Exploring MySQL

This lesson is a gentle exploration of MySQL.

Exploring MySQL

In this lesson, we'll start with the initial state of a MySQL installation. The DBMS doesn't contain any user databases as you'll shortly observe. Follow the steps below for this exercise:

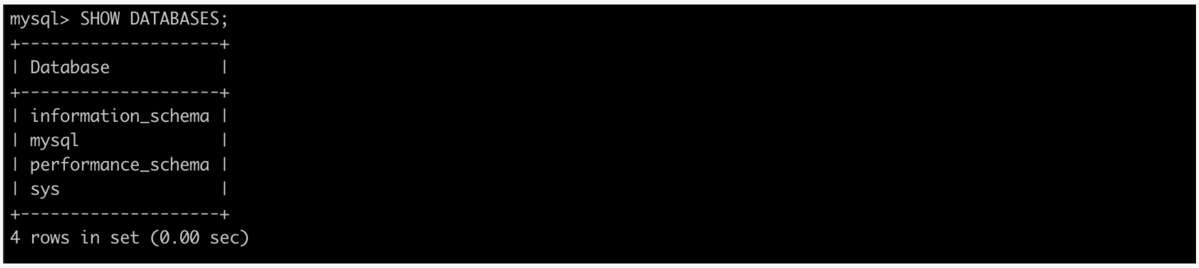
> Connect to the terminal below by clicking in the widget. Once connected, the command line prompt will show up. Enter or copy and paste the command **./DataJek/Lessons/1lesson.sh** and wait for the mysql prompt to start-up.

```
-- The lesson queries are reproduced below for convenient copy/paste into the terminal.

-- Query 1
SHOW DATABASES;

-- Query 2
USE mysql;

-- Query 3
SHOW CREATE DATABASE mysql;

-- Query 4
SHOW TABLES;

-- Query 5
DESCRIBE user;

-- Query 6
SHOW CREATE TABLE servers;

-- Query 7
SHOW COLUMNS FROM servers;
```

Terminal

1. Now execute the following command and observe the output:

   ```
   SHOW DATABASES;
   ```

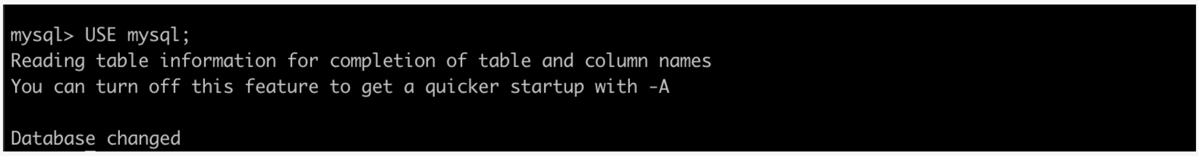
   

   ```
   mysql> SHOW DATABASES;
   +--------------------+
   | Database           |
   +--------------------+
   | information_schema |
   | mysql              |
   | performance_schema |
   | sys                |
   +--------------------+
   4 rows in set (0.00 sec)
   ```

   You'll see four databases that are used by the system. The query only shows databases that you have the privilege to view.

2. In order to explore a particular database, we need to tell the DBMS that we want our queries directed to the database of our choice. For our case, let's pick the existing MySQL database by executing the following command:

   ```
   USE mysql;
   ```

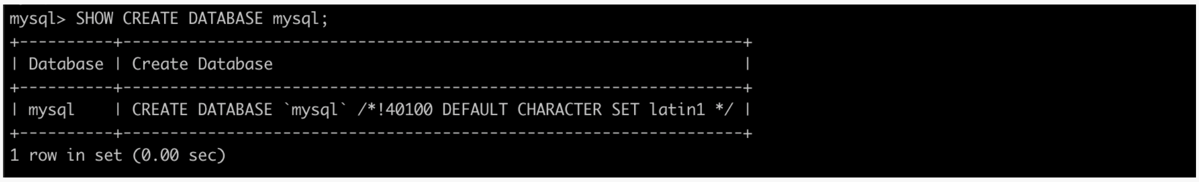
   

   ```
   mysql> USE mysql;
   Reading table information for completion of table and column names
   You can turn off this feature to get a quicker startup with -A

   Database changed
   ```

   The MySQL prompt will respond with a "Database changed" message. The **USE** statements allow us to let MySQL know the database we want to interact with. Any queries we execute in the future are directed to the selected database.

3. The database MySQL has been created for us already. We can examine how the database was created using the following query:

   ```
   SHOW CREATE DATABASE mysql;
   ```

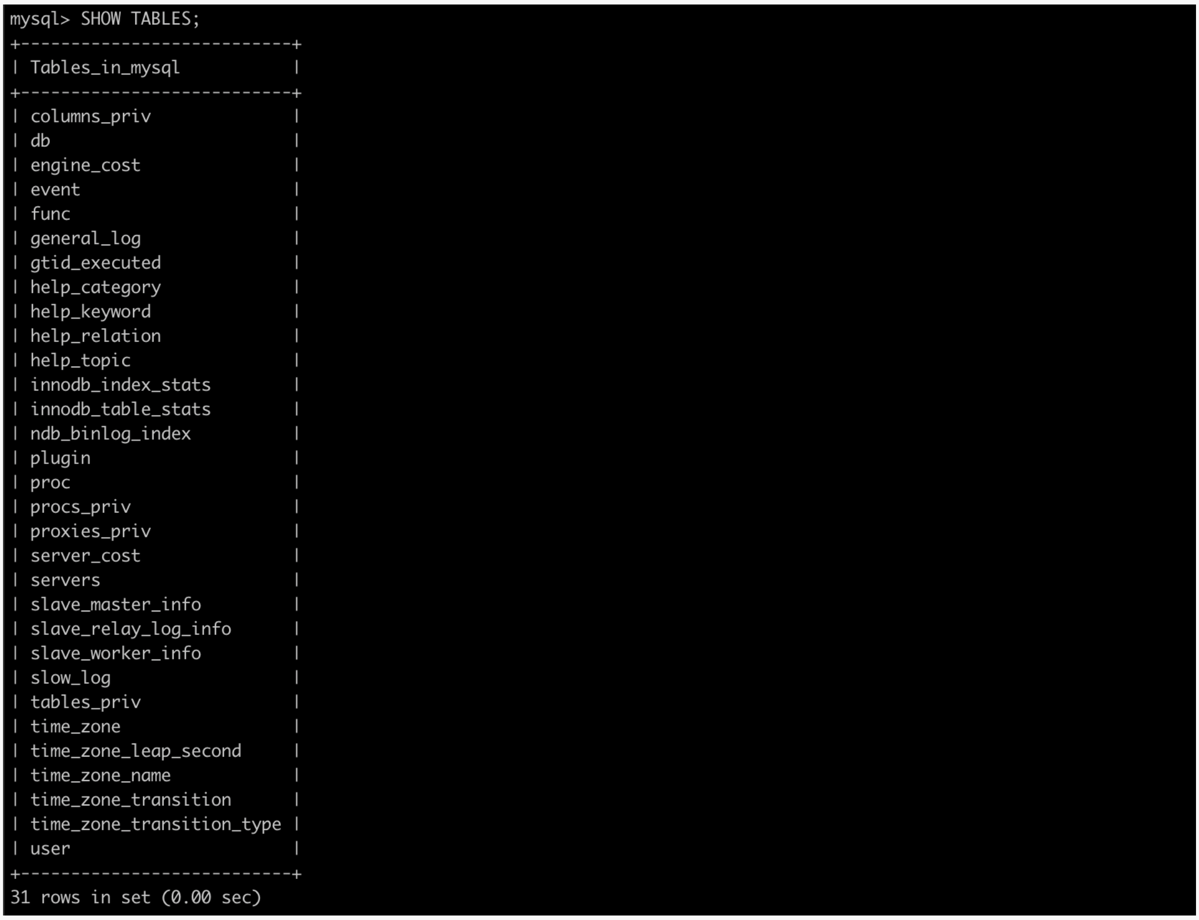
   

   ```
   mysql> SHOW CREATE DATABASE mysql;
   +----------+----------------------------------------------------------------+
   | Database | Create Database                                                |
   +----------+----------------------------------------------------------------+
   | mysql    | CREATE DATABASE `mysql` /*!40100 DEFAULT CHARACTER SET latin1 */ |
   +----------+----------------------------------------------------------------+
   1 row in set (0.00 sec)
   ```

The line **/*!40100 DEFAULT CHARACTER SET latin1 */** is a comment and encloses MySQL extensions to the SQL standard. For instance, the numeral 40100 indicates the minimum version of MySQL that can process the **SHOW CREATE DATABASE** query.

4. Let's explore the MySQL database further. We'd like to know what tables the MySQL database holds. We can do this by using the **SHOW** statement as follows:

```
SHOW TABLES;
```

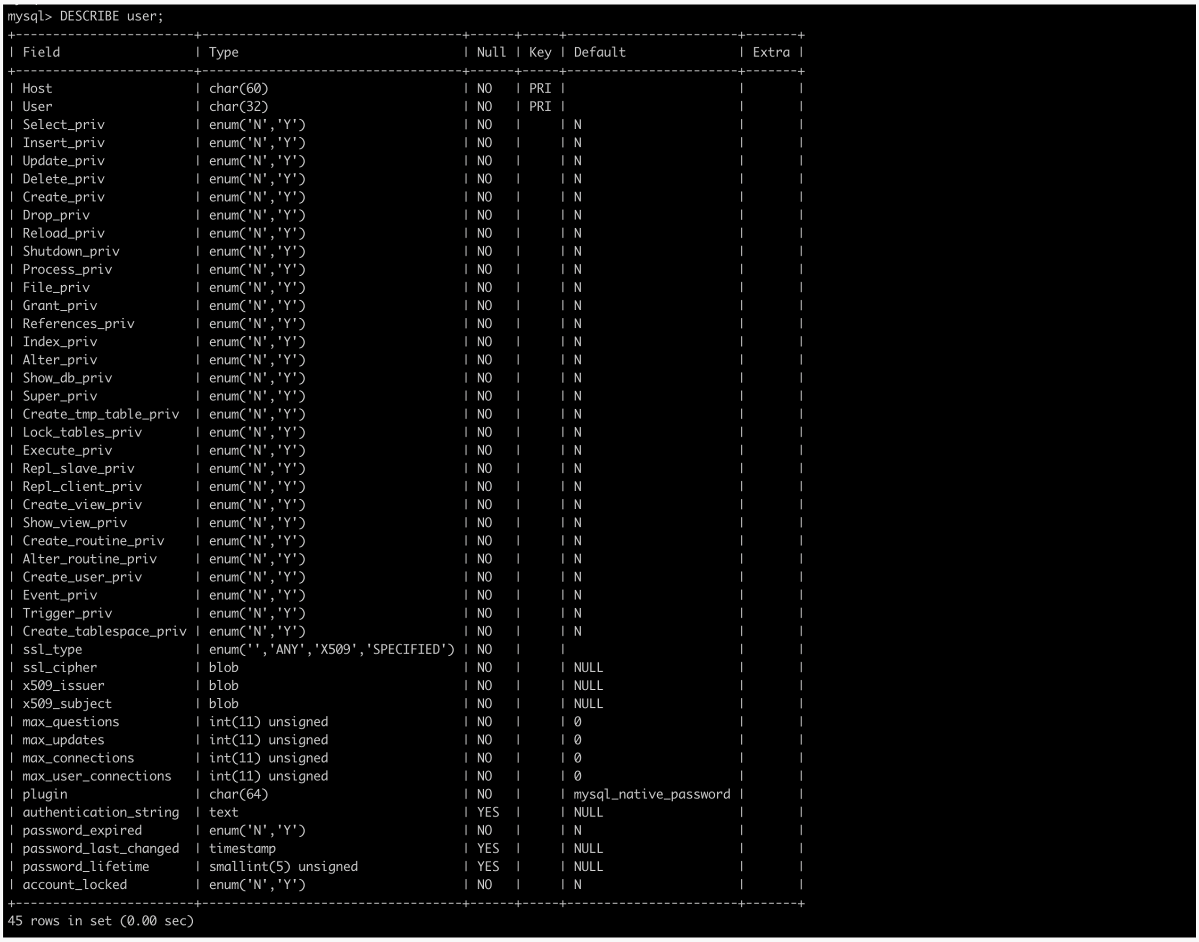


```
mysql> SHOW TABLES;
+---------------------------+
| Tables_in_mysql           |
+---------------------------+
| columns_priv              |
| db                        |
| engine_cost               |
| event                     |
| func                      |
| general_log               |
| gtid_executed             |
| help_category             |
| help_keyword              |
| help_relation             |
| help_topic                |
| innodb_index_stats        |
| innodb_table_stats        |
| ndb_binlog_index          |
| plugin                    |
| proc                      |
| procs_priv                |
| proxies_priv              |
| server_cost               |
| servers                   |
| slave_master_info         |
| slave_relay_log_info      |
| slave_worker_info         |
| slow_log                  |
| tables_priv               |
| time_zone                 |
| time_zone_leap_second     |
| time_zone_name            |
| time_zone_transition      |
| time_zone_transition_type |
| user                      |
+---------------------------+
31 rows in set (0.00 sec)
```

The response is a long list of tables, the mysql database holds.

5. We can also explore the structure of a table using the **DESCRIBE** command. Let's describe the user table as follows:

```
DESCRIBE user;
```

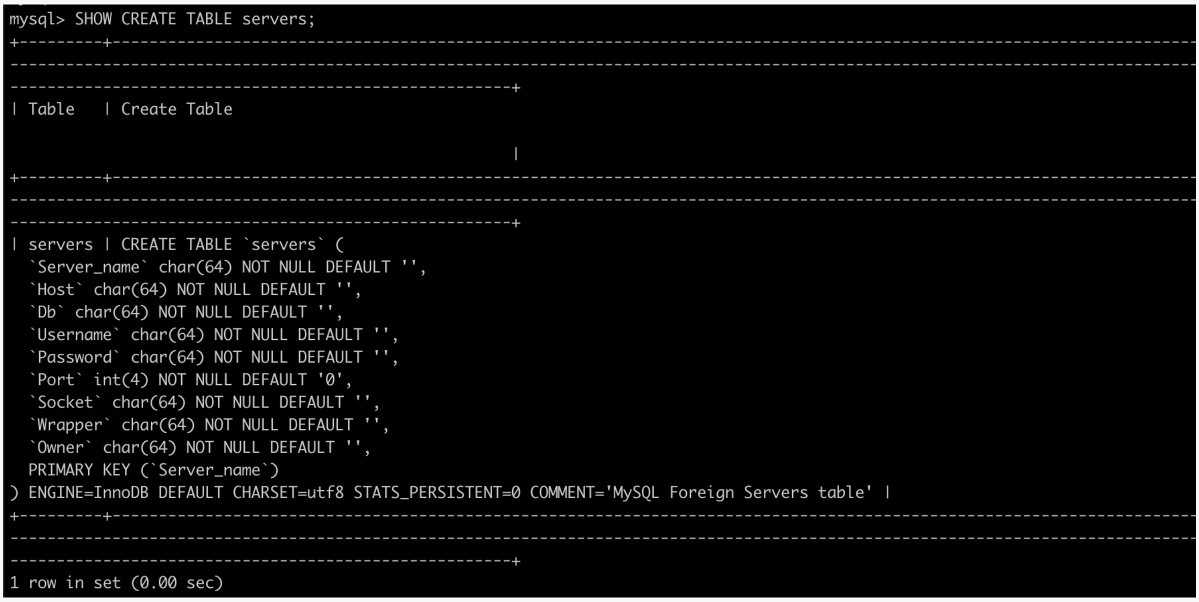


```
mysql> DESCRIBE user;
+------------------------+-----------------------------------+------+-----+-----------------------+-------+
| Field                  | Type                              | Null | Key | Default               | Extra |
+------------------------+-----------------------------------+------+-----+-----------------------+-------+
| Host                   | char(60)                          | NO   | PRI |                       |       |
| User                   | char(32)                          | NO   | PRI |                       |       |
| Select_priv            | enum('N','Y')                     | NO   |     | N                     |       |
| Insert_priv            | enum('N','Y')                     | NO   |     | N                     |       |
| Update_priv            | enum('N','Y')                     | NO   |     | N                     |       |
| Delete_priv            | enum('N','Y')                     | NO   |     | N                     |       |
| Create_priv            | enum('N','Y')                     | NO   |     | N                     |       |
| Drop_priv              | enum('N','Y')                     | NO   |     | N                     |       |
| Reload_priv            | enum('N','Y')                     | NO   |     | N                     |       |
| Shutdown_priv          | enum('N','Y')                     | NO   |     | N                     |       |
| Process_priv           | enum('N','Y')                     | NO   |     | N                     |       |
| File_priv              | enum('N','Y')                     | NO   |     | N                     |       |
| Grant_priv             | enum('N','Y')                     | NO   |     | N                     |       |
| References_priv        | enum('N','Y')                     | NO   |     | N                     |       |
| Index_priv             | enum('N','Y')                     | NO   |     | N                     |       |
| Alter_priv             | enum('N','Y')                     | NO   |     | N                     |       |
| Show_db_priv           | enum('N','Y')                     | NO   |     | N                     |       |
| Super_priv             | enum('N','Y')                     | NO   |     | N                     |       |
| Create_tmp_table_priv  | enum('N','Y')                     | NO   |     | N                     |       |
| Lock_tables_priv       | enum('N','Y')                     | NO   |     | N                     |       |
| Execute_priv           | enum('N','Y')                     | NO   |     | N                     |       |
| Repl_slave_priv        | enum('N','Y')                     | NO   |     | N                     |       |
| Repl_client_priv       | enum('N','Y')                     | NO   |     | N                     |       |
| Create_view_priv       | enum('N','Y')                     | NO   |     | N                     |       |
| Show_view_priv         | enum('N','Y')                     | NO   |     | N                     |       |
| Create_routine_priv    | enum('N','Y')                     | NO   |     | N                     |       |
| Alter_routine_priv     | enum('N','Y')                     | NO   |     | N                     |       |
| Create_user_priv       | enum('N','Y')                     | NO   |     | N                     |       |
| Event_priv             | enum('N','Y')                     | NO   |     | N                     |       |
| Trigger_priv           | enum('N','Y')                     | NO   |     | N                     |       |
| Create_tablespace_priv | enum('N','Y')                     | NO   |     | N                     |       |
| ssl_type               | enum('','ANY','X509','SPECIFIED') | NO   |     |                       |       |
| ssl_cipher             | blob                              | NO   |     | NULL                  |       |
| x509_issuer            | blob                              | NO   |     | NULL                  |       |
| x509_subject           | blob                              | NO   |     | NULL                  |       |
| max_questions          | int(11) unsigned                  | NO   |     | 0                     |       |
| max_updates            | int(11) unsigned                  | NO   |     | 0                     |       |
| max_connections        | int(11) unsigned                  | NO   |     | 0                     |       |
| max_user_connections   | int(11) unsigned                  | NO   |     | 0                     |       |
| plugin                 | char(64)                          | NO   |     | mysql_native_password |       |
| authentication_string  | text                              | YES  |     | NULL                  |       |
| password_expired       | enum('N','Y')                     | NO   |     | N                     |       |
| password_last_changed  | timestamp                         | YES  |     | NULL                  |       |
| password_lifetime      | smallint(5) unsigned              | YES  |     | NULL                  |       |
| account_locked         | enum('N','Y')                     | NO   |     | N                     |       |
+------------------------+-----------------------------------+------+-----+-----------------------+-------+
45 rows in set (0.00 sec)
```

The output will show the various columns the table is made of, the data type of each column, and other related metadata.

6. We can also use the **SHOW** statement to display how the table was created. For instance, the following query shows how the servers table was created:

```
SHOW CREATE TABLE servers;
```

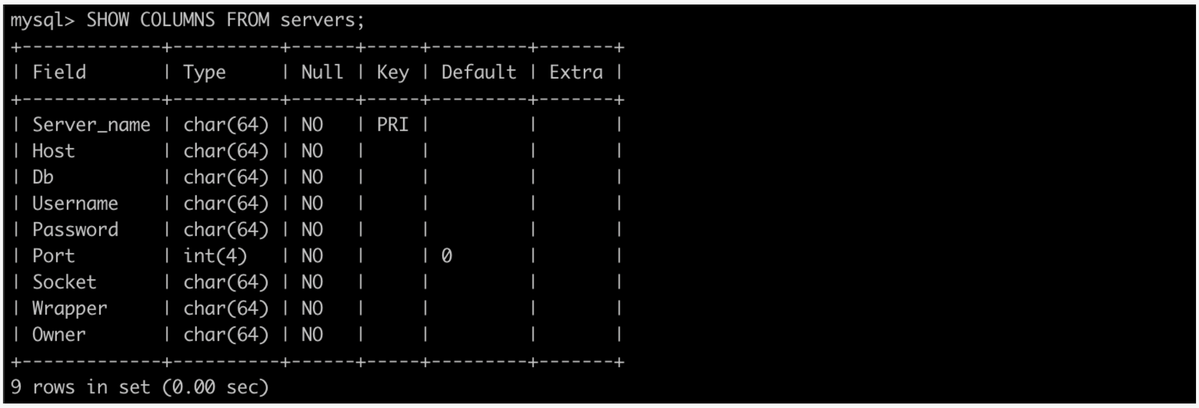


```
mysql> SHOW CREATE TABLE servers;
+---------+-----------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------+
| Table   | Create Table

                                                     |
+---------+-----------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------+
| servers | CREATE TABLE `servers` (
  `Server_name` char(64) NOT NULL DEFAULT '',
  `Host` char(64) NOT NULL DEFAULT '',
  `Db` char(64) NOT NULL DEFAULT '',
  `Username` char(64) NOT NULL DEFAULT '',
  `Password` char(64) NOT NULL DEFAULT '',
  `Port` int(4) NOT NULL DEFAULT '0',
  `Socket` char(64) NOT NULL DEFAULT '',
  `Wrapper` char(64) NOT NULL DEFAULT '',
  `Owner` char(64) NOT NULL DEFAULT '',
  PRIMARY KEY (`Server_name`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 STATS_PERSISTENT=0 COMMENT='MySQL Foreign Servers table' |
+---------+-----------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------+
1 row in set (0.00 sec)
```

7. We can display the column information for a table using the **SHOW**

7. We can display the column information for a table using the **SHOW** statement. For example:

```
SHOW COLUMNS FROM servers;
```



```
mysql> SHOW COLUMNS FROM servers;
+-------------+----------+------+-----+---------+-------+
| Field       | Type     | Null | Key | Default | Extra |
+-------------+----------+------+-----+---------+-------+
| Server_name | char(64) | NO   | PRI |         |       |
| Host        | char(64) | NO   |     |         |       |
| Db          | char(64) | NO   |     |         |       |
| Username    | char(64) | NO   |     |         |       |
| Password    | char(64) | NO   |     |         |       |
| Port        | int(4)   | NO   |     | 0       |       |
| Socket      | char(64) | NO   |     |         |       |
| Wrapper     | char(64) | NO   |     |         |       |
| Owner       | char(64) | NO   |     |         |       |
+-------------+----------+------+-----+---------+-------+
9 rows in set (0.00 sec)
```

This completes a brief exploratory tour of mysql and the various commands we can use to explore it.