**Virtual Private Database (VPD) by Context**

**TASK:**

In this project, you will implement a VPD case. Tasks are as follows:

1. Connect to user SYSTEM. When you need to connect to SYSTEM schema.

2. Grant "DBA" role to user DBSEC. This is the same user created on a previous project. If for some reason, the user was dropped, you need to recreate the user before proceeding.

3. Grant "CREATE SESSION" and "CREATE TABLE" privileges to user VPD\_CLERK1.

4. Connect to user DBSEC/secc$1new. Please note that the password was from our previous project. If for some reason, the user was dropped or changed, you need to change it back to the above user name and password.

5. Create "PRODUCTS" table with the following columns. Please note slight changes in the table from previous project.

Column Name Data Type

PRODUCT\_ID NUMBER (4) NOT NULL,

CATEGORY\_ID NUMBER (2) NULL,

PRODUCT\_NAME VARCHAR2(80) NULL,

PRODUCT\_DESC VARCHAR2(512) NULL,

STATUS CHAR (1) NULL,

COMMENTS VARCHAR2(1024) NULL,

UPD\_TIME DATE NULL,

UPD\_USER VARCHAR2(30) NULL,

UPD\_STAT CHAR (1) NULL

6. Populate the PRODUCTS table using the data in “Supplement for Project”.

7. Query the table "PRODUCTS" to show the total number of rows from each user.

8. Grant SELECT, DELETE, INSERT and UPDATE privileges on PRODUCTS table to user VPD\_CLERK1.

9. Create a policy function, named "DBSEC\_ROW\_OWNER\_FUNCTION" so that only the data that belong to the current user will be selected. In other words, you need to generate a predicate "CTL\_UPD\_USER=USER." Please note the owner of the table (DBSEC) has to be able to access its own table. (Please note the double quotation mark (") is not a part of the specification.)

10. Add the policy using DBMS\_RLS.ADD\_POLICY function. The policy will be named as "MY\_OWNER\_POLICY."

11. Connect to VPD\_CLERK1/Jessie#22.

12. Query the "PRODUCTS" table to show the total number of rows from each user. If everything is successful, you will see only one row.