Roles & Privileges

ROLES

PostgreSQL uses **roles to represent user accounts**. *It doesn't use the user concept like other database systems*. Typically, roles can log in are called login roles. They are equivalent to users in other database systems. When roles contain other roles, they are called group roles.

- `SELECT * FROM pg_roles;`
- `CREATE ROLE name [ [ WITH ] option [ ... ] ]`
- `where option can be:`

```
        SUPERUSER | NOSUPERUSER
    | CREATEDB | NOCREATEDB
    | CREATEROLE | NOCREATEROLE
    | LOGIN | NOLOGIN
    | CONNECTION LIMIT connlimit
    | [ ENCRYPTED ] PASSWORD 'password' | PASSWORD NULL
    | VALID UNTIL 'timestamp'
```

documentation

GRANT

After creating a role with the LOGIN attribute, the role can log in to the PostgreSQL database server. However, it cannot do anything to the database objects like tables, views, functions, etc.

Need to be **grant**ed privileges on the database objects to the user role by using the **GRANT statement**.

- ```
  GRANT privilege_list | ALL
        ON  table_name
        TO  role_name;
  ```

REVOKE

The **REVOKE** statement revokes previously granted privileges on database objects from a **role**.

- ```
  REVOKE privilege | ALL
  ON TABLE table_name |  ALL TABLES IN SCHEMA schema_name
  FROM role_name;
  ```

Postgresql documentation

DROP ROLE

- `DROP ROLE [IF EXISTS] target_role;`

[Postgresql documentation](#)

Interview Questions