

# Les opérations de mise à jour dans la base de données avec PHP

# Présentation

- Les opérations de mise à jour sont des opérations qui peuvent affecter une base de données. Ça peut être une **insertion** dans un table, une **modification** de données ou **suppression** d'enregistrements d'une table.
- Pour les trois opérations la méthode qu'on va utiliser c'est la méthode **exec()** au lieu de **query()** pour la sélection.

# Différence entre query() et exec()

- La différence entre ces deux méthodes est que :
  - **query()** retournera un **jeu de résultats** sous la forme d'un objet
  - **exec()** retournera uniquement **le nombre de lignes affectées**.
- En d'autres termes, vous utiliserez query() pour des requêtes de sélection (SELECT) et exec() pour des requêtes d'insertion (INSERT), de modification (UPDATE) ou de suppression (DELETE).

# Exécuter une commande d'insertion

- Pour exécuter une commande INSERT dans PHP, on utilise la syntaxe suivante:

```
$nb=$conn -> exec(" INSERT into MaTable values (.....) ");
```

Où

- \$conn: désigne l'objet connexion
- \$nb: c'est le nombre de lignes affectées, n'est pas obligatoire.

```
$nb=$conn -> exec(" INSERT into Clients values ('".$_POST['code']."',  
".$_POST['societe']."', ....) ");
```

# Quelques remarques

- **Remarque:**

1. Si nous avons un champ défini **auto incrémenté** dans une table, il faudra lui donner la valeur **null** si le nom de ce champ est mentionné dans la requête,
2. On peut se servir du nombre de lignes affectées pour afficher un message indiquant si l'opération de mise à jour a été bien effectuées ou non.

```
If ($nb>0){  
echo "opéarion réussite";  
}  
else{  
echo "Opération non effectuée !!"  
}
```

# Récupérer la valeur du dernier identifiant inséré

- On peut savoir quelle est la valeur du dernier identifiant inséré dans une table à l'aide de la méthode **lastInsertId()**
- **Syntaxe:**

```
$dernierId=$conn -> lastInsertId();
```

- **Remarque:** cette ligne doit être écrite après avoir exécuté la requête d'insertion.

# Exécuter une commande de modification

- Pour exécuter une commande UPDATE dans PHP, on utilise la syntaxe suivante:

```
$nb=$conn -> exec(" UPDATE MaTable set champ1=valeur ,  
champ2=valeur ,...");
```

- Exemple:

```
$nb=$conn -> exec(" UPDATE clients set societe='".$_POST['soc']."' where  
codeClient= '".$_POST['cd']."' ");
```

# Exécuter une commande de suppression

- Pour exécuter une commande DELETE dans PHP, on utilise la syntaxe suivante:

```
$nb=$conn -> exec(" DELETE from maTable where condition");
```

- Exemple:

```
$nb=$conn -> exec(" DELETE from clients where codeClient= '$_POST['cd']' ");
```



# La méthode `prepare()`

- C'est une méthode utilisée pour exécuter des requêtes dites préparées.
- Les requêtes préparées sont des requêtes (de mise à jour ou de sélection) dont le code SQL est donné, mais sans passer des valeurs directement dans la requête.
- Les avantages des requêtes préparées c'est qu'elles sont :
  1. Plus rapide dans l'exécution que les requêtes classiques
  2. Plus sécurisées, car les valeurs ne sont pas données directement dans la requête.

# Utilisation des requêtes préparés à l'aide des marqueur ?

- Exemple d'insertion:

```
$reqPrep=$conn → prepare (" insert into filieres values (?,?)");
```

- On passe les valeurs de prix et nom du produits dans un tableau comme suit:

```
$MonTableau=array ($_POST['x'], $_POST['y']);  
$reqPrep → execute( $MonTableau);
```

# Utilisation des requêtes préparés à l'aide des marqueur ?

```
$requete=$conn->prepare("select * from filiere where numfil=? ");  
//$T=[$a];  
$requete->execute([$a]);  
$ligne=$requete->fetch();
```

- Puis on exécute la requête avec les valeurs du tableau.
- **Remarque:** les valeurs passées dans le tableau doivent être dans le **même ordre** que celles de la requête.

# Utilisation des requêtes préparés à l'aide des paramètres nominatifs

- Syntaxe et exemple:
- On prépare la requête:

```
$reqPrep=$conn → prepare ("select * from produits where  
prix=:x and nomproduit=:y ");
```

- On passe les valeurs dans un tableau associatif comme suit:

```
$MonTableau=array (':x' => 13, ':y' => "mon produit");  
$reqPrep → execute( $MonTableau);  
$ligne=$requete->fetch();
```

- On exécute la requête en dernière étape.