

# TP2 Oracle DBA : Droits et Privilèges

1. Connectez-vous avec le compte system.

```
C:\Windows\system32>sqlplus /nolog

SQL*Plus: Release 19.0.0.0.0 - Production on Dim. F vr. 27 16:41:53 2022
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

SQL> conn sys /as sysdba
Entrez le mot de passe :
Connect .
SQL> _
```

2. Cr er un nouvel utilisateur student\_1 de fa on   ce qu'il devra changer son mot de passe apr s sa premi re connexion.

```
SQL> alter session set "_ORACLE_SCRIPT"=true ;

Session modifi e.

SQL> create user student_1 identified by student1123 password expire ;

Utilisateur cr  .

SQL>
```

3. Essayer de vous connecter avec le compte student\_1. Qu'est-ce que vous remarquez ?

```
SQL> conn student_1 /student1123
ERROR:
ORA-01045: l'utilisateur STUDENT_1 n'a pas le privil ge CREATE SESSION ;
connexion refus e

Avertissement : vous n' tes plus connect    ORACLE.
SQL>
```

4. Donner le droit n cessaire   l'utilisateur student\_1 pour se connecter   son compte.

```
SQL> grant create session to student_1 ;

Autorisation de privil ges (GRANT) accept e.

SQL> _
```

5. Connectez-vous avec le compte student\_1. Créer une table. Qu'est-ce que vous remarquez ?

```
SQL> conn student_1 /student1123
ERROR:
ORA-28001: le mot de passe est expiré
```

```
Modification de mot de passe pour student_1
Nouveau mot de passe :
Ressaisir le nouveau mot de passe :
Mot de passe modifié
Connecté.
SQL>
```

✚ Après la première connexion il est nécessaire de changer le mot de passe

```
SQL> create table stduents( nom varchar(20) , prenom varchar(20));
create table stduents( nom varchar(20) , prenom varchar(20))
*
ERREUR Ó la ligne 1 :
ORA-01031: privilèges insuffisants

SQL>
```

✚ On remarque que l'utilisateur student\_1 n'a pas le droit de créer une table .

Une erreur est produite : **privilèges insuffisants** .

6. Donner les droits system à l'utilisateur student\_1.

```
SQL> conn sys /as sysdba ;
Entrez le mot de passe :
Connecté.
SQL> grant create table , create view to student_1 ;

Autorisation de privilèges (GRANT) acceptée.
```

7. Créer une table et essayer d'insérer des enregistrements. Que se passe-t-il ?

```
SQL> conn student_1 /student1123
Connecté.
SQL> create table etudiants ( nom varchar(20) , prenom varchar(20));

Table créée.

SQL>
```

```
SQL> insert into etudiants(nom,prenom) values ('belouad','saad');
insert into etudiants(nom,prenom) values ('belouad','saad')
*
ERREUR Ó la ligne 1 :
ORA-01950: pas de privilèges sur le tablespace 'USERS'
```

✚ On remarque que l'utilisateur student\_1 n'a pas le droit d'insérer

8. Donner les droits objet à l'utilisateur student\_1.

```
SQL> grant unlimited tablespace to student_1;

Autorisation de privilèges (GRANT) acceptée.

SQL>
```

```
SQL> grant insert on student_1.etudiants to student_1;

Autorisation de privilèges (GRANT) acceptée.

SQL>
```

```
SQL> grant insert on student_1.etudiants to student_1;

Autorisation de privilèges (GRANT) acceptée.

SQL> conn student_1 /student1123
Connecté.
SQL> insert into etudiants(nom,prenom) values ('belaouad','saad');

1 ligne créée.
```

## 9. Créer un rôle role\_1.

```
SQL> create role role_1;

Rôle créé.

SQL> _
```

## 10. Donner les droits CREATE TABLE, CREATE SESSION et les droits objets à ce rôle.

```
SQL> grant create table , create session to role_1;

Autorisation de privilèges (GRANT) acceptée.

SQL> _
```

```
SQL> grant create table , create session , insert any table , delete any table , update any table to role_1;

Autorisation de privilèges (GRANT) acceptée.

SQL> _
```

## 11. Créer deux utilisateurs student\_2 et student\_3. Leurs attribué le rôle role\_1

```
SQL> create user student_2 identified by student2123 ;

Utilisateur créé.

SQL> create user student_3 identified by student3123;

Utilisateur créé.

SQL> _
```

```
SQL> grant role_1 to student_2 ;

Autorisation de privilèges (GRANT) acceptée.

SQL> grant role_1 to student_3 ;

Autorisation de privilèges (GRANT) acceptée.

SQL> _
```

## 12. Créer une table emp (id, nom, prenom) dans le compte student\_2 et insérer quelques enregistrements.

```
SQL> conn student_2 /student2123
Connecté.
SQL> insert into emp (id , nom , prenom ) values (1,'belaouad','saad') ;

1 ligne créée.

SQL>
```

13. Donner à l'utilisateur student\_1 le droit de visualiser la table emp de l'utilisateur student\_2

```
SQL> grant select on student_2.emp to student_1;

Autorisation de privilèges (GRANT) acceptée.

SQL>
```

14. Dans le compte student\_1 afficher le contenu de table emp.

```
SQL> conn student_1 /student1123;
Connecté.
SQL> select * from student_2.emp ;
```

ID	NOM	PRENOM
1	belaouad	saad

```
SQL>
```

15. Retirer le droit de visualisation de la table emp à l'utilisateur student\_1. Essayer d'afficher emp de nouveau.

```
SQL> revoke select on student_2.emp from student_1;

Suppression de privilèges (REVOKE) acceptée.

SQL> _
```

```
SQL> revoke select on student_2.emp from student_1;

Suppression de privilèges (REVOKE) acceptée.

SQL> conn student_1 /student1123;
Connecté.
SQL> select * from student_2.emp;
select * from student_2.emp
*
ERREUR Ó la ligne 1 :
ORA-00942: Table ou vue inexistante

SQL> _
```

16. Modifier le mot de passe de l'utilisateur student\_1.

```
SQL> conn sys as sysdba
Entrez le mot de passe : 
Connecté.
SQL> alter user student_1 identified by student1234;

Utilisateur modifié.

SQL>
```

17. Verrouiller le compte de l'utilisateur student\_1. Essayer de vous connectez à l'utilisateur student\_1

```
SQL> alter user student_1 account lock ;

Utilisateur modifié.

SQL> conn student_1 /student1123;
ERROR:
ORA-28000: Compte verrouillé.

Avertissement : vous n'êtes plus connecté à ORACLE.
SQL>
```

 Le compte verrouillé

18. Connectez-vous avec le compte student\_3, laissez le connecté et essayer de le supprimer. Que se passe-t-il ?

```
SQL> conn student_3 /student3123
Connecté.
SQL> drop user student_3;
drop user student_3
*
ERREUR à la ligne 1 :
ORA-01031: privilèges insuffisants
```

19. Déconnecter l'utilisateur student\_3 et supprimer son compte.

```
SQL> alter session set "_ORACLE_SCRIPT"=true;


Session modifiée.

SQL> drop user student_3;

Utilisateur supprimé.

SQL>
```

20. Essayer de supprimer l'utilisateur student\_2 avec l'instruction DROP USER student\_2, que ce passe-t-il ?

 Il faut utiliser le cascade puisque il existe des tables créées par student\_2

```
SQL> drop user student_2;
drop user student_2
*
ERREUR à la ligne 1 :
ORA-01922: CASCADE à indiquer pour supprimer 'STUDENT_2'

SQL> drop user student_2 cascade ;

Utilisateur supprimé.

SQL>
```

# Privilèges System

## 21- Créer deux utilisateurs Pilote1 et Pilote2

- 1) Donner les droits à l'utilisateur Prof1l pour qu'il puisse se connecter à la base sous SQL\*Plus par un outil (la console par exemple), par un programme. Il peut créer des séquences.

```
SQL> grant create sequence , create session to pilote1;  
Autorisation de privilèges (GRANT) acceptée.  
SQL>
```

- 2) Pilote1 peut créer des tables dans son schéma et peut retransmettre ce privilège à un tiers

```
SQL> grant create table to pilote1 with admin option;  
Autorisation de privilèges (GRANT) acceptée.  
SQL> _
```

- 3) Pilote2 peut se connecter à la base, créer et détruire des tables dans tout schéma.

```
SQL> grant create session , create any table , drop any table to pilote1;  
Autorisation de privilèges (GRANT) acceptée.  
SQL>
```

- 4) Pilore1 et Pilote2 ne peuvent plus se connecter à la base. Ils conservent néanmoins leurs autres privilèges

```
SQL> revoke create session from pilote1,pilote2;
```

```
Suppression de privilèges (REVOKE) acceptée.
```

```
SQL>
```

```
SQL> conn pilote1 /pilote123
```

ERROR:

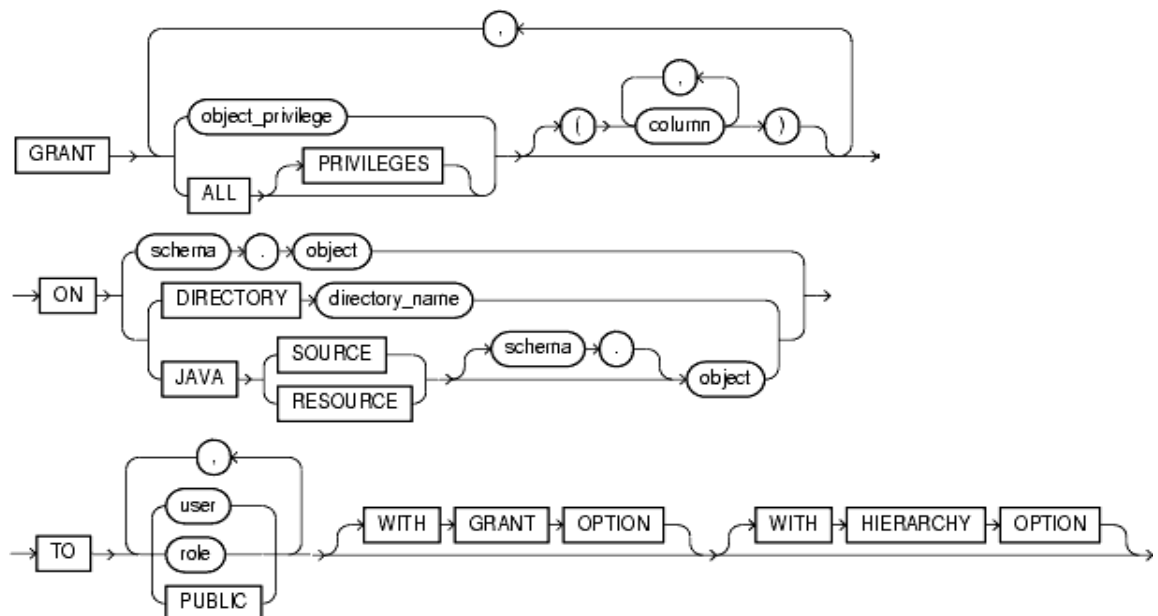
```
ORA-01045: l'utilisateur PILOTE1 n'a pas le privilège CREATE SESSION ;
connexion refusée
```

Avertissement : vous n'êtes plus connecté à ORACLE.

SQL&gt;

## Privilèges Objet

**Rappel :**



- **privilègeObjet** : description du privilège objet (ex : SELECT, DELETE, etc.).
- **colonne** précise la ou les colonnes sur lesquelles se porte le privilège INSERT, REFERENCES, ou UPDATE (exemple : UPDATE (typeAvion) pour n'autoriser que la modification de la colonne typeAvion).
- **ALL PRIVILEGES** donne tous les privilèges avec l'option GRANT OPTION) l'objet en question.
- **PUBLIC** : pour attribuer le(s) privilège(s) à tous les utilisateurs.
- **WITH GRANT OPTION** : permet de donner aux bénéficiaires le droit de retransmettre les privilèges reçus à une tierce personne (utilisateur(s) ou rôle(s)).
- **WITH HIERARCHY OPTION** : accorder le privilège d'objet spécifié sur tous les sous-objets, tels que les sous-vues créées.

**22-Affecter des privilèges de lecture de la table Pilote, de modification des colonnes nom et âge et référence à la clé primaire brevet à l'utilisateur Ahmed.**

```
--Table pilote
```

BREVET	NOM	AGE	ADRESSE
--------	-----	-----	---------

P1	Alaoui	30	Marrakech
P2	Benani	35	Rabat
P3	bensghir	35	Casablanca
P4	benali	41	Tanger

```
SQL> grant select on sys.pilote to ahmed ;
Autorisation de privilèges (GRANT) acceptée.
SQL> grant update(nom,prenom,brevet) on sys.pilote to ahmed ;
Autorisation de privilèges (GRANT) acceptée.
SQL> █
```

**23-** Ahmed ne peut plus modifier ni lire la table Pilote de TEST.

```
SQL> revoke select , update on sys.pilote from ahmed ;
Suppression de privilèges (REVOKE) acceptée.
SQL>
```