In PostgreSQL, SERIAL is a data type used for creating auto-incrementing integer columns. It is often used when defining a primary key for a table, ensuring that a unique, sequential integer is automatically assigned to each new row inserted into the table.

Here’s how it works:

* SERIAL is a shorthand for creating an integer column that automatically increments by 1 with each new row.
* When you declare a column as SERIAL, PostgreSQL automatically creates a sequence and links it to the column.

**Example:**

sql

Copy code

CREATE TABLE users (

id SERIAL PRIMARY KEY,

name VARCHAR(100)

);

In this example:

* The id column is defined as SERIAL, so each time a new row is inserted, PostgreSQL will automatically assign a unique integer to the id column.
* The PRIMARY KEY ensures that the id values are unique across the table.

**Variants of SERIAL:**

* SMALLSERIAL: Auto-incrementing 2-byte integer (range 1 to 32,767).
* SERIAL: Auto-incrementing 4-byte integer (range 1 to 2,147,483,647).
* BIGSERIAL: Auto-incrementing 8-byte integer (range 1 to 9,223,372,036,854,775,807).

If you ever need to generate auto-incrementing IDs, SERIAL is a straightforward and commonly used solution in PostgreSQL.