|  |
| --- |
| Máster en Big Data |
| Tecnologías de Almacenamiento |
| 10. Hands-On: Importar datos con Sqoop |

Presentado por: Jose David Angulo y Albert Ripoll

Índice

[1. Introducción3](#_Toc96244736)

[2. Entorno3](#_Toc96244737)

[3. Explorar los datos origen3](#_Toc96244738)

[4. Uso de Sqoop4](#_Toc96244739)

# Introducción

El objetivo de este Hands-On es ver un ejemplo de uso de Sqoop para importar datos de una base de datos MySQL

# Entorno

Para este Handos On, utilizaremos la máquina virtual desplegada en Hands-On anteriores llamada Developer\_Hadoop y todo será ejecutado vía shell

# Explorar los datos origen

Lo primero que haremos es login en MySQL para ejecutar las instrucciones mediante Shell. Los datos que exploraremos son los relativos a base de datos movielens. Por todo ello nos conectaremos mediante el comando:

mysql --user=training --password=training movielens

Entramos en mySql.

Texto, Carta

Descripción generada automáticamente

1. Revisa la estructura de la tabla movie y observa un ejemplo de datos de 5 registros.

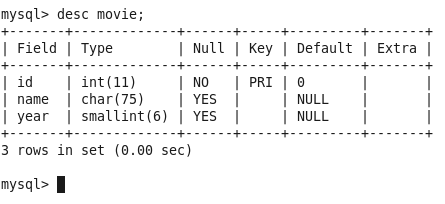
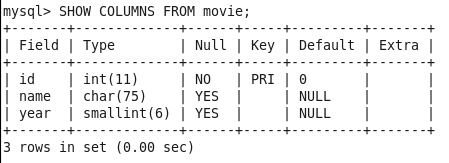
Para revisar la estructura podemos usar

Descr movie;

O también

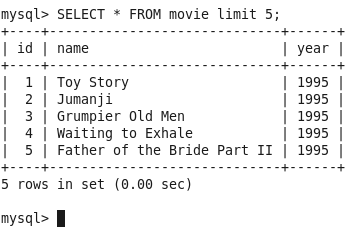
SHOW COLUMNS FROM movie;

ambos comandos son equivalentes



Para observar un ejemplo de datos de 5 registros usamos:

SELECT \* FROM movie limit 5;



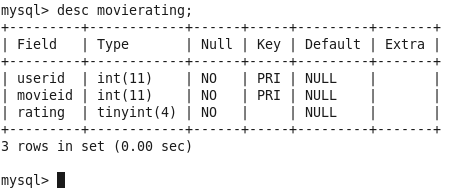
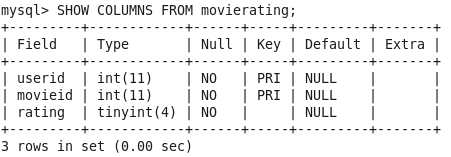
1. Revisa la estructura de la tabla movierating y observa un ejemplo de datos de 5 registros

Se puede usar

SHOW COLUMNS FROM movierating;

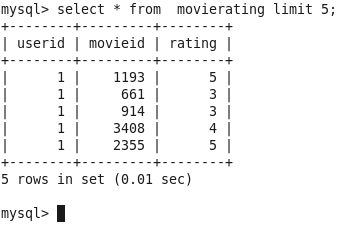
O también

desc movierating;



Para observar un ejemplo de datos de 5 registros usamos:

select \* from  movierating limit 5;

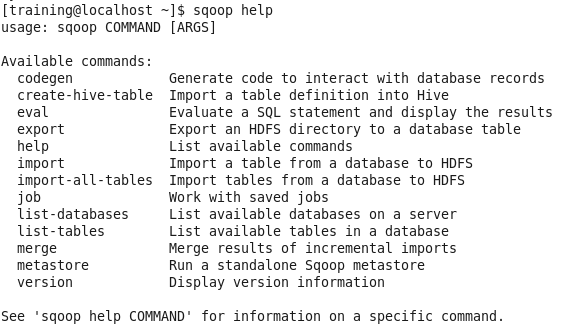


# Uso de Sqoop

1. Muestra la ayuda de Sqoop

Para mostrar la ayuda de Sqoop, hemos salido de mysql y se volvió a ejectuar la terminal antes de mandar el siguiente comando.

Sqoop help



1. Lista las bases de datos de MySQL desde Sqoop

sqoop list-databases --connect jdbc:mysql://localhost:3306/movielens --username training --password training

Texto

Descripción generada automáticamente

1. Lista las tablas de la base de datos movielens

sqoop list-tables \

--connect jdbc:mysql://localhost:3306/movielens \

--username training \

--password training

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

1. Importar la tabla movie y movierating  
    • Con movie.

importamos sqoop

conectamos a mysql al movilens con el usuario y password

pedimos la tabla movie localizada en user/hadoop/movielens/movie

y indicamos a sqoop que procese solo una sola tarea (un único maper) con m1.

El comando es el siguiente:

sqoop import \

--connect jdbc:mysql://localhost:3306/movielens \

--username training \

--password training \

--table movie \

--target-dir /user/hadoop/movielens/movie \

-m 1

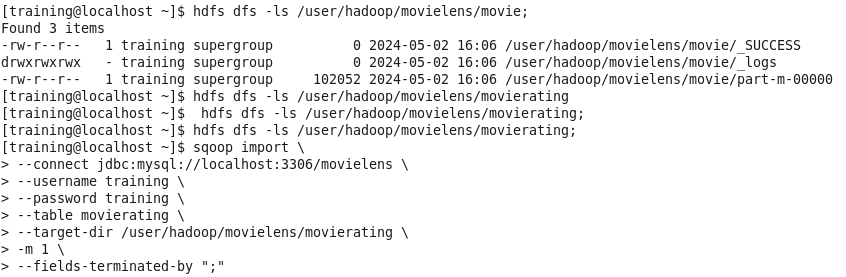
--fields-terminated-by";"

Texto

Descripción generada automáticamente

• Con movierating.

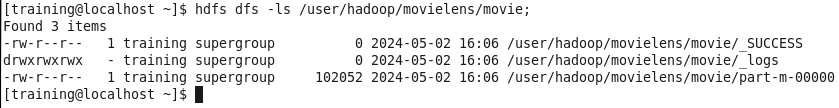
sqoop import \  
--connect jdbc:mysql://localhost:3306/movielens \  
--username training \  
--password training \  
--table movierating \  
--target-dir /user/hadoop/movielens/movierating \  
-m 1 \  
--fields-terminated-by ";"



1. Lista los archivos importados. Según la nomenclatura de los archivos resultantes, ¿Que característica tiene el Job de Sqoop que se ha ejecutado?

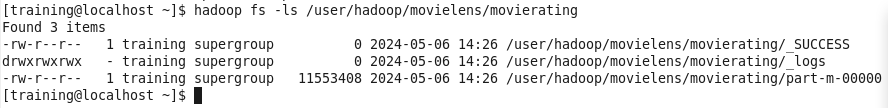
• Con movie.

hdfs dfs -ls /user/hadoop/movielens/movie;



• Con movierating.

hdfs dfs -ls /user/hadoop/movielens/movierating;

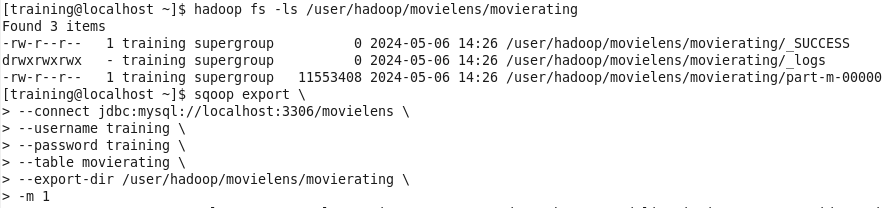


El sufijo de los archivos resultantes es **part-m-00000**. Esto indica que solo se ha generado una única partición de datos ya que se ha ejecutado un único mapper. Anterirormente al escribir -m1 hemos usado un mapper.

Aparece 1 porque nosotros comandamos para 1 pero por defecto lo normal sería que fuesen 4.

1. Hacer un export de la misma tabla.

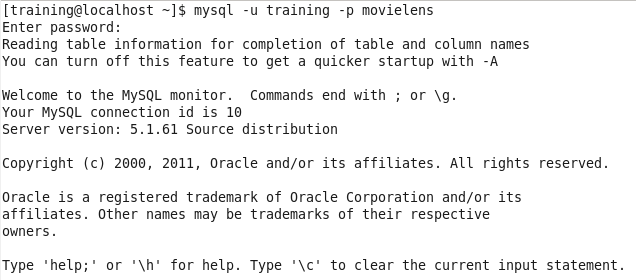
sqoop export \  
--connect jdbc:mysql://localhost:3306/movielens \  
--username training \  
--password training \  
--table movierating \  
--export-dir /user/hadoop/movielens/movierating \  
-m 1



1. Mirar que este en la base de datos.

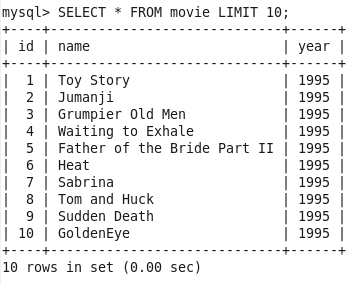
mysql -u training -p movielens

Enter password: training



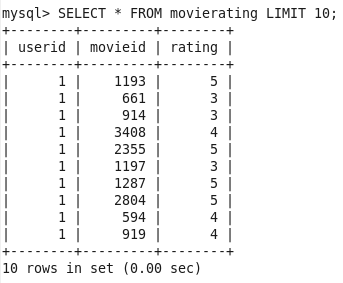
• Con movie.

SELECT \* FROM movie LIMIT 10;



• Con movierating.

SELECT \* FROM movierating LIMIT 10;



1. Exportar los datos a una nueva base de datos en el servidor de MySQL.

CREATE DATABASE albert;

C:\Users\alber\Downloads\ripoll.png

Use albert

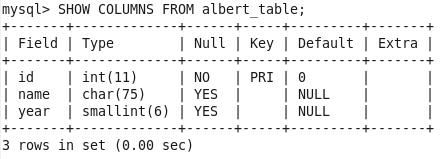
Des de allí especificamos que queremos usar la base de datos albert, antes de crear las tablas ya que si no especificamos use albert, vamos a crear las tablas en la base de datos que estemos, que en este caso seria movilens.

C:\Users\alber\Downloads\use.png

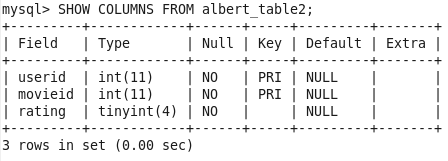
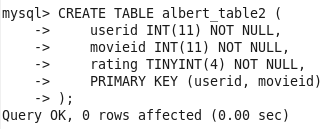
Ahora creamos las tablas.

CREATE TABLE albert\_table (id INT(11) NOT NULL PRIMARY KEY DEFAULT 0,name CHAR(75),year SMALLINT(6));

C:\Users\alber\Downloads\tal.png



CREATE TABLE albert\_table2 (  
    userid INT(11) NOT NULL,  
    movieid INT(11) NOT NULL,  
    rating TINYINT(4) NOT NULL,  
    PRIMARY KEY (userid, movieid)  
);



1. Explorar las tablas de la nueva base de datos para comprovar que ha funcionado.

• Exportamos los datos de movie a la base de datos albert en la tabla albert\_table

sqoop export \

--connect jdbc:mysql://localhost:3306/albert\  
--username training \  
--password training \  
--table albert\_table \  
--export-dir /user/hadoop/movielens/movie \  
-m 1

sqoop export --connect jdbc:mysql://localhost:3306/albert--username training --password training --table albert\_table --export-dir /user/hadoop/movielens/movie

• Exportamos los datos de movierating a la base de datos albert en la tabla albert\_table2

sqoop export \

--connect jdbc:mysql://localhost:3306/albert\  
--username training \  
--password training \  
--table albert\_table2 \  
--export-dir /user/hadoop/movielens/movierating \  
-m 1

sqoop export --connect jdbc:mysql://localhost:3306/albert--username training --password training --table albert\_table2 --export-dir /user/hadoop/movielens/movie