
COMPTE RENDU TP05 – DEVWEB

Christopher Marie-Angélique – Groupe 2B

Table des matières

1 – Introduction.....	2
2 – Structures des tables MySQL sous phpmyadmin.....	2
3 – Consultations des biens par prix	5
4 – Consultations des biens par types	7
5 – Conclusion	10

1 – Introduction

Ce TP5 de DevWeb à pour objectif de nous faire manipuler les requêtes préparées.

Pour ce faire, nous utiliserons le PDO (PHP Data Objects) qui est une extension PHP fournissant une interface pour accéder à une base de données, ainsi que phpmyadmin qui est une interface Web pour MySQL écrite en PHP.

2 – Structures des tables MySQL sous phpmyadmin

Phpmyadmin est un logiciel pour MySQL livré avec MAMP et XAMPP écrit en PHP qui nous permet de visualiser rapidement l'état de notre base de données et de la modifier sans devoir écrire de requêtes SQL.

Avant développé le code PHP qui nous permet d'afficher les données de notre base de données, nous devons naturellement la créer et insérer ces données. Pour ce faire, nous avons complété le script fournit « CreateBaseBienImmoAtrous.sql » que nous avons ensuite exécuter sur phpmyadmin :

Figure 1 : script de création

```
DROP TABLE IF EXISTS Biens;
DROP TABLE IF EXISTS Types;

CREATE TABLE Types (
  idType VARCHAR(3),
  nomType VARCHAR(25),
  CONSTRAINT PK_Types PRIMARY KEY(idType)
);

CREATE TABLE Biens (
  idBien VARCHAR(5),
  idType VARCHAR(3) NOT NULL,
  titreBien VARCHAR(30),
  detailBien VARCHAR(50),
  prixBien mediumint(5),
  CONSTRAINT PK_Biens_Types PRIMARY KEY (idBien),
  CONSTRAINT FK_Biens_Types FOREIGN KEY (idType)
  REFERENCES Types(idType)
);
```

Figure 2 : script d'insertion

```
INSERT INTO Types (idType, nomType) VALUES ('F1', 'Une pièce');
INSERT INTO Types (idType, nomType) VALUES ('F2', 'Deux pièces');
INSERT INTO Types (idType, nomType) VALUES ('F3', 'Trois pièces');
INSERT INTO Types (idType, nomType) VALUES ('F4', 'Quatre pièces');
INSERT INTO Types (idType, nomType) VALUES ('F5', 'Cinq pièces');
INSERT INTO Types (idType, nomType) VALUES ('F6', 'Six pièce');
INSERT INTO Types (idType, nomType) VALUES ('F7', 'Sept pièces');
INSERT INTO Types (idType, nomType) VALUES ('FG', 'Plus de sept pièces');

INSERT INTO Biens(idBien, titreBien, detailBien, prixBien, idType)
VALUES('B0001', 'Villa Rieumes', 'Jolie maison de plain-pied', 270000, 'F7');

INSERT INTO Biens(idBien, titreBien, detailBien, prixBien, idType)
VALUES('B0002', 'Villa Garidech', 'Villa de plain-pied, cuisine équipée', 320000, 'F5');

INSERT INTO Biens(idBien, titreBien, detailBien, prixBien, idType)
VALUES('B0003', 'Villa Plaisance', 'Belle maison de caractère', 250000, 'F5');

INSERT INTO Biens(idBien, titreBien, detailBien, prixBien, idType)
VALUES('B0004', 'Villa Beaumont', 'Ferme authentique', 175000, 'F4');

INSERT INTO Biens(idBien, titreBien, detailBien, prixBien, idType)
VALUES('B0005', 'Villa Auterive', 'Villa neuve avec terrain', 215000, 'F4');

INSERT INTO Biens(idBien, titreBien, detailBien, prixBien, idType)
VALUES('B0006', 'Villa St Rustice', 'Maison ancienne', 245000, 'F4');

INSERT INTO Biens(idBien, titreBien, detailBien, prixBien, idType)
VALUES('B0007', 'Villa L'Union', 'Charmante maison de village', 195000, 'F4');

INSERT INTO Biens(idBien, titreBien, detailBien, prixBien, idType)
VALUES('B0008', 'Villa L'éguevin', 'Maison avec beaux volumes', 250000, 'F5');

INSERT INTO Biens(idBien, titreBien, detailBien, prixBien, idType)
VALUES('B0009', 'Villa Bessieres', 'Agréable maison de plain pied', 275000, 'F5');

INSERT INTO Biens(idBien, titreBien, detailBien, prixBien, idType)
VALUES('B0010', 'Villa St-Lys', 'Belle maison d'architecte', 245000, 'F4')

COMMIT;
```

Finalement, sous phpmyadmin les résultats sont les suivant :

Figure 3 : les tables sous phpmyadmin

Table	Action	Lignes	Type	Interclassement	Taille	Perte
<input type="checkbox"/> Biens	★ Parcourir Structure Rechercher Insérer Vider Supprimer	10	InnoDB	utf8mb4_0900_ai_ci	32,0 kio	-
<input type="checkbox"/> Types	★ Parcourir Structure Rechercher Insérer Vider Supprimer	8	InnoDB	utf8mb4_0900_ai_ci	16,0 kio	-
2 tables	Somme	18	InnoDB	utf8mb4_0900_ai_ci	48,0 kio	0 o

Figure 4 : les colonnes de la table bien

<input type="checkbox"/>	1	idBien		varchar(5)	utf8mb4_0900_ai_ci	Non	Aucun(e)	Modifier	Supprimer	Plus
<input type="checkbox"/>	2	idType		varchar(3)	utf8mb4_0900_ai_ci	Non	Aucun(e)	Modifier	Supprimer	Plus
<input type="checkbox"/>	3	titreBien		varchar(30)	utf8mb4_0900_ai_ci	Oui	NULL	Modifier	Supprimer	Plus
<input type="checkbox"/>	4	détailBien		varchar(50)	utf8mb4_0900_ai_ci	Oui	NULL	Modifier	Supprimer	Plus
<input type="checkbox"/>	5	prixBien		mediumint		Oui	NULL	Modifier	Supprimer	Plus

Figure 5 : les valeurs de la table biens

<input type="checkbox"/>	Éditer	Copier	Supprimer	B0001	F7	Villa Rieumes	Jolie maison de plain-pied	270000
<input type="checkbox"/>	Éditer	Copier	Supprimer	B0002	F5	Villa Garidech	Villa de plain-pied, cuisine équipée	320000
<input type="checkbox"/>	Éditer	Copier	Supprimer	B0003	F5	Villa Plaisance	Belle maison de caractère	250000
<input type="checkbox"/>	Éditer	Copier	Supprimer	B0004	F4	Villa Beaumont	Ferme authentique	175000
<input type="checkbox"/>	Éditer	Copier	Supprimer	B0005	F4	Villa Auterive	Villa neuve avec terrain	215000
<input type="checkbox"/>	Éditer	Copier	Supprimer	B0006	F4	Villa St Rustice	Maison ancienne	245000
<input type="checkbox"/>	Éditer	Copier	Supprimer	B0007	F4	Villa L'Union	Charmante maison de village	195000
<input type="checkbox"/>	Éditer	Copier	Supprimer	B0008	F5	Villa Léguevin	Maison avec beaux volumes	250000
<input type="checkbox"/>	Éditer	Copier	Supprimer	B0009	F5	Villa Bessieres	Agréable maison de plain pied	275000
<input type="checkbox"/>	Éditer	Copier	Supprimer	B0010	F4	Villa St-Lys	Belle maison d'architecte	245000

Figure 6 : les colonnes de la table Types

<input type="checkbox"/>	1	idType		varchar(3)	utf8mb4_0900_ai_ci	Non	Aucun(e)	Modifier	Supprimer	Plus
<input type="checkbox"/>	2	nomType		varchar(25)	utf8mb4_0900_ai_ci	Oui	NULL	Modifier	Supprimer	Plus

Figure 7 : les valeurs de la table Types

<input type="checkbox"/>	Éditer	Copier	Supprimer	F1	Une pièce
<input type="checkbox"/>	Éditer	Copier	Supprimer	F2	Deux pièces
<input type="checkbox"/>	Éditer	Copier	Supprimer	F3	Trois pièces
<input type="checkbox"/>	Éditer	Copier	Supprimer	F4	Quatre pièces
<input type="checkbox"/>	Éditer	Copier	Supprimer	F5	Cinq pièces
<input type="checkbox"/>	Éditer	Copier	Supprimer	F6	Six pièce
<input type="checkbox"/>	Éditer	Copier	Supprimer	F7	Sept pièces
<input type="checkbox"/>	Éditer	Copier	Supprimer	FG	Plus de sept pièces

Pour permettre la connexion à notre base de données, nous avons créé un fichier php « connext.inc.php » qui créera un objet PDO qui permettra d'accéder à notre base de données et d'exécuter des requêtes préparées :

```

<?php
// Ce fichier connect.inc.php sera inclus dans
// chaque page PHP qui travaille avec la BD

try {
    $user='mysql2047';
    $pass='7Pd5n58GetPQ';

    $conn = new PDO('mysql:host=localhost:3306;dbname=mysql2047;charset=utf8',$user, $pass, array(PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION));
} catch (PDOException $e) {
    echo "Erreur: ".$e->getMessage()."<br>" ;
    die() ;
}

?>

```

3 – Consultations des biens par prix

Dans cette partie, il nous a été demandé d’afficher les biens selon leurs prix. L’utilisateur aura le choix entre sélectionner les biens qui sont soit :

- > à 200k €
- Entre 200k et 300k €
- < à 300k €

Pour ce faire, nous créons tout d’abord un formulaire avec des radioButton qui représenterons les choix que peut faire l’utilisateur :

Je suis la page consultation des biens par prix

Choisir une tranche de prix :

☒ < 200.000 €
 ☐ de 200.000 \$ à 300.000 €
 ☐ > 300.000 €

Afficher

```

<div style="text-align: center" class="col-md-12">
  <h2>Je suis la page consultation des biens par prix</h2> <br>
  <h2>Choisir une tranche de prix : <br></h2>

  <form method="post">
    <fieldset>
      <input type="radio" name="prix" value="200k" checked>
      <label for="radio1">< 200.000 €</label> <br>

      <input type="radio" name="prix" value="200k300k">
      <label for="radio2">de 200.000 $ à 300.000 €</label> <br>

      <input type="radio" name="prix" value="300k">
      <label for="radio3">> 300.000 €</label> <br>

      <input type="submit" name="afficher" value="Afficher">
    </fieldset>
  </form>

```

Pour afficher le résultat au clique du bouton « afficher », nous créons un script PHP vérifiant qu'un des radioButton a bien été sélectionner. Dans un switch nous afficherons des résultats différents selon que le bouton est soit :

- > à 200K €
- Entre 200k et 300k €
- < à 300k €

Ces vérifications sont réalisés grâce à une requête qui vérifie les valeurs de la table prixBien :

```

$stmt = $conn->prepare('SELECT * FROM Biens where prixBien < 200000');
$stmt->execute();

```

```

$stmt = $conn->prepare('SELECT * FROM `Biens` WHERE prixBien BETWEEN "200000" AND "300000"');
$stmt->execute();

```

```

$stmt = $conn->prepare('SELECT * FROM Biens where prixBien > 300000');
$stmt->execute();

```

Par la suite, nous stockons les résultats dans un tableau en parcour les résultats de notre requête :

```
foreach ($stmt as $row) {
    $tabFinale[] = $row;
}
```

Pour afficher proprement les biens, nous utilisons la propriété « table » en HTML que nous initialisons en mettant les noms des différentes tables de notre base de données :

```
$table = "<br><br><table border='2'>
    <tr>
        <td><strong>idBien</strong></td>
        <td><strong>titreBien</strong></td>
        <td><strong>prixBien</strong></td>
        <td><strong>detailBien</strong></td>
        <td><strong>idType</strong></td>
    </tr>
";
```

Ensuite dans une boucle foreach, nous concaténons les données présentes dans le tableau qui stocke les données pour finalement l'afficher :

```
foreach($tabFinale as $tab){
    $table .= '<tr>
        <td>'.$tab['idBien'].'</td>
        <td>'.$tab['titreBien'].'</td>
        <td>'.$tab['prixBien'].'</td>
        <td>'.$tab['detailBien'].'</td>
        <td>'.$tab['idType'].'</td>
    </tr>';
}
```

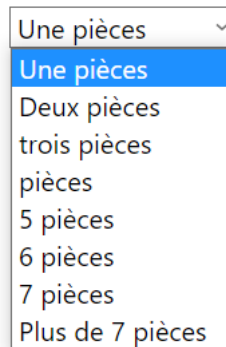
4 – Consultations des biens par types

Dans cette partie, il nous a été demandé d'afficher les biens selon leurs types (F1/F2/F3/F4/F5/F6/F7/FG)

Pour ce faire, nous créons tout d'abord un formulaire avec une balise « select » qui possèdera plusieurs options :

Je suis la page consultation des biens par type

Choisir un type :



```
<div style="text-align: center" class="col-md-12">
  <h2> Je suis la page consultation des biens par type</h2> <br>
  <h3>Choisir un type :</h3> <br>

  <form method="post">
    <select name="type">
      <option value="F1">Une pièces</option>
      <option value="F2">Deux pièces</option>
      <option value="F3">trois pièces</option>
      <option value="F4">pièces</option>
      <option value="F5">5 pièces</option>
      <option value="F6">6 pièces</option>
      <option value="F7">7 pièces</option>
      <option value="FG">Plus de 7 pièces</option>
    </select>
    <br><br>
    <input type="submit" name='afficher' value='Afficher'>
  </form>
```

Pour afficher le résultat au clique du bouton « afficher », nous créons un script PHP vérifiant qu'une option a bien été sélectionner. Si c'est cas, nous préparons et exécutons une requête qui récupère tous les biens :


```
if(!empty($_POST['type'])){

    $stmt = $conn->prepare('SELECT * FROM Biens');
    $stmt->execute();
```

Pour afficher proprement les Biens selon leur types, nous utilisons la propriété HTML : « table » que nous initions en mettant les noms des différentes tables de notre base de données :

```
$table = "<br><table border='2'>
    <tr>
        <td><strong>idBien</strong></td>
        <td><strong>titreBien</strong></td>
        <td><strong>prixBien</strong></td>
        <td><strong>idType</strong></td>
    </tr>";
```

Par la suite dans une boucle foreach, nous vérifions que les biens soit bien du type sélectionner par l'utilisateur. Si c'est le cas nous concaténons ces données dans le tableau que nous avons créé pour finalement l'afficher :

```
foreach($stmt as $row){
    if($row['idType'] == $_POST['type']){
        $table .= '<tr>
            <td>'.$row['idBien'].'</td>
            <td>'.$row['titreBien'].'</td>
            <td>'.$row['prixBien'].'</td>
            <td>'.$row['idType'].'</td>
        </tr>';
    }
}

$table .= "</table>";

echo ($table);
}
```

5 – Conclusion

Nous avons vu dans ce TP5 de DevWeb comment utiliser le logiciel phpmyadmin pour créer et visualiser base de données. Par la suite, nous avons créé un PDO qui nous permet d'accéder à notre base de données avec PHP. Grâce à cela nous avons afficher les éléments de nos différentes tables, soit les Biens selon leur prix ou biens leur types. Pour ce faire, nous avons créé des requêtes préparées.