

🦈 Tutorial (/lesson/introduction) 📱 More Topics (/topic/introduction)

SQL Lesson 2: Queries with constraints (Pt. 1)

Now we know how to select for specific columns of data from a table, but if you had a table with a hundred million rows of data, reading through all the rows would be inefficient and perhaps even impossible.

In order to filter certain results from being returned, we need to use a **WHERE** clause in the query. The clause is applied to each row of data by checking specific column values to determine whether it should be included in the results or not.

Select query with constraints

SELECT column, another_column, ...

FROM mytable

WHERE condition

AND/OR another_condition

AND/OR ...;

More complex clauses can be constructed by joining numerous **AND** or **OR** logical keywords (ie. num_wheels >= 4 AND doors <= 2). And below are some useful operators that you can use for numerical data (ie. integer or floating point):

Operator	Condition	SQL Example
=,!=, < <=, >, >=	Standard numerical operators	col_name != 4
BETWEEN AND	Number is within range of two values (inclusive)	col_name BETWEEN 1.5 AND 10.5
NOT BETWEEN AND	Number is not within range of two values (inclusive)	col_name NOT BETWEEN 1 AND 10
IN ()	Number exists in a list	col_name IN (2, 4, 6)
NOT IN ()	Number does not exist in a list	col_name NOT IN (1, 3, 5)

In addition to making the results more manageable to understand, writing clauses to constrain the set of rows returned also allows the query to run faster due to the reduction in unnecessary data being returned.

Did you know?

As you might have noticed by now, SQL doesn't *require* you to write the keywords all capitalized, but as a convention, it helps people distinguish SQL keywords from column and tables names, and makes the guery easier to read.

Exercise

Using the right constraints, find the information we need from the Movies table for each task below.

Table: Movies

ld	Title	Director	Year	Length_minutes
1	Toy Story	John Lasseter	1995	81
2	A Bug's Life	John Lasseter	1998	95
3	Toy Story 2	John Lasseter	1999	93
4	Monsters, Inc.	Pete Docter	2001	92
5	Finding Nemo	Andrew Stanton	2003	107
6	The Incredibles	Brad Bird	2004	116
7	Cars	John Lasseter	2006	117

SELECT * FROM movies;

RESET

Exercise 2 — Tasks

- 1. Find the movie with a row id of 6
- 2. Find the movies released in the year s between 2000 and 2010
- 3. Find the movies **not** released in the **year** s between 2000 and 2010
- 4. Find the first 5 Pixar movies and their release year

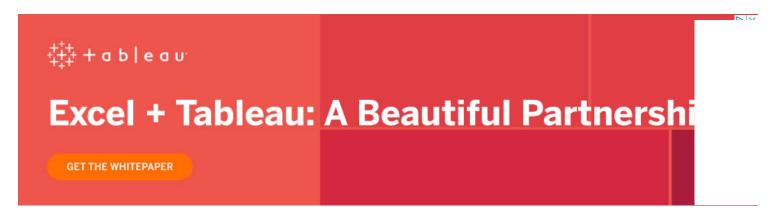
Stuck? Read this task's Solution.
Solve all tasks to continue to the next lesson.

Finish above Tasks (/lesson/select_queries_with_constraints_pt_2)

Next – SQL Lesson 3: Queries with constraints (Pt. 2) (/lesson/select_queries_with_constraints_pt_2)

Previous – SQL Lesson 1: SELECT queries 101 (/lesson/select_queries_introduction)

Find SQLBolt useful? Please consider Donating (\$4) via Paypal to support our site.



Like 1.1K Share Tweet

2019 © SQLBolt Email (mailto:sqlbolt@gmail.com) | Twitter (http://twitter.com/sqlbolt)

Additional Courses Interactive Regular Expressions Lessons (https://regexone.com)