Databases and SQL$^{\beta}$ → Basics of SQL → What is SQL

Theory: What is SQL

🕒 10 minutes 0 / 3 problems solved

Skip this topic Start practicing

5916 users solved this topic. Latest completion was about 2 hours ago.

§1. Introduction

SQL (Structured Query Language) is a domain-specific programming language designed to handle data in tables. It was developed in the 1970s, and to this day SQL-like interfaces are widely used and supported in data management systems, even those that are not based on the concept of a table.

Because of its applicability, you'll probably find SQL quite useful. If you are a software engineer, it's a good idea to learn it because many software systems store and process business data via services that support SQL. For example, the information system backend of an insurance company may use SQL to extract and update data about the clients.

If you are interested in analytics, with the help of SQL you'll be able to easily aggregate data and calculate statistics. Suppose you need to evaluate changes in popularity of the name Jessie between 1920 and 2000 (inclusive) based on census data. In SQL, this task would take only 11 strings of code! You may not understand the code now, just try to read it as a sentence in English. It selects records about individuals named Jessie and born between 1920 and 2000; then it groups records for each pair of year and gender; counts the number of records in each group; and generates, as a result, a table with columns "year", "gender", and "cnt" sorted by year and gender.

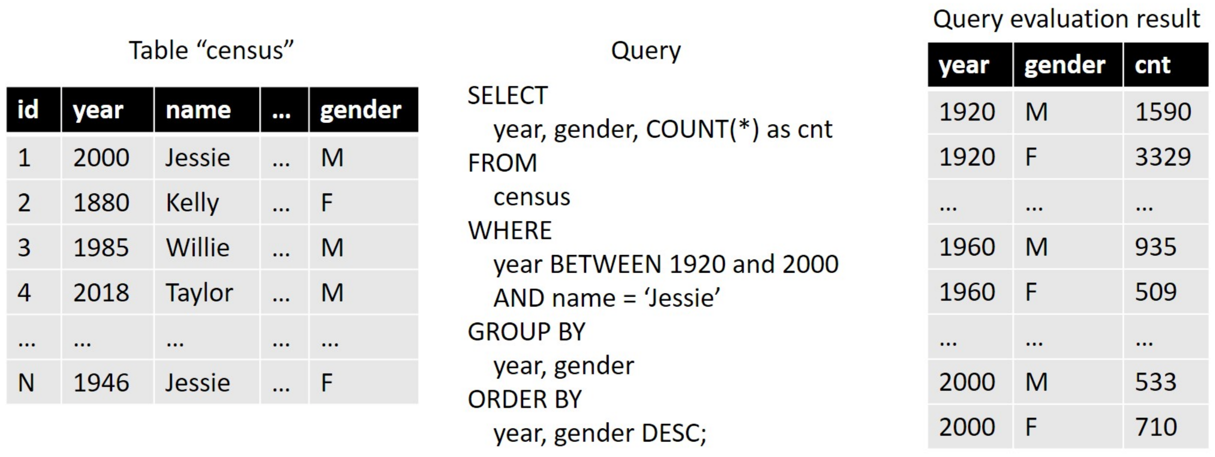

Table "census"

id	year	name	...	gender
1	2000	Jessie	...	M
2	1880	Kelly	...	F
3	1985	Willie	...	M
4	2018	Taylor	...	M
...	...	...	...	...
N	1946	Jessie	...	F

Query

```
SELECT
   year, gender, COUNT(*) as cnt
FROM
   census
WHERE
   year BETWEEN 1920 and 2000
   AND name = 'Jessie'
GROUP BY
   year, gender
ORDER BY
   year, gender DESC;
```

Query evaluation result

year	gender	cnt
1920	M	1590
1920	F	3329
...	...	...
1960	M	935
1960	F	509
...	...	...
2000	M	533
2000	F	710

If you work in a data-based company, know that SQL is basically a standard of data manipulation language.

There's a lot to take from SQL and there's a lot to learn. We suggest to start with the very basics: that is, what SQL stands for. Let's take apart the abbreviation to know what we're dealing with.

§2. S is for Structured

SQL is a language used to extract and update data **structured** as tables. Such data appears in various application areas: for example, Excel sheets with accounting data, census statistics in Google BigQuery, or online stores that utilize a special software system for storing and using tables called Relational Database Management System (RDBMS) which helps process information about goods, orders, and customers.

SQL is designed for tables with the following structure: a table contains **rows** each representing an entity or an object and **columns** with attribute names of these entities. A table cell from row R and column C stores the value of attribute C of entity R. For instance, in table "census" from our example rows represent individuals and each of them having attributes "id", "year", "name", and "gender". For instance, the third row contains data about Willie, a man born in 1985.

Quite often, data is organized in a bunch of tables with names, what is usually called a **database**, and one may address these tables by their names. For example, in a database for an online store, a table "Customers" contains

Current topic:

What is SQL ···

Topic depends on:

✕ Introduction to databases ···

Topic is required for:

Basic data types ···

Object-Relational Mapping(ORM) ···

Table of contents:

data about customers (their names and contacts), while "Orders" stores information about orders (customer, goods, payment details).

§3. Q is for Query

SQL is a programming language with a large feature set for data processing. SQL is a **declarative** language and thus any statement written in it is a **query** that states to the system ***what*** should be done or evaluated, but not ***how***.

Let's focus on the simplest and most basic functionality of data extraction from a table. For example, if we have a table named "Customers", a query that extracts all rows and columns is the following:

```
SELECT * FROM Customers;
```

By default, the best practice in SQL is to terminate all queries with a semicolon. You should use it to terminate statements unless you've set a new statement terminator.

§4. L is for Language

That simple query from the example above may be read as "select everything from customers". SQL was designed to be similar to natural **language**. The declarative nature of SQL hides the complexities from the user and, to some extent, you just declare your will while the system analyses the query, chooses the control flow, and executes it.

Just like any natural language, SQL has a standard of the American National Standards Institute (ANSI) and many dialects implemented by vendors of software that support SQL. Usually, dialects are based on the standard, but they still show differences in some technical details, for example, in processing dates and strings. This means that SQL queries in different dialects are not compatible. However, once you know the SQL basics, you'll be able to adapt to different dialects, like with American and British English.

Here and further, in case of incompatibility between vendors and ANSI standard we provide syntax for MySQL.

§5. Conclusion

Now you know that SQL is a domain-specific declarative language for those who work with structured data.

If you have data that can be organized in tables and you want to know how to select rows and columns according to some criteria, join facts, create groups of entities, calculate statistics and much more, let's dive deeper in SQL.

Report a typo

500 users liked this theory. **3** didn't like it. **What about you?**

😍 🙂 😐 🙁 😡

Start practicing

Comments (10) Hints (0) Useful links (0) Show discussion