

# Compte-rendu du TP de MySQL

## Objectif du TP

L'objectif de ce cours était de nous familiariser avec le langage PHP qui, associé à du code HTML, est couramment utilisé pour la création de pages web dynamiques. De manière pratique, nous avons appliqué ceci à la création d'un formulaire permettant de concevoir des cartes de visite en ligne. Les données qui doivent apparaître sur la carte de visite (nom, prénom, adresse, numéro de téléphone,...) sont saisies par l'intermédiaire d'une page web dynamique puis récupérées à l'aide du système de gestion de bases de données MySQL.

## Introduction

D'une manière générale, il est bien plus simple de stocker et de gérer des données dans une base de données que de les manipuler directement à l'aide de structures comme les fichiers et les tableaux. La recherche d'informations est notamment bien plus facile de cette façon.

Mes fichiers se trouvent à l'adresse suivante:  
[http://agserver3.free.fr/CDV\\_Jessica/](http://agserver3.free.fr/CDV_Jessica/)

## 1. Création du formulaire en ligne

Nous avons d'abord utilisé le codage HTML pour créer le formulaire de saisie. Quelque soit ce que l'on veut coder en HTML, nous retrouvons toujours la structure principale suivante:

```
<html>
<head>
<title> Création d'une carte de visite </title>
</head>
<body>
</body>
</html>
```

Dans le BODY, nous insérons une balise FORM qui délimitera le formulaire. Dans cette balise, nous trouverons METHOD qui permet de choisir la manière de laquelle les réponses seront transmises (Post, le plus utilisé en php ou Get) et ACTION qui permet de définir l'url qui recevra les réponses.

Nous insérons ensuite une balise FIELDSET qui permet de regrouper l'ensemble des champs du formulaire. Parmi ces champs, certains sont des zones de texte à remplir, d'autres sont des listes déroulantes, d'autres encore sont des boutons de type radio (cases à cocher).

### a) Zones de texte

Dans le formulaire de carte de visite, l'utilisateur doit pouvoir entrer son nom, son prénom, son adresse complète (rue, code postal, ville). Pour ce faire, nous avons inséré une balise INPUT dans laquelle nous avons spécifié le type (type="text"), le nom (name="nom"), la valeur de la variable (value="...") qui correspond à ce que l'utilisateur a

saisi pour ce champ. Dans l'exemple ci-dessous, value="<? echo @\$\_POST['nom'] ?>" signifie que l'information fournie lors de l'envoi du précédent formulaire et stockée dans la variable 'nom' sera affichée dans la case Nom.

```
<input type="text" name="nom" value="<? echo @$_POST['nom'] ?>"/>
```

Nous avons utilisé la création de zones de texte pour les informations suivantes: Nom, Prénom, Rue, Code Postal, Ville, Téléphone et Portable.

### **b) Zones de texte libre**

Il est possible que l'utilisateur veuille entrer des informations spécifiques qui devront apparaître sur sa carte de visite et qui peuvent être plus ou moins longues. Pour cela, nous avons inséré une balise TEXTAREA dans laquelle nous avons spécifié le nom (name="description") et les dimensions (cols="30" rows="5").

```
<textarea name="description" rows="5" cols="30"> <?= $_POST['description']?>
</textarea>
```

### **c) Listes déroulantes**

Pour créer cette liste déroulante, nous avons inséré une balise SELECT dans laquelle nous avons spécifié le nom de la variable (ici "pays"). A l'intérieur de cette balise, nous avons listé les différentes valeurs que peut prendre la variable "pays" à l'aide d'une balise OPTION.

```
<select name="pays">
<option value="Allemagne">Allemagne</option>
<option value="Angleterre">Angleterre</option>
<option value="Chine">Chine</option>
<option value="Espagne">Espagne</option>
<option value="France" selected="selected">France</option>
<option value="Japon">Japon</option>
<option value="USA">USA</option>
</select>
```

### **d) Cases à cocher**

Certaines informations qui devront apparaître sur la carte de visite ne nécessitent que 2 ou 3 choix possibles. Dans ce cas, il est inapproprié de créer une zone de texte et plus pratique pour l'utilisateur de créer 2 ou 3 cases à cocher. C'est le cas de l'exemple ci-dessus.

Pour créer ces boutons de choix, nous avons inséré une balise INPUT dans laquelle nous avons spécifié le type (type="radio"), le nom (name="titre") et la valeur (value="..."). Ce type de boutons est très intéressant lorsque l'on veut que l'utilisateur ne puisse choisir qu'une seule possibilité (cocher une seule case).

```
<input type="radio" name="titre" value="madame" <? If ($_POST['titre']=='madame')
echo ""checked"" ?>Mme
<input type="radio" name="titre" value="monsieur" <? If ($_POST['titre']=='monsieur')
echo ""checked"" ?>Mr
```

```
<input type="radio" name="titre" value="mademoiselle" <? If  
($_POST['titre']=='mademoiselle') echo "checked" ?>Melle
```

#### **e) Bouton "Envoyer"**

Afin que l'utilisateur puisse valider les informations qu'il a saisies et que ces informations soient enregistrées, nous avons créé un bouton "Envoyer". Ce bouton est réalisé à l'aide d'une balise INPUT dans laquelle nous avons spécifié le type (type="submit"), la valeur de la variable (value="Envoyer") et le nom de la variable (name="faire").

```
<input type="submit" value="Envoyer" name="faire" />
```

## **2. Enregistrement des informations dans une base de données**

Nous allons maintenant devoir enregistrer et conserver les informations ayant été saisies par l'utilisateur afin de pouvoir les exploiter par la suite. Pour ce faire, nous allons créer une base de données grâce à MySQL. Cette base de données peut être créée sur le site <http://phpmyadmin.free.fr>

Tout d'abord, nous avons créé une table qui rassemble l'ensemble des informations envoyées à chaque validation du formulaire. Les variables nom, prenom, rue, codepostal, ville, telephone,... apparaissent dans cette table et sont définies par le type de données qu'elles permettent d'enregistrer.

J'ai rencontré quelques problèmes quant à la création de la base de données et je ne pourrai donc pas présenter la table.