



# CHAPITRE 3

## SQL-LMD

Mise à jour des données

# 1. Ajout de données

- L'instruction **INSERT INTO** permet d'insérer de nouveaux enregistrements dans une table.
- Il est possible d'écrire l'instruction **INSERT INTO** de deux manières :

- Spécifiez à la fois les noms de colonnes et les valeurs à insérer.

```
INSERT INTO table_name (column1, column2, column3, ...)
VALUES (value1, value2, value3, ...);
```

- Si vous ajoutez des valeurs pour toutes les colonnes de la table, vous n'avez pas besoin de spécifier les noms de colonnes dans la requête SQL. Cependant, assurez-vous que l'ordre des valeurs est dans le même ordre que les colonnes du tableau.

```
INSERT INTO table_name
VALUES (value1, value2, value3, ...)
```

# 1. Ajout de données

- **Exemple 1 :** Préciser toutes les colonnes

```
insert into Employe(Matricule, Nom, Prenom, DateNaissance, Ville, Salaire, Num_Service)
values
(7, 'NOM7', 'PRENOM7', '1/1/1990', 'TANGER', 5000, 'S1')
```

- **Exemple 2:** Préciser quelques colonnes seulement

```
insert into Employe(Matricule, Nom, Prenom)
values
(8, 'NOM8', 'PRENOM8')
```

- **Exemple 3:** ne pas préciser les colonnes

```
insert into Employe
values
(9, 'NOM9', 'PRENOM9', '1/1/1990', 'SAFI', 6000, 'S2')
```

# 1. Ajout de données

- Il est possible d'insérer plusieurs lignes simultanément.

- **Exemple :**

```
insert into Employe values  
(1, 'NOM1', 'PRENOM1', '2000/01/01', 'VILLE1', 1000, 1),  
(2, 'NOM2', 'PRENOM2', '2000/02/02', 'VILLE2', 2000, 2),  
(3, 'NOM2', 'PRENOM2', '2000/03/03', 'VILLE3', 3000, 3);
```

- L'instruction **INSERT INTO SELECT** copie les données d'une table et les insère dans une autre table.
- L'instruction **INSERT INTO SELECT** requiert que les types de données dans les tables source et cible correspondent.

- **Exemple:**

```
insert into employe2  
select matricule,nom,prenom,datenaissance,ville from employe;
```

## 2. Suppression de données

- L'instruction DELETE est utilisée pour supprimer des enregistrements existants dans une table.

- Syntaxe:

```
DELETE FROM table_name WHERE condition;
```

- Exemple :

```
Delete from Employe where Matricule=12
```

```
Delete from Employe where Salaire is Null
```

- Soyez prudent lorsque vous supprimez des enregistrements dans une table ! Notez la clause WHERE dans l'instruction DELETE. La clause WHERE spécifie quel(s) enregistrement(s) doivent être supprimés. Si vous omettez la clause WHERE, tous les enregistrements de la table seront supprimés !

- Exemple :

```
Delete from Employe
```

### 3. Modification de données

- L'instruction **UPDATE** permet de modifier les enregistrements existants dans une table.

- Syntaxe:

```
UPDATE table_name  
SET column1 = value1, column2 = value2, ...  
WHERE condition;
```

- Exemple :

```
Update Employe Set Ville='RABAT', Salaire=6000 where Matricule=1
```

```
update employe set salaire=salaire+ 100 where num_service=1;
```

- Soyez prudent lors de la mise à jour des enregistrements. Si vous omettez la clause WHERE, TOUS les enregistrements seront mis à jour !

- Exemple :

```
update employe set salaire=salaire+ 1000;
```





# FIN