

PHP et Bases de Données

Section 1. Se connecter à une base de donnée

Vous avez une table dans un fichier .sql , vous pouvez la transférer dans un fichier .sqlite en écrivant sur la console:
sqlite3 exemple.sqlite < exemple.sql

Dans un fichier .php on commence par établir la connexion avec la table en indiquant le chemin du fichier .sqlite, puis on crée un data-object pour la BD

```
<?php
$file = "../bd/departements.sqlite";

try{
$dbd = new PDO("sqlite:$file");
}
catch(PDOException $e){
exit("Non connecté à la BD : ".$e->getMessage());
}
```

Attention! Si vous faites une erreur de spelling pour le nom du fichier .sqlite, il va créer un fichier .sqlite vide

En suite on fait la requête sqlite avec la méthode query: (à l'intérieur du try) on écrit

```
$requeteSQLITE="SELECT * FROM departements";
$requete = $dbd->query($requeteSQLITE);
```

On peut maintenant afficher les données de la table sur la page avec un foreach sachant que les éléments de \$requete sont des arrays correspondants aux lignes du résultat de la requête.

```
1  <?php
2
3  $file = "../bd/departements.sqlite";
4  //attention si spelling incorrecte, il va créer un fichier vide avec ce nom
5
6  try{
7
8  $dbd = new PDO("sqlite:$file"); //création d'un data-object pour la BD
9  $requeteSQLITE= "SELECT * FROM departements";
10 $requete = $dbd->query($requeteSQLITE); //on lance la requête
11
12
13 //affichage du résultat de la requête sous forme de tableau
14 echo "<table border=1px>
15 <tr>
16 |   <th>No</th>
17 |   <th>Nom</th>
18 </tr>
19 ";
20
21 foreach($requete as $ligne){
22     echo "
23     <tr>
24         <td>$ligne[no]</td>
25         <td>$ligne[nom]</td>
26     </tr>
27     ";
28 }
29 echo "</table>";
30
31 }
32 catch(PDOException $e){
33     exit("Non connecté à la BD : ".$e->getMessage());
34 }
```

Vous pouvez aussi faire des “requêtes préparées” avec la méthode prepare puis lancer la requête avec execute.

```
1  <?php
2
3  $file = "./bd/produits.sqlite";
4
5  try{
6  $bd = new PDO("sqlite:$file");
7  $requete = "SELECT * FROM produits WHERE prix< ?";
8  $result = $bd->prepare($requete);
9  $result->execute([5]);
10
11  echo "<table border = 1px>";
12
13  foreach($result as $line){
14      echo "<tr>
15          <td>$line[nom]</td>
16          <td>$line[prix] euros</td>
17          </tr>";
18  }
19
20  echo "</table>";
21
22  }
23  catch(PDOException $e){
24      exit($e->getMessage());
25  }
26
27
28
```

Section 2. Créer une base de données en PHP

Dans le repertoire de travail on va créer un fichier .sqlite vide:

```
<?php
```

```
$file = "./bd/carnet.sqlite";
```

```
$bd = new PDO("sqlite:$file");
```

On peut créer la table comme suit:

```
$bd->exec("
DROP TABLE IF EXISTS carnet;
CREATE TABLE carnet (id INTEGER PRIMARY KEY, nom text, phone text)
");
```

Attention! DROP TABLE efface la table si elle existe déjà

On peut insérer des données sur la table comme suit:

```
$bd->exec("INSERT INTO carnet (nom, phone) VALUES ('Jean', '0612354777')");
```

Attention! À chaque fois qu'on relance la page, ces commandes sont re-exécutées. Donc si on ne veut pas rajouter deux fois les mêmes données il faut effacer ces lignes une fois que c'est fait.

On peut afficher les données de la table comme dans la section 1:

```
1  <?php
2
3  $file = "../bd/carnet.sqlite";
4
5  $bd = new PDO("sqlite:$file");
6
7  //ATTENTION! c'est executé à chaque fois que la page est chargée
8  $bd->exec("
9  DROP TABLE IF EXISTS carnet;
10 CREATE TABLE carnet (id INTEGER PRIMARY KEY, nom text, phone text)
11 ");
12
13 //ATTENTION! c'est executé à chaque fois que la page est chargée
14 $bd->exec("INSERT INTO carnet (nom, phone) VALUES ('Jean', '0612354777')");
15
16 $requete = $bd->query("SELECT nom, phone FROM carnet");
17
18 echo "<table border = 1px>
19     <tr>
20         <th>Nom</th>
21         <th>Phone</th>
22     </tr>
23 ";
24
25 foreach($requete as $row){
26     echo "
27     <tr>
28         <td>$row[nom]</td>
29         <td>$row[phone]</td>
30     </tr>
31     ";
32 }
33
34 echo "</table>";
```