

Oracle DROP USER Statement

Summary: In this lab, you will learn how to use the Oracle `DROP USER` to delete a user from the database.

Oracle DROP USER statement

The `DROP USER` statement allows you to delete a user from the Oracle Database. If the user has schema objects, the `DROP USER` statement also can remove all the user's schema objects along with the user.

The following illustrates the basic syntax of the `DROP USER` statement:

```
DROP USER username [CASCADE];
```

In this syntax, you need to specify the user that you want to drop after the `DROP USER` keywords.

If the user whose schemas contain objects such as views and tables, you need to delete all schema objects of the user first and then drop the user.

Deleting all schema objects of the users first before removing the user is quite tedious. Therefore, Oracle provides you with the `CASCADE` option.

If you specify the `CASCADE` option, Oracle will remove all schema objects of the user before deleting the user.

If the schema objects of the dropped user are referenced by objects in other schemas, Oracle will invalidate these objects after deleting the user.

If a table of the dropped user is referenced by materialized views in other schemas, Oracle will not drop these materialized views. However, the materialized views can no longer be refreshed because the base table doesn't exist anymore.

Note that Oracle does not drop roles created by the user even after it deletes the user.

Notice that if you attempt to delete the user `SYS` or `SYSTEM`, your database will corrupt.

Oracle DROP USER statement

Let's take some examples of removing a user from the database.

1) Using Oracle DROP USER to remove a user that has no schema object

First, log in to the Oracle database using the user `sys` using SQL Developer:

Second, create a new user called `foo` :

```
CREATE USER foo IDENTIFIED BY abcd1234;
```

Third, drop the user `foo` using the `DROP USER` statement:

```
DROP USER foo;
```

You should see the following message:

```
User dropped.
```

Because the user `foo` has no schema objects, you could delete it without specifying the `CASCADE` option.

2) Using Oracle DROP USER to delete a user that has schema objects

First, create a new user called `bar` and grant the `CREATE SESSION` and `CREATE TABLE` system privileges to the user:

```
CREATE USER bar
  IDENTIFIED BY abcd1234
  QUOTA 5m ON users;

GRANT
  CREATE SESSION,
  CREATE TABLE
TO bar;
```

Second, use the user `bar` to log in to the Oracle database with SQL Developer

Third, create a new table named `t1` in the `bar` user's schema:

```
CREATE TABLE t1 (
  id NUMBER PRIMARY KEY,
  v VARCHAR2(100) NOT NULL
);

INSERT INTO t1(id,v)
VALUES (1, 'A');
```

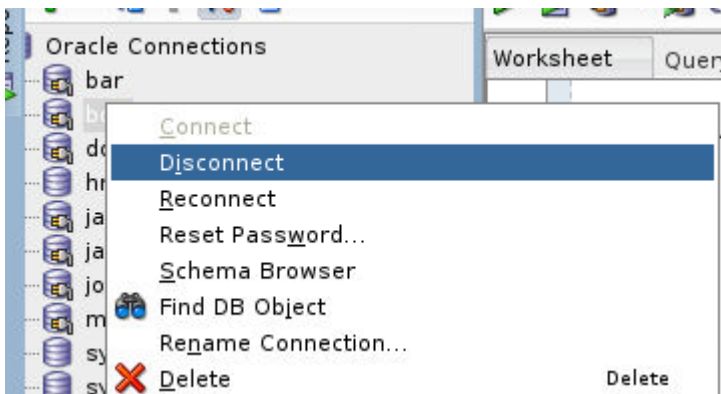
Fourth, go back to the session of the user `sys` and drop user `bar` :

```
DROP USER bar;
```

Oracle issued the following error:

```
ORA-01940: cannot drop a user that is currently connected
```

Fifth, end the user `bar`'s session first:



And issue the `DROP USER` statement again in the user `sys` session:

```
DROP USER bar;
```

Oracle issued the following message:

```
ORA-01922: CASCADE must be specified to drop 'BAR'
```

You could not delete the user `bar` without specifying `CASCADE` because the user `bar` has the table `t1` as a schema object.

Seventh, use the `DROP USER` statement with `CASCADE` option to delete the user `bar` :

```
DROP USER bar CASCADE;
```

Oracle could delete the user `bar` and also the table `t1` .

In this lab, you have learned how to use the Oracle `DROP USER` statement to delete a user and optionally remove all objects in the user's schema from the Oracle database.