GRANT to Assign Privileges

by Sophia

WHAT'S COVERED

This lesson explores the GRANT and REVOKE commands to assign privileges to roles, in three parts. Specifically, this lesson will cover:

1. Introduction

Each database object (such as a table, view, or index) has an owner, which is typically the user that created it. For most object types, only its owner or a superuser can do anything with the object. If you need some other user or group to be able to interact with the object, you must grant the privileges needed to make that possible.

2. Possible Privileges

There are many types of privileges that you can assign. Each privilege represents a permitted action. Here are some examples:

- SELECT—Allows the role to select from any column or from specific columns listed within a table, view, or sequence. This privilege would also be required if there is a need to reference existing column values in an UPDATE or DELETE statement.
- INSERT—Allows the role to INSERT a new row within a table. If specific columns are listed, only those columns may be inserted into the other columns automatically being set with default values.
- UPDATE—Allows the role to UPDATE a column or a list of columns in a table. If the UPDATE privilege is granted, the SELECT privilege should also be granted since it has to reference the table columns to determine which rows of data should be updated.
- DELETE—Allows the role to DELETE a row from a table. If the DELETE privilege is granted, the SELECT privilege should also be granted since it has to reference the table columns to determine which rows of

data should be removed.

- ALL—This grants access to all available privileges to which object.

Other privileges beyond the scope of this course include TRUNCATE, REFERENCES, TRIGGER, CREATE, CONNECT, TEMPORARY, EXECUTE, and USAGE, that allows interaction with objects.

---

3. Examples

Let us say we have a group role called employee_role. We may want to allow this role to query the employee table (via SELECT) but not allow it to make any changes to the data. As such, we would run the following:

```
GRANT SELECT ON employee TO employee_role;
```

Alternatively, we could grant the employee_role editing and query access to the customer table by specifying additional allowed statements:

```
GRANT SELECT, INSERT, UPDATE, DELETE ON customer TO employee_role;
```

For an admin_role, you might need to grant all privileges. This is done by adding ALL after GRANT:

```
GRANT ALL ON customer TO admin_role;
```

If we wanted to grant access to specific columns, we would add those columns in round brackets after the privilege type. For example, if we wanted to grant UPDATE privileges to the employee_role on the product table, but only on the unit_price, we would do the following:

```
GRANT UPDATE(unit_price) ON product TO employee_role;
```

This grants only the ability to update the price, but none of the other columns.

Now let's say we want to grant SELECT permission for the track table to all users. There is a special group role in PostgreSQL called public. All roles belong to that group. So, you could use this statement:

```
GRANT SELECT ON track TO public;
```

Your turn! Open the SQL tool by clicking on the LAUNCH DATABASE button below. Then, enter one of the examples above and see how it works. Next, try your own choices for which columns you want the query to provide.

In this lesson, you learned how to use GRANT to grant the many types of **possible privileges** for various types of access to database objects such as tables and views. The GRANT statement specifies the type of privilege being granted and the database object to which the privilege is applied.

Instead of assigning privileges directly to users, administrators typically assign privileges to group roles and then assign those group roles to users, streamlining permission management and ensuring a consistent access control strategy.

In the **examples**, you learned how to structure a GRANT statement to assign specific user privileges to access certain objects in certain ways, based on statement names such as SELECT and UPDATE.

Source: THIS TUTORIAL WAS AUTHORED BY DR. VINCENT TRAN, PHD (2020) AND FAITHE WEMPEN (2024) FOR SOPHIA LEARNING. PLEASE SEE OUR **TERMS OF USE**.