

TP PWS N°5

Tables des matières

Introduction.....	2
1 - Page de connexion.....	3
2 - Modifier pages existantes.....	4
Conclusion.....	6

Introduction

Ce TP va porter sur la liaison entre le sql et php et sur comment utiliser l'interface utilisateur phpMyAdmin.

1 - Page de connexion

Une fois le script SQL remplie, on retrouve ceci:

```
DROP TABLE IF EXISTS Medias;
DROP TABLE IF EXISTS Types;

CREATE TABLE Types (
  idType VARCHAR(4),
  nomType VARCHAR(25),
  PRIMARY KEY (idType);
) Engine=InnoDB;

CREATE TABLE Medias (
  idMedia VARCHAR(3),
  idType VARCHAR(4) NOT NULL,
  nomMedia VARCHAR(50),
  dateParution mediumint(4),
  PRIMARY KEY (idMedia),
  FOREIGN KEY (idType) REFERENCES Types(idType)
) Engine=InnoDB;

INSERT INTO Types (idType, nomType) VALUES ('100', 'Romans');
INSERT INTO Types (idType, nomType) VALUES ('200', 'Essais');
INSERT INTO Types (idType, nomType) VALUES ('300', 'DVD');

INSERT INTO Medias (idMedia, idType, nomMedia, dateParution) VALUES ('115', '100', 'Crime et châtiment', 1866);
INSERT INTO Medias (idMedia, idType, nomMedia, dateParution) VALUES ('198', '100', 'Madame Bovary', 1857);
INSERT INTO Medias (idMedia, idType, nomMedia, dateParution) VALUES ('231', '200', 'Notre Patrie', 1905);
INSERT INTO Medias (idMedia, idType, nomMedia, dateParution) VALUES ('302', '300', 'Django Unchained', 2013);
INSERT INTO Medias (idMedia, idType, nomMedia, dateParution) VALUES ('357', '300', 'Forrest Gump', 1994);

COMMIT;
```

Nous voyons donc après avoir exécuté ce script sur phpMyAdmin nos tables apparaître:



Contenu de la table Medias:

				idMedia	idType	nomMedia	dateParution			
<input type="checkbox"/>		Modifier		Copier		Effacer	115	100	Crime et châtiment	1866
<input type="checkbox"/>		Modifier		Copier		Effacer	198	100	Madame Bovary	1857
<input type="checkbox"/>		Modifier		Copier		Effacer	231	200	Notre Patrie	1905
<input type="checkbox"/>		Modifier		Copier		Effacer	302	300	Django Unchained	2013
<input type="checkbox"/>		Modifier		Copier		Effacer	357	300	Forrest Gump	1994

2 - Modifier pages existantes

Fichier `connect.inc.php`:

```
<?php

try {
    $strConnection = 'mysql:host=localhost;dbname=base_nbriet';
    $arrExtraParam= array(PDO::MYSQL_ATTR_INIT_COMMAND => "SET NAMES utf8");
    $conn = new PDO($strConnection, 'nbriet', 'cabarouto31', $arrExtraParam);
} catch (PDOException $e) {
    echo 'Erreur';
    die();
}

?>
```

Fichier `consultDate.php`:

Partie 1, formulaire en HTML:

```
<form action='consultDate.php' method="POST">
    Date de parution inférieure à 1900 <input type="radio" name="group1" value="inf1900"><br>
    Date de parution entre 1900 et 1999 <input type="radio" name="group1" value="medium"><br>
    Date de parution postérieure à 2000 <input type="radio" name="group1" value="sup2000"><br>
    <input type="submit" value="Afficher" name="submit">
</form>
```

Partie2, préparation du tableau, en PHP:

```
if (isset($_POST["group1"])){
    echo "<br><br><br>";
    echo "<table border=1 style=\"margin : 0 auto\"> ";
    echo "<tr><th>idMedia</th><th>idType</th><th>nomMedia</th><th>dateParution</th></tr>";
```

Partie3, Requête et affichage.

```
if($_POST["group1"] == "inf1900"){
    $pdoStat = $conn->prepare("Select * from Medias where Medias.dateParution < 1900");
    $pdoStat->execute();
    while($nom = $pdoStat->fetch())
    {
        echo "<tr> <td>".$nom["idMedia"]."</td><td>".$nom["idType"]."</td><td>".$nom["nomMedia"]."</td><td>".$nom["dateParution"]."</td></tr>";
    }
}
```

Remarque: Ceci est l'extrait de code pour le bouton de médias datant d'avant 1900.

Ainsi pour chaque bouton nous préparons la requête avec un test sur *dateParution* différent et nous l'exécutons.

J'aurais put factoriser la préparation de la requête (pour la préparer une fois pour tout les boutons et pas une a chaque bouton) et donc exécuter en donnant des paramètres, comme j'ai fais pour exercice suivant.

Fichier `consultType.php`:

Partie 1, formulaire en HTML:

```
<form action='consultType.php' method="POST">
    Romans <input type="radio" name="group1" value="romans"><br>
    Essais <input type="radio" name="group1" value="essais"><br>
    DVD <input type="radio" name="group1" value="dvd"><br>
    <input type="submit" value="Rechercher" name="submit">
</form>
```

Partie 2, fonction pour récupérer de la table Type l'id du media dont le nom est donné en paramètre

```
// $pNomType doit etre compris dans {Romans,Essais,DVD}
function getIdType ($pNomType , $pConn){
    $pdoStat = $pConn->prepare("Select idType from Types where Types.nomType = :nomType");
    $pdoStat->execute(array('nomType' => $pNomType));
    while ($id = $pdoStat->fetch()){
        return($id['idType']); // je ne sais pas faire autrement que avec while et fetch
    }
    // cette boucle n'est pas utile ici car une seule valeur est récupérée
}
```

Partie 3, fonction pour afficher les medias récupéré de la table Medias dont l'id (récupéré avec la fonction `getIdType`)

```
function afficherMedia ($pIdMedia , $pConn){
    $pdoStat = $pConn->prepare("Select idMedia , idType , nomMedia , dateParution from Medias where Medias.idType = :pIdType");
    $pdoStat->execute(array('pIdType' => $pIdMedia));
    echo "<table border=1 style='margin:0 auto'> ";
    while ($row = $pdoStat->fetch()){
        echo "<tr> <td>".$row["idMedia"]."</td><td>".$row["idType"]."</td><td>".$row["nomMedia"]."</td><td>".$row["dateParution"]."</td></tr>";
    }
    echo "</table>";
}
```

Partie 4, partie maitresse du php qui appelle les fonction avec les bons paramètre en fonctions des boutons sélectionnés:

```
if (isset($_POST["group1"]) and isset($_POST["submit"])){
    if ($_POST["group1"] == "romans"){
        $id = getIdType('Romans' , $conn);
        afficherMedia ($id, $conn);
    }
    if ($_POST["group1"] == "essais"){
        $id = getIdType('Essais', $conn);
        afficherMedia ($id, $conn);
    }
    if ($_POST["group1"] == "dvd"){
        $id = getIdType('DVD', $conn);
        afficherMedia ($id, $conn);
    }
}
```

Conclusion

Ce TP nous à permis de mettre en pratique ce que nous avons vu en TD pour ce qui est de la liaison entre PHP et MYSQL c'est a dire sur la façon d'utiliser des objet de type PDO avec la préparation l'exécution avec ou sans argument et la récupération des données.