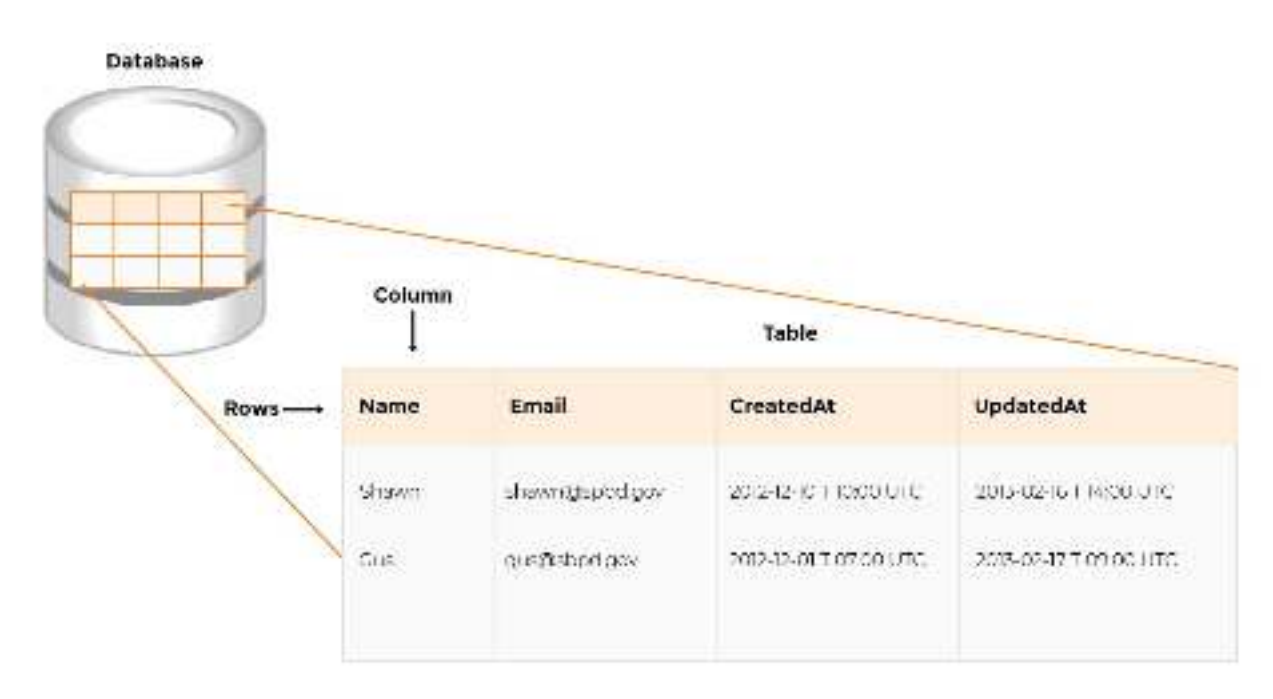
Create Database

This lesson demonstrates how to create a database in MySQL.

Create Database

A database is a container that holds all your tables. A table is a container for a subset of your data. A table holds data organized in rows and columns. Consider a column to be a piece of data that is an attribute of an entity. A row is a set of columns that define attributes of an entity.

The relationships are conceptually shown below:



In this lesson, we'll learn how to use the **CREATE** statement.

We can use the **CREATE** statement to create a database.

> 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/2lesson.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
CREATE DATABASE MovieIndustry;

-- Query 2
CREATE DATABASE IF NOT EXISTS MovieIndustry;

-- Query 3
SHOW DATABASES;

-- Query 4
DROP DATABASE MovieIndustry;
```

Terminal

In this course, we'll create and work with a database related to the movie industry. Without further ado, start executing the following commands in the terminal window.

1. Let's create the database first. We'll name it **MovieIndustry**.

   ```
   CREATE DATABASE MovieIndustry;
   ```

   ```
   mysql> CREATE DATABASE MovieIndustry;
   Query OK, 1 row affected (0.00 sec)
   ```

   If the above statement is executed successfully you should see a message similar to **"Query OK, 1 row affected (0.00 sec)"**. Behind the scenes, a **db.opt** file is created by MySQL that holds database options. The following screenshot shows the **db.opt** file for each of the databases:

   ```
   root@0c82101c601e:/# find . -name db.opt
   ./var/lib/mysql/performance_schema/db.opt
   ./var/lib/mysql/sys/db.opt
   ./var/lib/mysql/mysql/db.opt
   ./var/lib/mysql/MovieIndustry/db.opt
   ```

The contents of the **db.opt** file for the **MovieIndustry** database is shown below:

```
root@0c82101c601e:/# cat ./var/lib/mysql/MovieIndustry/db.opt
default-character-set=latin1
default-collation=latin1_swedish_ci
```

The contents mention the default character set and the collation for the database.

2. If we attempt to re-create an existing database, MySQL will report an error. We can circumvent this error by using the **IF NOT EXISTS** command as shown below:

```
CREATE DATABASE IF NOT EXISTS MovieIndustry;
```

```
mysql> CREATE DATABASE IF NOT EXISTS MovieIndustry;
Query OK, 1 row affected, 1 warning (0.00 sec)
```

The **IF NOT EXISTS** clause is useful when writing scripts that may be invoked repeatedly and will abort when creating a database that already exists.

When we create a database, MySQL creates a physical directory by the same name. Directories are case-sensitive in Linux and Mac, and correspondingly, MySQL will take case into account. On a Linux or Mac, **MovieIndustry** isn't the same as **movieindustry** as shown below:

```
mysql> USE MovieIndustry;
Database changed
mysql> USE movieindustry;
ERROR 1049 (42000): Unknown database 'movieindustry'
```

Windows operating system is case-insensitive, and we can use **MovieIndustry** and **movieindustry** interchangeably.

3. Now you can inspect the database you created by using the following command:

```
SHOW DATABASES;
```

The output lists all the databases in the system and one of them is **MovieIndustry**.

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

4. We can drop a database using the **DROP** statement. All the tables, indexes, and other structures created within the database are also deleted.

```
DROP DATABASE MovieIndustry;
```

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

mysql> DROP DATABASE MovieIndustry;
Query OK, 0 rows affected (0.01 sec)

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