

Ingénierie des Bases de Données

TP n°3

. Exercice 1 : Gestion des utilisateurs et des privilèges :

1- Afficher les utilisateurs de la base de données.

- `SELECT * FROM all_users;`

2- Afficher les objets de l'utilisateur 'HR'

- `SELECT * FROM all_objects where owner = 'HR';`

3- Afficher le rôle de l'utilisateur 'SYSTEM'

- `SELECT * FROM dba_role_privs WHERE grantee = 'SYSTEM';`

4- Créer un utilisateur 'etud1' identifié par 'isims'. Il doit changer son mot de passe à la première connexion.

- `CREATE USER etud1 IDENTIFIED BY isims PASSWORD EXPIRE;`

5- Créer un utilisateur 'etud2' ayant un compte bloqué.

- `CREATE USER etud2 IDENTIFIED BY isims ACCOUNT LOCK;`

6- Changer le mot de passe de 'etud1' par 'sfax'

- `ALTER USER etud1 IDENTIFIED BY sfax;`

7- Débloquer l'utilisateur 'etud2'

- `ALTER USER etud2 ACCOUNT UNLOCK;`

8- Connecter en tant que l'utilisateur 'etud1' et créez la table PERSONNEL

PERSONNEL (numEmp NUMBER, nomEmp VARCHAR (20), fonction VARCHAR(20))

- `CREATE TABLE PERSONNEL (numEmp NUMBER(50), nomEmp VARCHAR(20), fonction VARCHAR(20));`

- Que se passe-t-il lors de la connexion ? Comment résoudre le problème ?

- Lorsque l'utilisateur 'etud1' tente de créer la table 'PERSONNEL' et rencontre un problème d'autorisation, la solution consiste à accorder les privilèges nécessaires à 'etud1' :

- GRANT CREATE SESSION, CREATE TABLE TO etud1;

9- Insérer dans la table PERSONNEL les employés de la table EMP (créée lors du TP1)

**- INSERT INTO PERSONNEL (numEmp, nomEmp, fonction)
SELECT empno, ename, job FROM EMP;**

10- Ajouter la contrainte clé étrangère dans PERSONNEL : numEmp référence empno dans EMP

**- ALTER TABLE PERSONNEL ADD CONSTRAINT numEMPky FOREIGN
KEY (numEmp) REFERENCES EMP(empno);**

11- A partir de l'utilisateur 'etud1', transmettre les privilèges systèmes de connexion et de création de tables à l'utilisateur 'etud2'

- GRANT CONNECT, CREATE TABLE TO etud2;

12- A partir de 'etud1', transmettre à 'etud2' le privilège objet de visualisation de la table EMP.

- GRANT SELECT ON EMP TO etud2;

13- A partir de l'utilisateur SYSTEM, accordez à tous les utilisateurs le privilège système de créer des vues et à l'utilisateur 'etud1' le privilège objet pour pouvoir supprimer de la table EMP

- GRANT CREATE VIEW TO PUBLIC;

14- Visualiser tous les privilèges accordés (objets et systèmes) à l'utilisateur 'etud1' à partir du dictionnaire de données.

- `SELECT * FROM dba_tab_privs WHERE grantee = 'ETUD1';`
- `SELECT * FROM dba_sys_privs WHERE grantee = 'ETUD1';`

15- Supprimer de tous les utilisateurs le privilège de visualisation de la table EMP.

- `REVOKE SELECT ON EMP FROM PUBLIC;`

Exercice 2 : Gestion des rôles :

En tant qu'administrateur :

1- Créer l'utilisateur ayant comme nom : votre prénom et comme mot de passe votre numéro d'inscription.

- `CREATE USER AYOUB IDENTIFIED BY A22LSI165;`

2- Créer le rôle ROLE1, puis lui attribuer les privilèges système nécessaires pour se connecter à ORACLE et pour créer des tables dans n'importe quel schéma.

- `CREATE ROLE ROLE1;`
- `GRANT CREATE SESSION, CREATE TABLE TO ROLE1;`

3- Créer le rôle ROLE2 identifié par ora11, puis lui attribuer les privilèges système nécessaires pour créer des vues, des séquences, des procédures et des bases de données. Ajouter aussi à ROLE2 les privilèges lui permettant d'effectuer toutes les manipulations (select, insert, update, delete) sur la table PERSONNEL. Vérifier les privilèges attribués à ROLE1 et ROLE2 en interrogeant les vues nécessaires du dictionnaire de données.

- `CREATE ROLE ROLE2 IDENTIFIED BY ora11;`

- GRANT CREATE VIEW, CREATE SEQUENCE, CREATE PROCEDURE, CREATE DATABASE TO ROLE2;
 - GRANT SELECT, INSERT, UPDATE, DELETE ON PERSONNEL TO ROLE2;
- 4- Attribuer les rôles ROLE1 et ROLE2 à l'utilisateur que vous avez créé. Cet utilisateur doit pouvoir transmettre les rôles à d'autres utilisateurs. Consulter la vue correspondante pour vérifier l'attribution de ces rôles.

- GRANT ROLE1, ROLE2 TO AYOUB WITH ADMIN OPTION;

5- Le rôle ROLE1 doit être un rôle par défaut pour l'utilisateur

- ALTER USER AYOUB DEFAULT ROLE ROLE1;

En tant qu'utilisateur :

6- Créer la table produit (cod_prod, desig_prod, prix_prod, qtite_stock)

- CREATE TABLE produit (cod_prod NUMBER, desig_prod VARCHAR2(70), prix_prod NUMBER, qtite_stock NUMBER);

7- Créer la vue vue_nondisp qui donne les produits dont les quantités stockées sont épuisées. Que remarquez-vous ?

- CREATE VIEW vue_nondisp AS SELECT * FROM produit WHERE qtite_stock = 0;
- ..

8- Activer les rôles qui ne sont pas des rôles par défaut puis recréer la vue vue_nondisp.

- SET ROLE ALL;
- CREATE OR REPLACE VIEW vue_nondisp AS SELECT * FROM produit WHERE qtite_stock = 0;

9- Afficher tous les rôles attribués à cet utilisateur.

- `SELECT * FROM DBA_ROLE_PRIVS WHERE GRANTEE = 'AYOUB' ;`

10- Désactiver tous les rôles puis supprimer le role2.

- `SET ROLE NONE;`
- `DROP ROLE ROLE2;`
-