DB2 - ROLES

Introduction

A role is a database object that groups multiple privileges that can be assigned to users, groups, PUBLIC or other roles by using GRANT statement.

Restrictions on roles

- A role cannot own database objects.
- Permissions and roles granted to groups are not considered when you create the following database objects.
  - Package Containing static SQL
  - Views
  - Materialized Query Tables *MQT*
  - Triggers
  - SQL Routines

Creating and granting membership in roles

Syntax: [To create a new role]

```
db2 create role <role_name>
```

Example: [To create a new role named 'sales' to add some table to be managed by some user or group]

```
db2 create role sales
```

Output:

```
DB20000I The SQL command completed successfully.
```

Granting role from DBADM to a particular table

Syntax: [To grant permission of a role to a table]

```
db2 grant select on table <table_name> to role <role_name>
```

Example: [To add permission to manage a table 'shope.books' to role 'sales']

```
db2 grant select on table shope.books to role sales
```

Output:

```
DB20000I  The SQL command completed successfully.
```

Security administrator grants role to the required users. *Before you use this command, you need to create the users.*

Syntax: [To add users to a role]

```
db2 grant role <role_name> to user <username>
```

Example: [To add a user 'mastanvali' to a role 'sales']

```
db2 grant sales to user mastanvali
```

Output:

```
DB20000I   The SQL command completed successfully.
```

Role hierarchies

For creating a hierarchies for roles, each role is granted permissions/ membership with another role.

Syntax: [before this syntax create a new role with name of "production"]

```
db2 grant role <roll_name> to role <role_name>
```

Example: [To provide permission of a role 'sales' to another role 'production']

```
db2 grant sales to role production
```