PostgreSQL SELECT

Summary: in this tutorial, you are going to learn how to use basic **PostgreSQL SELECT** statement to query data from a table.

One of the most common tasks when you work with PostgreSQL is to query data from tables by using the `SELECT` statement. The `SELECT` statement is one of the most complex statements in PostgreSQL. It has many clauses that you can combine to form a powerful query.

Because of its complexity, we divide the PostgreSQL `SELECT` statement tutorial into many short tutorials so that you can learn each clause of the

SELECT statement easier. The following are the clauses that appear in the SELECT statement:

- Select distinct rows by using DISTINCT operator.
- Filter rows by using WHERE clause.
- Sort rows by using the ORDER BY clause.
- Select rows based on various operator such as BETWEEN, IN and LIKE.
- Group rows into groups by using GROUP BY clause
- Apply condition for groups by using HAVING clause.
- Join to other table by using INNER JOIN, LEFT JOIN, RIGHT JOIN clauses.

In this tutorial, you are going to focus on the SELECT statement that has SELECT and FROM clauses.

PostgreSQL SELECT statement syntax

Let's start with a basic form of the SELECT statement to query data from a table. The following illustrates the syntax of the SELECT statement:

```
SELECT column_1,
       column_2,
       ...
FROM table_name
```

[PostgreSQL Subquery](#)

Let’s examine the `SELECT` statement in more detail:

- First, you specify a list of columns in the table from which you want query data in the `SELECT` clause. You use a comma between each column in case you want to query data from multiple columns. If you want to query data from all column, you can use an asterisk (*) as the shorthand for all columns.
- Second, you indicate the table name after the `FROM` keyword

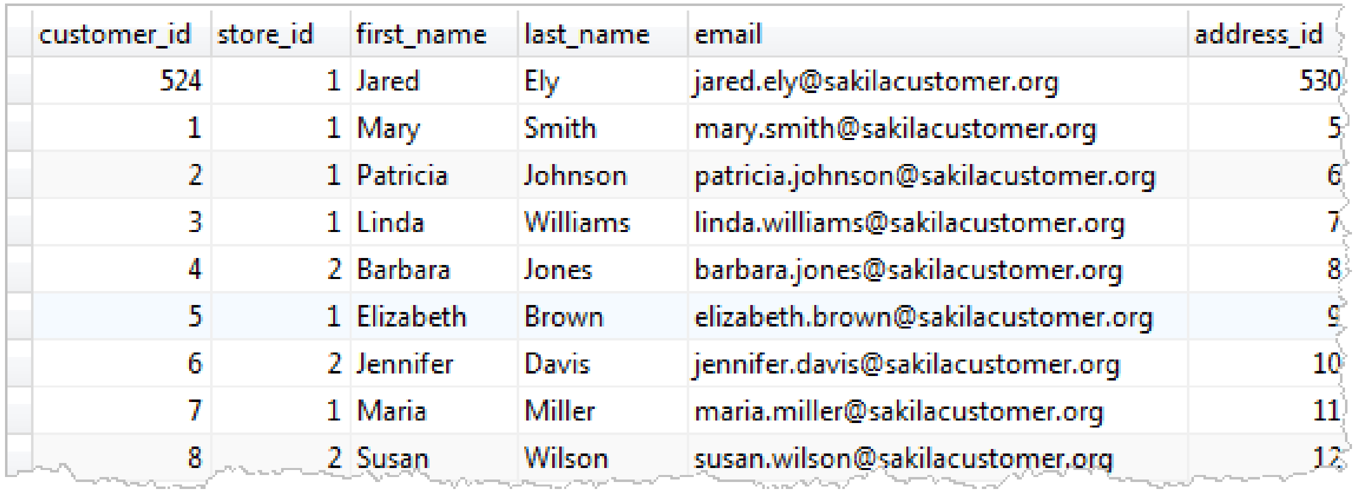
Notice that SQL language is case insensitive. It means if you use `SELECT` or select, the effect is the same. By convention, we will use SQL keywords in upper case to make the code easier to read and stand out clearly.

PostgreSQL SELECT examples

Let’s take a look at several examples of using PostgreSQL `SELECT` statement to query the data from customers table in the sample database.

To query data from all rows and all columns from the customer table, you use the following query:

```
SELECT * FROM customer;
```



customer_id	store_id	first_name	last_name	email	address_id
524	1	Jared	Ely	jared.ely@sakilacustomer.org	530
1	1	Mary	Smith	mary.smith@sakilacustomer.org	5
2	1	Patricia	Johnson	patricia.johnson@sakilacustomer.org	6
3	1	Linda	Williams	linda.williams@sakilacustomer.org	7
4	2	Barbara	Jones	barbara.jones@sakilacustomer.org	8
5	1	Elizabeth	Brown	elizabeth.brown@sakilacustomer.org	9
6	2	Jennifer	Davis	jennifer.davis@sakilacustomer.org	10
7	1	Maria	Miller	maria.miller@sakilacustomer.org	11
8	2	Susan	Wilson	susan.wilson@sakilacustomer.org	12

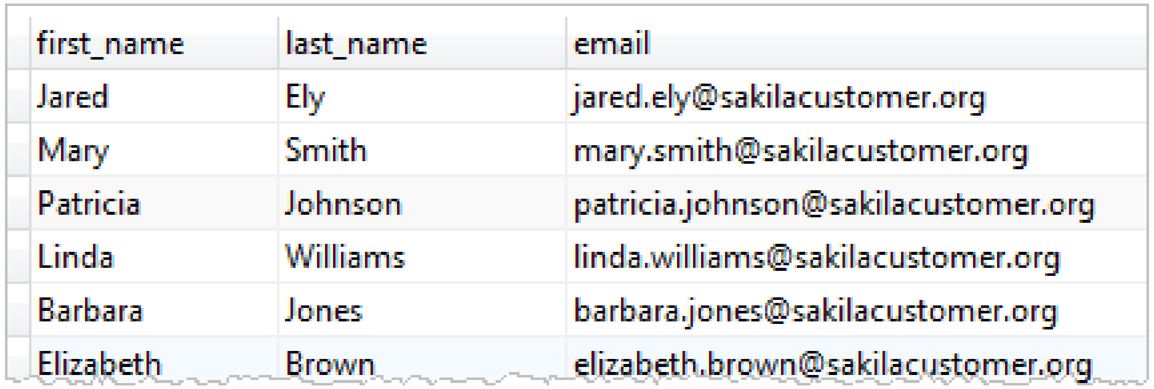
Notice that we have added a semicolon at the end of the `SELECT` statement. The semicolon is not a part of SQL statement. It is only for PostgreSQL to specify the end of the SQL statement.

It is not good practice to use the asterisk (*) in `SELECT` statement. Imagine that you have a large table with many columns, the `SELECT` statement with asterisk (*) will query all the data from the entire columns, which may not necessary. It makes your database server work harder and increase the traffic between the database server and applications. As the result, it slows down your application. Therefore, you should specify the column names in the

SELECT clause whenever possible to get only necessary data from a table.

Suppose you just need to know first name, last name and email of customers, you can list the column names in the SELECT statement as follows:

```
SELECT first_name,
 last_name,
 email
FROM customer;
```



first_name	last_name	email
Jared	Ely	jared.ely@sakilacustomer.org
Mary	Smith	mary.smith@sakilacustomer.org
Patricia	Johnson	patricia.johnson@sakilacustomer.org
Linda	Williams	linda.williams@sakilacustomer.org
Barbara	Jones	barbara.jones@sakilacustomer.org
Elizabeth	Brown	elizabeth.brown@sakilacustomer.org

In this tutorial, you have learned how to use a basic form of PostgreSQL SELECT statement to query data from database table.

« Previous Tutorial:
PostgreSQL Server and Database Objects

Next Tutorial: »
PostgreSQL SELECT DISTINCT

About PostgreSQL Tutorial Website

PostgreSQLTutorial.com is a website dedicated to developers and database administrators who are working on PostgreSQL database management system.

We constantly publish useful PostgreSQL tutorials to keep you up-to-date with the latest PostgreSQL features and technologies. All PostgreSQL tutorials are simple, easy-to-follow and practical.

Recent Tutorials

Developing User-defined Functions Using PostgreSQL CREATE FUNCTION Statement

PostgreSQL Stored Procedures

Introduction to PostgreSQL Stored Procedures

PostgresQL Roles Management

PostgreSQL Restore Database

Backing Up Databases Using PostgreSQL Backup Tools

Deleting Tablespaces Using PostgreSQL DROP TABLESPACE Statement

PostgreSQL ALTER TABLESPACE

Site Info

Home

About Us

Contact Us

Privacy Policy