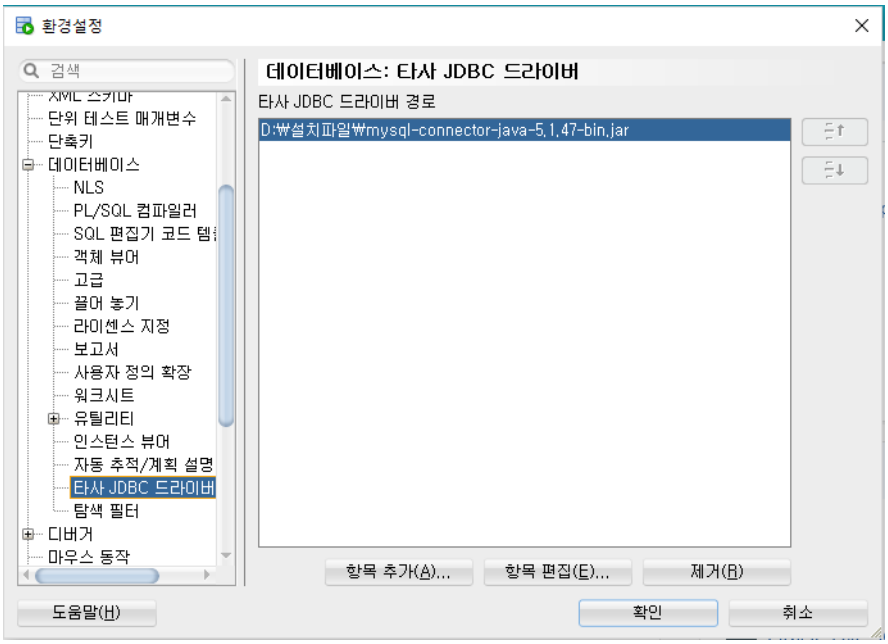
$ sqoop import --connect jdbc:oracle:thin:@10.100.3.152:1521:DWD --table MOVIEGENRE --fields-terminated-by '\t' --username selectonly --password 'ise1212' --hive-import --hive-table movieanalysis_oracle_hkkim.moviegenre

$ 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

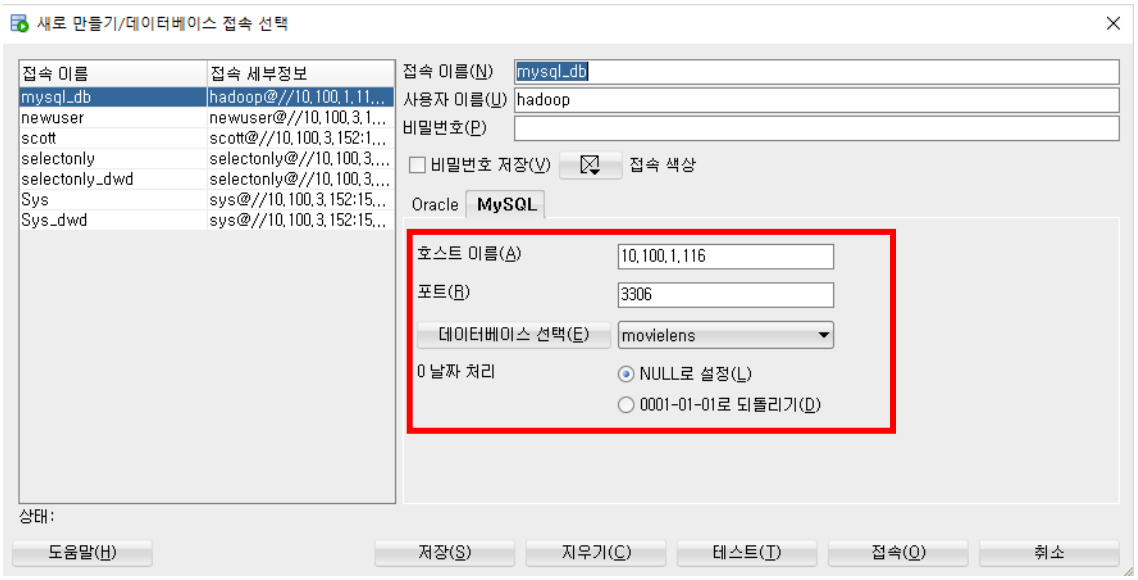
$ sqoop import --connect jdbc:oracle:thin:@10.100.3.152:1521:DWD --table OCCUPATION --fields-terminated-by '\t' --username selectonly --password 'ise1212' --hive-import --hive-table movieanalysis_oracle_hkkim.occupation

+ MySQL에 있는 DB를 SQL Developer에서 연결하는 방법

(1) “도구” -> “환경설정” -> “데이터베이스” -> “타사 JDBC 드라이버” -> mysql JDBC 드라이버 셋팅



(2) Mysql 연결하기

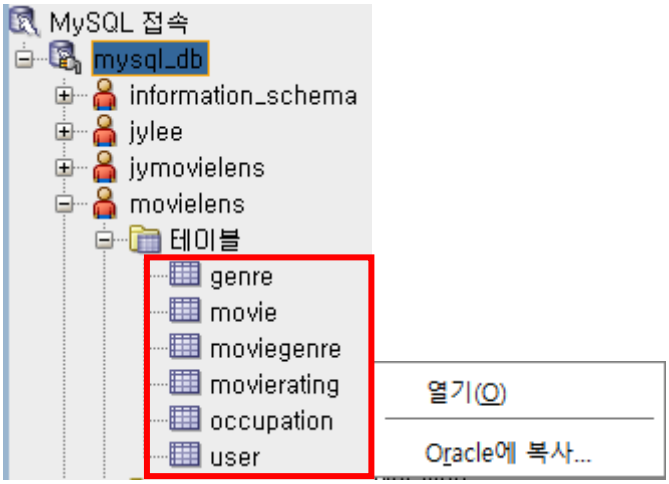


호스트 이름 : ex) 10.100.1.116

포트 : 3306

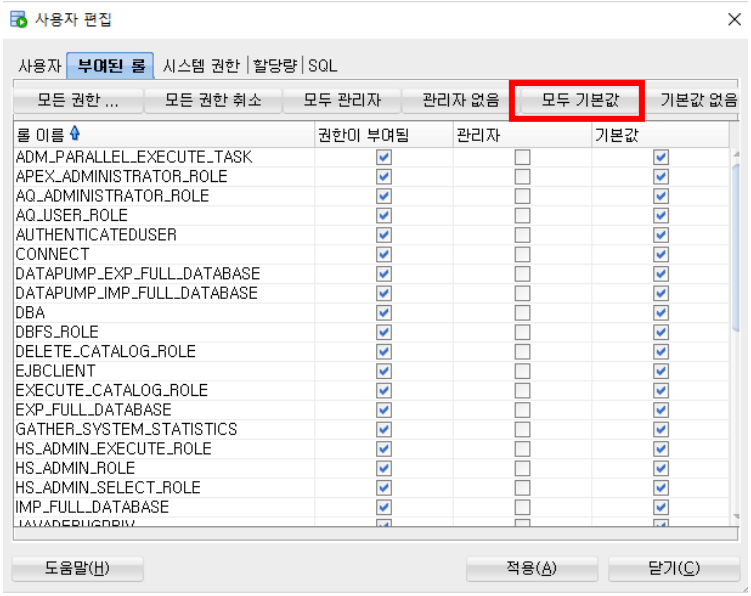
데이터베이스 선택 : (연결하고 싶은 DB 선택)

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


MySQL 접속
mysql_db
information_schema
jylee
jymovielens
movielens
테이블
genre
movie
moviegenre
movierating
occupation
user
열기(O)
Oracle에 복사...


Oracle에 복사
옵션
요약
대상 접속 이름(D):
selectonly_dwd
DDL 복사(L)
기존 대상 객체:
바꾸지 않음
바꾸기
데이터 복사(P)
기존 대상 객체:
데이터 복사 안함(Y)
추가(E)
복사 전 자르기(B)
도움말(H)
확인
취소

***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.