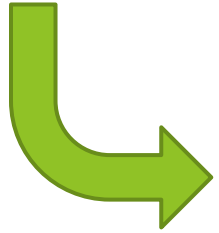


# LANGAGE DE CONTRÔLE DE DONNÉES (LCD)

# Langage de Contrôle de Données (LCD)

## Introduction

Un SGBD oracle est un système **multiutilisateur** complexe



- Il doit **protéger** les données de l'utilisateur
- Il doit permettre un **partage contrôlé** des données

Dans une BD, les droits des utilisateurs sont gérés avec la notion de **privilege**.

Un privilege est le droit :

- D'exécuter un ordre SQL (par exemple créer une table) : c'est la notion de **privilege système**.
- D'accéder à un objet d'un autre utilisateur (par exemple mettre à jour les données de la table CLIENT) : c'est la notion de **privilege objet**.

# Langage de Contrôle de Données (LCD)

## Introduction

- Plusieurs personnes peuvent travailler simultanément sur une base de données, toutefois ces personnes n'ont pas forcément les mêmes besoins : certaines peuvent par exemple nécessiter de modifier des données dans la table, tandis que les autres ne l'utiliseront que pour la consulter. Ainsi, il est possible de définir des permissions pour chaque personne en leur octroyant un mot de passe.
- Cette tâche incombe à l'administrateur de la base de données (en anglais DBA, DataBase Administrator). Il doit dans un premier temps définir les besoins de chacun, puis les appliquer à la base de donnée sous forme de permissions.

# Gestion des utilisateurs

## Identification

### Créer un utilisateur

- Mode d'identification de l'utilisateur

Un utilisateur peut être identifié en se connectant à la base par un nom et un mot de passe:



```
CONNECT user_name/pwd;
```

- Créer un utilisateur:



```
CREATE USER User_name IDENTIFIED BY pwd;
```

Exemple

```
CREATE USER user1 IDENTIFIED BY pwd_user1 ;
```

# Langage de Contrôle de Données (LCD)

## modifier un utilisateur

- Modifier le mot de passe:



```
ALTER USER User_name Identified by NW_pass_word;
```

Ou

```
ALTER USER User_name Identified by NW_pass_word REPLACE pass_word;
```

- Modification de status



```
ALTER USER User_name ACCOUNT LOCK; #Verrouillage
```

Ou



```
ALTER USER User_name ACCOUNT UNLOCK; #Activation
```

# Langage de Contrôle de Données (LCD)

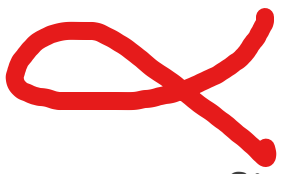
## supprimer un utilisateur

- Consulter les différents utilisateurs:



```
SELECT * FROM ALL_USERS;
```

- Supprimer un utilisateur:



```
DROP USER User_name CASCADE;
```

- Si l'utilisateur possède des objets, l'option CASCADE doit être présente pour forcer la suppression préalable des objets.
- Si l'utilisateur possède des objets et l'option CASCADE n'est pas présente :  
Erreur.
- Un utilisateur connecté ne peut pas être supprimé

# Langage de Contrôle de Données (LCD)

## GRANT et REVOKE

Le langage SQL permet de gérer les droits sur les objets de la base grâce à deux clauses :

- ▶ **GRANT** permet d'**accorder des droits** à un (parfois plusieurs sur certains SGBD) utilisateur.
- ▶ **REVOKE** permet de **retirer des droits** à un (ou plusieurs sur certains SGBD) utilisateur. Les permissions (appelées aussi droits ou privilèges) peuvent être définies pour chaque (un grand nombre) clause. D'autre part il est aussi possible de définir des rôles c'est-à-dire de permettre à d'autres utilisateurs d'accorder des permissions.

# Langage de Contrôle de Données (LCD)

## GRANT

**GRANT:** La commande GRANT permet d'attribuer des droits à des utilisateurs sur vos objets.



```
GRANT privilège (ou ALL) ON object_name TO user_name  
[ou PUBLIC] [WITH GRANT OPTION];
```

- Les privilèges sont : CONNECT, RESOURCE, DBA, SELECT, INSERT, UPDATE, DELETE, ALTER ou (tout cela et donc ALL).
- On peut attribuer plusieurs privilèges, mais il faut les séparer par une virgule.
- On peut préciser plusieurs colonnes, en autant qu'elles soient séparées par une virgule.
- WITH GRANT OPTION, permet à l'utilisateur à qui on vient d'attribuer des droits d'attribuer les mêmes droits à d'autres usagers sur le même objet.
- PUBLIC permet d'attribuer les droits à TOUS.



# Langage de Contrôle de Données (LCD)

## PRIVILÈGE et RÔLE

Un rôle représente un ou plusieurs privilèges. On crée des rôles lorsqu'un ensemble de privilèges vont être attribués à plusieurs usagers. Il existe des rôles déjà prédéfinis.

- ▶ Le rôle **CONNECT**. Ce rôle permet à un usager de se connecter à la base de données. Il n'a de sens que si d'autres privilèges lui sont accordés.
- ▶ Le rôle **RESOURCES** : ce rôle permet à un utilisateur de créer ses propres objets.
- ▶ Le rôle **DBA**, ce rôle permet d'attribuer à un usager des rôles d'administration.
- ▶ Syntaxe de création d'un ROLE



```
CREATE ROLE name_role;
```

- ▶ Exemple:

```
CREATE ROLE role1;
```

# Langage de Contrôle de Données (LCD)

## PRIVILÈGE et RÔLE

- Attribuer des privilèges à un ROLE



```
GRANT <privilège> ON name_object TO name_role;
```

Exemple: Attribuer les privilèges « select », « update », et « insert » au rôle « role1 » dans la table « etudiants ».

```
GRANT SELECT, UPDATE, INSERT ON etudiants TO role1;
```

- Attribuer un ROLE à un utilisateur

```
GRANT role_name TO user_name;
```

Exemple:

```
GRANT role1 TO Alex, Michelle;
```

- Détruire un ROLE

```
DROP role_name
```

# Langage de Contrôle de Données (LCD)

## REVOKE

**REVOKE:** permet de supprimer des privilèges.



```
REVOKE <privilège> (ou ALL) ON object_name FROM user_name  
[ou PUBLIC];
```

Exemple:

- Supprimer le droit d'insérer des données dans la table « etudiant » de l'utilisateur « Alex »



```
REVOKE insert ON etudiant FROM Alex;
```

- Supprimer tous les droits de tous les utilisateurs dans la table « employe »



```
REVOKE ALL ON employe TO PUBLIC;
```