




SQL | Creating Roles

 shreyanshi_arun

[Read](#) [Discuss](#) [Courses](#) [Practice](#)

A role is created to ease the setup and maintenance of the security model. It is a named group of related privileges that can be granted to the user. When there are many users in a [database](#) it becomes difficult to grant or revoke privileges to users. Therefore, if you define roles:

- 1. You can grant or revoke privileges to users, thereby automatically granting or revoking privileges.
- 2. You can either create Roles or use the system roles pre-defined.

Some of the privileges granted to the system roles are as given below:

System Roles	Privileges Granted to the Role
Connect	Create table, Create view, Create synonym, Create sequence, Create session, etc.
Resource	Create Procedure, Create Sequence, Create Table, Create Trigger etc. The primary usage of the Resource role is to restrict access to database objects.
DBA	All system privileges

Creating and Assigning a Role

First, the (Database Administrator)DBA must create the role. Then the DBA can assign privileges to the role and users to the role.

Syntax:

AD

```
CREATE ROLE manager;
```

Role created.

Arguments

- *role_name*: Is the name of the role to be created
- *owner_name*: AUTHORIZATION: If no user is specified, the user who executes CREATE ROLE will be the owner of the role. The role's members can be added or removed at the discretion of the role's owner or any member of an owning role.

Permissions

It requires either membership in the fixed database role db_securityadmin or the CREATE ROLE permission on the database. The following authorizations are also necessary when using the AUTHORIZATION option:

1. It takes IMPERSONATE permission from that user in order to transfer ownership of a role to another user.
2. It takes membership in the recipient role or the ALTER permission on that role to transfer ownership of one role to another.
3. An application role must have [ALTER](#) permission in order to transfer ownership of a role to it.

In the syntax: 'manager' is the name of the role to be created.

- Now that the role is created, the DBA can use the GRANT statement to assign users to the role as well as assign privileges to the role.
- It's easier to GRANT or REVOKE privileges to the users through a role rather than assigning a privilege directly to every user.
- If a role is identified by a password, then GRANT or REVOKE privileges have to be identified by the password.

Grant Privileges To a Role

```
GRANT create table, create view  
TO manager;  
Grant succeeded.
```

Grant a Role To Users

```
GRANT manager TO SAM, STARK;  
Grant succeeded.
```

Revoke Privilege from a Role

```
REVOKE create table FROM manager;
```

Drop a Role

```
DROP ROLE manager;
```

Explanation

Firstly it creates a manager role and then allows managers to create tables and views. It then grants Sam and Stark the role of managers. Now Sam and Stark can create tables and views. If users have multiple roles granted to them, they receive all of the privileges associated with all of the roles. Then create table privilege is removed from the role 'manager' using Revoke. The role is dropped from the database using drop.

Last Updated : 30 May, 2023

95

Similar Reads

1. Difference between Structured Query Language (SQL) and Transact-SQL (T-SQL)
2. Configure SQL Jobs in SQL Server using T-SQL
3. SQL vs NO SQL vs NEW SQL
4. Database Roles in CQL (Cassandra Query Language)
5. Granting Permissions to Roles in Cassandra
6. Key Roles for Data Analytics project
7. Privilege and Roles in DBMS