How to Use info.GetPrivileges

# Summary

The `info.GetPrivileges` function is a reusable PostgreSQL stored procedure that returns all schema-level, database-level, and default ACL privileges for roles within the current database. It supports optional filtering by a specific role and dynamically adapts based on the server's version, excluding unsupported default ACLs if on an older version.

# TL;DR

Use this function to audit database and schema privileges per role.  
Call it from any user-connected database (except 'postgres' or 'rdsadmin').  
Supports optional filtering by role name.

# How to Use

1. 🏁 Ensure the function has been deployed by running the script `info.GetPrivileges.sql` in your target database.

2. 💡 Call the function without parameters to return all privileges across all roles:

```sql  
 SELECT \* FROM info.GetPrivileges();  
 ```

3. 🎯 Call it with a specific role to filter the results:

```sql  
 SELECT \* FROM info.GetPrivileges('dba\_team');  
 ```

4. 📤 The output includes columns:

- `object\_scope`: The schema or database name  
 - `grantee`: The role receiving the privilege  
 - `privilege`: The granted right (e.g., CREATE, USAGE, TEMP)  
 - `inherited\_role`: If applicable, the role it inherits from  
 - `source`: The origin of the privilege (e.g., actual\_database\_privilege)

5. 🔒 This function skips installation in system databases (`postgres`, `rdsadmin`) and places the function in the appropriate schema based on database context.