Self Inner Join

This lesson discusses how to join a table with itself.

Self Inner Join

The simplest join one can use is the inner join. Rows from two tables are joined together using a common column between them.

Syntax

SELECT *

FROM **table1**

INNER JOIN **table1**

ON <**join condition**>;

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/25lesson.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 a INNER JOIN Actors b;

-- Query 2
SELECT * FROM Actors a INNER JOIN Actors b USING(FirstName);

-- Query 3
SELECT * FROM Actors a INNER JOIN Actors b USING(NetWorthInMillions);
```

```
SELECT * FROM Actors a INNER JOIN Actors b USING(NetworthInMillions);
```

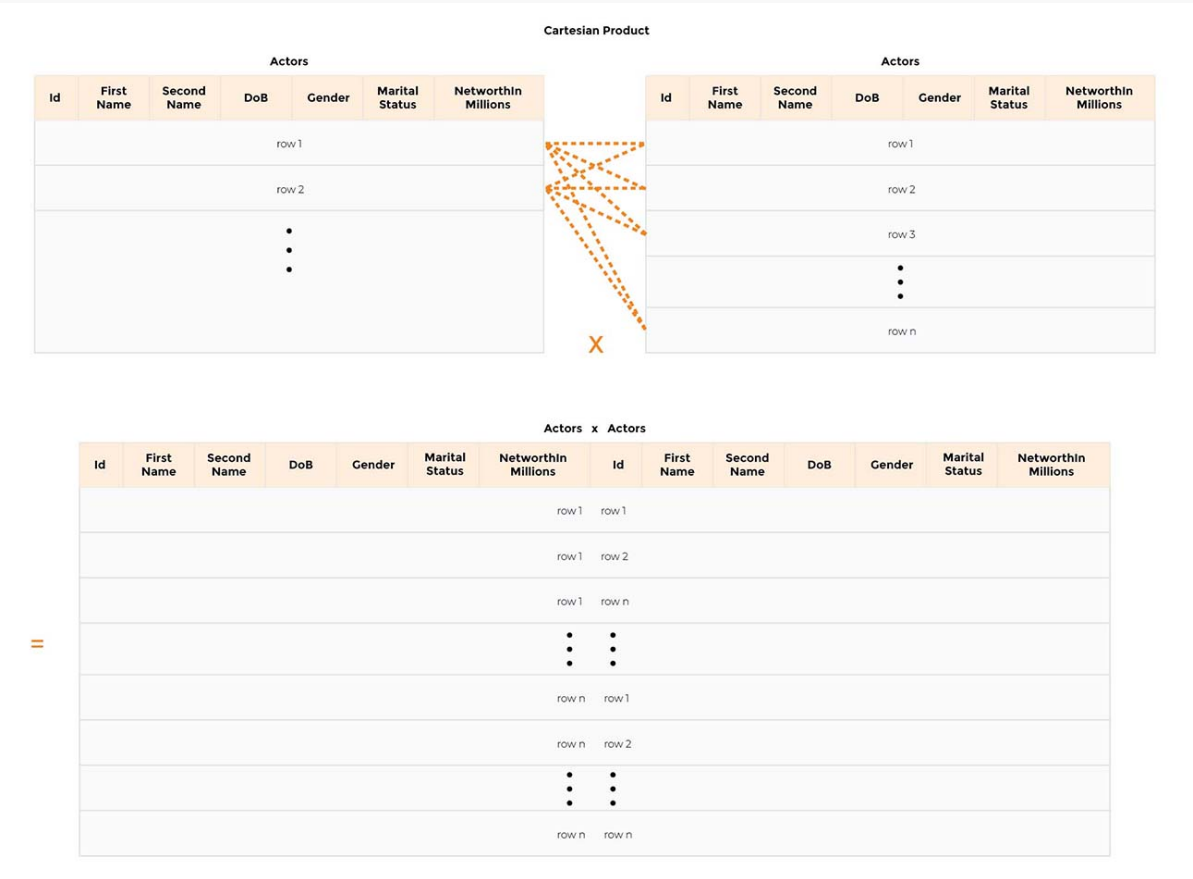
Terminal

1. It may come as a surprise, but we can also join a table with itself. However, we need to use aliases as the **INNER JOIN** clause requires the two tables to be unique.

   ```
   SELECT * FROM Actors a INNER JOIN Actors b;
   ```

   If you run the above query, the result will be a cartesian product, i.e., each row will join with every other row of the second table. The total number of rows in the resulting table will be 121 because the table has 11 rows. The count for the inner join is shown below:

   ```
   mysql> SELECT COUNT(*) FROM Actors a INNER JOIN Actors b;
   +----------+
   | COUNT(*) |
   +----------+
   |      121 |
   +----------+
   1 row in set (0.00 sec)
   ```



2. We can use the **USING** clause to specify the column to join the two tables on. For example:

```
SELECT * FROM Actors a INNER JOIN Actors b USING(FirstName);
```

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

Note we have exactly 11 rows now because each row in the first table matches exactly one row in the second table. However, if we change the query and specify the NetWorthInMillions column in the **USING** clause, we'll get 13 rows in the result because the two rows with value 240 for the NetWorthInMillions column match twice for a total of four rows.

```
SELECT * FROM Actors a INNER JOIN Actors b USING(NetWorthInMillio
ns);
```

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

3. Remember, the **USING** clause defines one or more columns that are in both tables or results and used to join or match rows. Furthermore, if any rows from the two tables don't match, they aren't included in the output. This obviously, will not happen in the case of a self-join.