Oracle SET ROLE: Enable and Disable Roles for Your Current Session

Summary: In this lab, you will learn how to use the Oracle `SET ROLE` statement to enable or disable roles for your current session.

Oracle SET ROLE statement

The `SET ROLE` statement allows you to enable and disable **roles** for your current session.

It is possible to enable multiple roles at once like the following statement:

```
SET ROLE role1, role2, ...;
```

Or

```
SET ROLE
    role1,
    role2 IDENTIFIED BY password,
    ...;
```

Note that you cannot enable more than 148 user-defined roles at one time.

To enable all roles previously granted to your account, you use the following syntax:

```
SET ROLE ALL;
```

Note the `SET ROLE ALL` statement will not enable the roles with passwords, which have been granted directly to you.

To disable all roles including the `DEFAULT` role, you use the following statement:

```
SET ROLE NONE;
```

The `session_roles` data dictionary view provides the currently enabled roles in your current session:

```
SELECT * FROM session_roles;
```

Oracle SET ROLE statement examples

First, create a user named `scott` and grant him the `CREATE SESSION` privilege so that he can log in to the database:

```
CREATE USER scott IDENTIFIED BY abcd1234;

GRANT CREATE SESSION TO scott;
```

Second, create two roles called `warehouse_manager` and `warehouse_staff`:

```
CREATE ROLE warehouse_staff;
CREATE ROLE warehouse_manager IDENTIFIED BY xyz123;
```

Third, grant object privileges on `inventories` table to the `warehouse_staff` role:

```
CREATE TABLE inventories(
    id INT PRIMARY KEY,
    inventory_name varchar(50)
);

GRANT SELECT, INSERT, UPDATE, DELETE
ON inventories
TO warehouse_staff;
```

Fourth, grant object privileges on `warehouses` table to the `warehouse_manager` role:

```
CREATE TABLE warehouses(
    id INT PRIMARY KEY,
    warehouses_name varchar(50),
    location varchar(50)
);


GRANT SELECT, INSERT, UPDATE, DELETE
ON warehouses
TO warehouse_manager;
```

Fifth, grant privileges of the `warehouse_staff` role to `warehouse_manager` role:

```
GRANT warehouse_staff to warehouse_manager;
```

Sixth, grant the role `warehouse_manager` to `scott`:

```
GRANT warehouse_manager TO scott;
```

Seventh, log in to the database as `scott` and enable the `warehouse_manager` role:

```
SET ROLE warehouse_manager IDENTIFIED BY xyz123;
```

Eighth, view the current roles of `scott`:

```
SELECT * FROM session_roles;
```

Here is the output:

```
ROLE

WAREHOUSE_STAFF
WAREHOUSE_MANAGER
```

The user `scott` has two roles: `warehouse_manager` which was directly granted and `warehouse_staff` that was indirectly granted via the `warehouse_manager` role.

Ninth, to disable all roles of `scott`, you use this statement:

```
SET ROLE NONE;
```

In this lab, you have learned how to use the Oracle `SET ROLE` statement to enable and disable roles for your current session.