

LIKE Operator

This lesson discusses how to use the LIKE operator for filtering rows.

LIKE Operator

In this lesson we'll learn how to query the data using the **LIKE** operator in the **WHERE** clause.

Example Syntax

```
SELECT col1, col2, ... coln  
  
FROM table  
  
WHERE col3 LIKE "%some-string%"
```

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/9lesson.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  
SELECT * from Actors WHERE FirstName LIKE "Jen%";  
  
-- Query 2  
SELECT * from Actors where FirstName LIKE "Jennifer%";  
  
-- Query 3  
SELECT * from Actors where FirstName LIKE "%";  
  
-- Query 4
```

```

SELECT * from Actors WHERE FirstName LIKE "_enn%";

-- Query 5

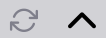
SHOW DATABASES LIKE "M%";

-- Query 6

SHOW TABLES LIKE "A%";

```

● Terminal



The **LIKE** operator works only with string data types and allows us to retrieve rows based on pattern matching on a particular column.

1. Say, you have forgotten the full name of a particular actor but remember that the name starts with the string "Jen". You can check if there is an actor with a name that has "Jen" as the prefix by executing the following query in the terminal.

```

SELECT * from Actors WHERE FirstName LIKE "Jen%";

```

```

mysql> SELECT * from Actors WHERE FirstName LIKE "Jen%";
+-----+-----+-----+-----+-----+-----+-----+
| Id | FirstName | SecondName | DoB          | Gender | MaritalStatus | NetWorthInMillions |
+-----+-----+-----+-----+-----+-----+-----+
| 2 | Jennifer | Aniston    | 1969-11-02 | Female | Single        | 240                |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.01 sec)

```

Note that we use the string "**Jen%**" and not "**Jen**". In fact, if you used the latter, no rows will be matched.

```

mysql> SELECT * from Actors WHERE FirstName LIKE "Jen";
Empty set (0.00 sec)

```

The % symbol is a wildcard character that matches all strings. It can match zero or more characters. For instance, the following query returns one row and the wildcard character matches zero characters.

```

SELECT * from Actors where FirstName LIKE "Jennifer%";

```

```

mysql> SELECT * from Actors where FirstName LIKE "Jennifer%";
+-----+-----+-----+-----+-----+-----+-----+
| Id | FirstName | SecondName | DoB          | Gender | MaritalStatus | NetWorthInMillions |
+-----+-----+-----+-----+-----+-----+-----+
| 2 | Jennifer | Aniston    | 1969-11-02 | Female | Single        | 240                |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

```

As a final example, executing the following query will match all the rows in the table.

```
SELECT * from Actors where FirstName LIKE "%";
```

```
mysql> SELECT * from Actors where FirstName LIKE "%";
+----+-----+-----+-----+-----+-----+-----+
| Id | FirstName | SecondName | DoB          | Gender | MaritalStatus | NetWorthInMillions |
+----+-----+-----+-----+-----+-----+-----+
| 1 | Brad      | Pitt       | 1963-12-18   | Male   | Single        | 240 |
| 2 | Jennifer  | Aniston    | 1969-11-02   | Female | Single        | 240 |
| 3 | Angelina  | Jolie      | 1975-06-04   | Female | Single        | 100 |
| 4 | Johnny    | Depp       | 1963-06-09   | Male   | Single        | 200 |
| 5 | Natalie   | Portman    | 1981-06-09   | Male   | Married       | 60  |
| 6 | Tom       | Cruise     | 1962-07-03   | Male   | Divorced      | 570 |
| 7 | Kylie     | Jenner     | 1997-08-10   | Female | Married       | 1000 |
| 8 | Kim       | Kardashian | 1980-10-21   | Female | Married       | 370 |
| 9 | Amitabh   | Bachchan   | 1942-10-11   | Male   | Married       | 400 |
| 10 | Shahrukh | Khan       | 1965-11-02   | Male   | Married       | 600 |
| 11 | priyanka  | Chopra     | 1982-07-18   | Female | Married       | 28  |
+----+-----+-----+-----+-----+-----+-----+
11 rows in set (0.00 sec)
```

2. We can use the underscore _ character to match exactly one character. For instance, the expression **LIKE "_enn%"** will match the string "Jennifer".

```
SELECT * from Actors WHERE FirstName LIKE "_enn%";
```

```
mysql> SELECT * from Actors WHERE FirstName LIKE "_enn%";
+----+-----+-----+-----+-----+-----+-----+
| Id | FirstName | SecondName | DoB          | Gender | MaritalStatus | NetWorthInMillions |
+----+-----+-----+-----+-----+-----+-----+
| 2 | Jennifer  | Aniston    | 1969-11-02   | Female | Single        | 240 |
+----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

3. The **LIKE** clause can also be used with the **SHOW** command. For example:

```
SHOW DATABASES LIKE "M%";
```

```
mysql> SHOW DATABASES LIKE "M%";
+-----+
| Database (M%) |
+-----+
| MovieIndustry |
+-----+
```

