

Sqoop Hands-on – Activity 1

Validating MySQL install on the Linux machine

Go to the command prompt and type

```
mysql -u root -p
```

This will get you the password prompt. There is no password for the root user, so please hit the Enter key to login and get to the mysql prompt.

Exploring MySQL

Show existing databases:

```
Show databases;
```

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| hue       |
| loudacre  |
| metastore |
| mysql     |
| test      |
+-----+
6 rows in set (0.01 sec)
```

Using the Loudacre database

```
use loudacre;
```

View the tables in the Loudacre database

```
show tables;
```

```
mysql> use loudacre;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_loudacre |
+-----+
| accountdevice      |
| accounts            |
| basestations        |
| callog              |
| customerservicerep  |
| device              |
| knowledgebase        |
| mostactivestations  |
| toparticles         |
| webpage             |
| websitehit          |
+-----+
11 rows in set (0.00 sec)
```

Note that the Cludera VM comes configured with the MySQL database and the JDBC connectivity already set.

Importing a table from MySQL

Exit out from mysql using the following command:

```
exit;
```

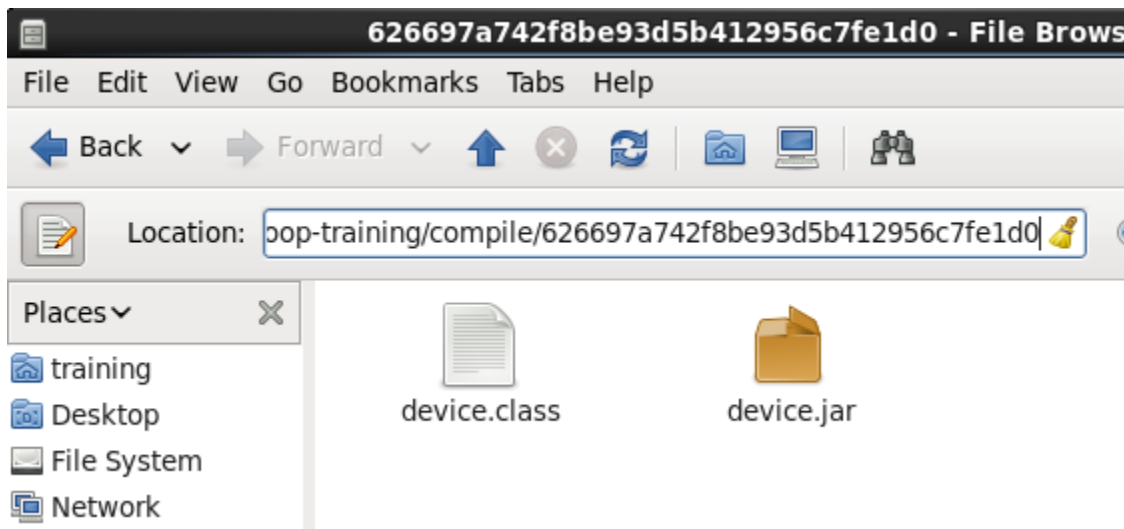
Let's try to import the 'device' table into HDFS using the following command:

```
sqoop import \
--connect jdbc:mysql://localhost/loudacre \
--username training --password training \
--table device \
-m 1
```

The `-m 1` command imports the data as one file.

```
[training@localhost ~]$ sqoop import \
> --connect jdbc:mysql://localhost/loudacre \
> --username training --password training \
> --table device \
> -m 1
17/01/30 20:18:10 INFO sqoop.Sqoop: Running Sqoop version: 1.4.5-cdh5.4.3
17/01/30 20:18:10 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
17/01/30 20:18:10 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
17/01/30 20:18:10 INFO tool.CodeGenTool: Beginning code generation
17/01/30 20:18:11 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `device` A
S t LIMIT 1
17/01/30 20:18:11 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `device` A
S t LIMIT 1
17/01/30 20:18:11 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-mapreduce
Note: /tmp/sqoop-training/compile/626697a742f8be93d5b412956c7fe1d0/device.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
17/01/30 20:18:13 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-training/compile/626697a742f8be93d5b412956c7fe1d0/device.jar
17/01/30 20:18:14 WARN manager.MySQLManager: It looks like you are importing from mysql.
17/01/30 20:18:14 WARN manager.MySQLManager: This transfer can be faster! Use the --direct option to exercise a MySQL-specific fast path.
```

Notice that Sqoop creates a .JAR file for this table. Copy the location mentioned on the highlighted line to a file browser to see the file.



Now validate the creation of the file in HDFS.

```
hdfs dfs -ls;
```

Notice that a directory is created by the same name as the name of the table at the root HDFS

```
hdfs dfs -ls device;
```

```
[training@localhost ~]$ hdfs dfs -ls device
Found 2 items
-rw-rw-rw- 1 training supergroup          0 2017-01-30 20:18 device/_SUCCESS
-rw-rw-rw- 1 training supergroup    2183 2017-01-30 20:18 device/part-m-00000
```

Notice that a `_SUCCESS` file is created and the table data is loaded to one file called `part-m-0000`.

View the content of the file using the CAT command.

```
hdfs dfs -cat device/part-m-00000;
```

```
[training@localhost ~]$ hdfs dfs -cat device/part-m-00000
1,2008-10-21 00:00:00.0,Sorrento F00L,phone
2,2010-04-19 00:00:00.0,Titanic 2100,phone
3,2011-02-18 00:00:00.0,MeeToo 3.0,phone
4,2011-09-21 00:00:00.0,MeeToo 3.1,phone
5,2008-10-21 00:00:00.0,iFruit 1,phone
6,2011-11-02 00:00:00.0,iFruit 3,phone
7,2010-05-20 00:00:00.0,iFruit 2,phone
8,2013-07-02 00:00:00.0,iFruit 5,phone
9,2008-10-21 00:00:00.0,Titanic 1000,phone
10,2008-10-21 00:00:00.0,MeeToo 1.0,phone
11,2011-02-28 00:00:00.0,Sorrento F21L,phone
12,2012-10-25 00:00:00.0,iFruit 4,phone
13,2011-11-21 00:00:00.0,Sorrento F23L,phone
14,2010-05-25 00:00:00.0,Titanic 2200,phone
15,2010-06-20 00:00:00.0,Ronin Novelty Note 1,phone
```

Notice that since we did not provide a directory or delimiter, the data is imported to the 'device' directory under the default HDFS directory as comma delimited.

Now import the Device table to the `/loudacre/device` directory.

```
sqoop import \  
--connect jdbc:mysql://localhost/loudacre \  
--username training --password training \  
--table device \  
--target-dir /loudacre/device \  
-m 1
```

* Note – If you try to import a table and the directory already exists in the path specified, you will get an error message.

Excluding tables while Import

You can exclude tables during the import using the `--exclude-tables` command

```
sqoop import-all-tables \  
--connect jdbc:mysql://localhost/loudacre \  
--username training --password training \  
--exclude-tables "accounts,device" -m 1
```

Deleting Directories in HDFS

Delete the directories created in the previous exercise. One of the commands for deletion is shown below.

```
hdfs dfs -rmr \knowledgebase
```