

Conception de Sites Web Dynamiques : TD 12 (Upload image JPG) Upload d'images

1 Objectif

Dans ce TD, nous allons voir comment permettre aux utilisateurs de votre site de joindre une photo JPG lorsqu'il remplissent un formulaire. Nous prenons un exemple simple de fournir un nom de paysage, et joindre une photo. Les noms des paysages seront enregistré dans une table **paysage** (avec les champs **id**, **nom**, et **dateajout**), et les images fournies par les utilisateurs seront stockées dans un dossier **images_upload**, et les noms des fichiers seront construits à partir de l'identifiant *id* de la table MySQL, afin de pouvoir les associer au bon enregistrement de la table MySQL **paysage**. Par exemple, quand l'identifiant *id* est 1, le fichier s'appellera **paysage-1.jpg**, dans le dossier **images_upload**

2 TD pour téléverser un fichier (upload)

Aujourd'hui, nous travaillons dans le dossier **td12**. Copiez votre fichier **connexion_base.php** dans ce dossier **td12**. Soit vous créez ce dossier directement sur votre espace web **espaces/www** au CREMI, soit vous testez d'abord avec votre logiciel WAMPserver/MAMP.

Exercice 12.1 *phpinfo*

1. Créer un fichier **phpinfo.php** qui contient uniquement la ligne suivant (même pas les en-têtes HTML).

```
<?php phpinfo(); ?>
```

2. Ouvrir ce fichier avec un navigateur.
3. Regarder surtout les variables **upload_max_filesize**, et **DOCUMENT_ROOT**.

Exercice 12.2 *Création d'une table dans la base de données*

Dans cet exercice, on créera une table **paysage** qui permet de définir un nom pour le paysage, et la date et l'heure de l'ajout. *Remarque : Bien sûr, dans un plus grand projet, vous pourriez ajouter des autres informations, comme une description détaillée, le pays où se trouve le paysage (p.ex. via un clé étrangère vers une table **pays**), les coordonnées GPS exact où se trouve le paysage, etc. Mais ce n'est pas le sujet de cet exercice.*

1. Créez la structure de cette table comme suit. N'oubliez pas de déclarer l'identifiant **id** en clé primaire et **A_I** (auto increment), comme d'habitude.

Table **paysage**

id	nom	dateajout
INT	TEXT	DATETIME

Exercice 12.3 *Formulaire pour un nouveau paysage*

Dans cet exercice, nous créons un formulaire pour ajouter un paysage.

1. Avec votre éditeur de texte, créez une nouvelle page **paysage-formulaire.php**. Cette page contiendra toutes les balises d'une page HTML5. Dans le corps de cette page (après **<body>**), créez un titre **h2** avec le texte "Ajoutez un nouveau paysage", ainsi qu'un formulaire qui renvoie vers la page **paysage-enregistrer.php** avec la méthode **post** comme suit :

```

<h2>Ajoutez un nouveau paysage</h2>

<form action="paysage-enregistrer.php" enctype="multipart/form-data" method="post">

<input type="text" name="nom" /><br />

    <input type="hidden" name="MAX_FILE_SIZE" value="3000000" />
    <input type="file" name="fichier" /><br />
    <input type="submit" />

</form>

```

Notez :

Pour pouvoir téléverser un fichier, il faut travailler avec la méthode POST, et d'avoir l'attribut suivant dans votre formulaire :

```
enctype="multipart/form-data"
```

Exercice 12.4 *Enregistrer le paysage dans la base de données*

Dans cet exercice, nous allons enregistrer le nouveau paysage dans la table **paysage**.

1. Créer un fichier `paysage-enregistrer.php` avec toutes les balises d'une page HTML5.
2. N'oubliez pas d'inclure le fichier `connexion_base.php` en tout début du fichier :

```
<?php require_once("connexion_base.php"); ?>
```

3. Pour récupérer le nom du paysage depuis le formulaire, et pour insérer les données dans votre table `categorie`, vous pouvez faire comme d'habitude. Pour simplifier, je vous donne la solution :

```

<?php
if (!empty($_POST['nom']))
{
    $nom = $_POST['nom'];

    // exécuter une requete MySQL de type INSERT
    $requete="INSERT INTO paysage (nom, dateajout) VALUES (?,NOW())";
    $reponse=$pdo->prepare($requete);
    $reponse->execute(array($nom));

    echo "<p>Votre paysage a été ajouté.</p>";

    // code du prochain exercice ici

}
?>

```

4. Testez votre formulaire en ajoutant un nouveau paysage, pour l'instant sans fichier image.

Exercice 12.5 *Enregistrer le fichier avec le bon nom de fichier*

Dans cet exercice, nous allons enregistrer l'image du nouveau paysage dans le dossier `images_upload`, et donner le bon nom du fichier pour pouvoir l'associer au bon enregistrement de la table `paysage`.

1. Créer un dossier `images_upload` dans votre dossier de travail `td12`.
2. Après l'insertion dans la base de données, il faut récupérer l'identifiant `id` du nouvel enregistrement afin de pouvoir créer le bon nom de fichier. Vous pouvez faire ceci en appelant la fonction suivante :

```
$dernier_id = $pdo->lastInsertId();
```

3. Insérer le script PHP suivant : Essayer de comprendre ce code PHP qui va rajouter le fichier dans votre dossier `images_upload`, et qui lui donne un nom qu'on peut facilement associer avec l'identifiant dans la base de données.

```
print_r($_FILES['fichier']); // Sortir cette ligne quand tout va bien
print_r($_FILES); // Sortir cette ligne tout va bien

if(!empty($_FILES['fichier']['tmp_name']))
{
    $size = getimagesize($_FILES['fichier']['tmp_name']);
    print_r($size);
    echo "Filetype : ".$_size['mime'];

    if ($size['mime'] == "image/jpeg")
    {
        $uploaddir = $_SERVER['DOCUMENT_ROOT']."/cswd/td12/images_upload/";
        $uploadfile = "paysage-".$_dernier_id.".jpg";

        if (move_uploaded_file($_FILES['fichier']['tmp_name'], $uploaddir.$uploadfile))
        {
            echo "<p>Votre image du paysage a bien été ajouté : ".$uploadfile."</p>";
        }
        else
        {
            echo "<p>Probleme sur le serveur : ".$uploaddir."</p>";
        }
    }
    else
    {
        echo "<p>Pas le bon type de fichier : ".$_size['mime']."</p>";
    }
}
else
{
    echo "<p>Pas de fichier spécifié.</p>";
}
```

4. Testez votre formulaire en mettant un nom de paysage, et en spécifiant un fichier JPG de votre choix.
5. Vérifier que le fichier est bien enregistré dans votre dossier `image_upload`, et que le nom du fichier est associé au bon enregistrement de la table `paysage`. Attention ! Si vous travaillez dans un autre dossier que `cswd/td12`, il faut adapter la ligne correspondante dans le code ci-dessus !

Exercice 12.6 *CREMI*

1. Transférer votre dossier td12 au CREMI (avec Filezilla ou via le canal de stockage).
2. Transférer votre table **paysage** de la base de données au CREMI via les interfaces web phpmyadmin. Ou alors, créez la table **paysage** dans votre base de données du CREMI, via phpmyadmin.
3. Testez l'ajout de paysage sur votre site hébergé au CREMI.

Exercice 12.7 *Afficher les paysage*

1. Créez une page `paysages.php` avec les balises HTML5 habituelles, et qui affichent tous les noms des paysages dans une liste `ul` avec des éléments de liste `li`.
2. Pour chaque élément de liste, affichez également l'image du paysage concernée.

Notez :

Lors du téléversement d'une image par un utilisateur, il est souvent intéressant de changer la définition de l'image en terme de nombre de pixels et/ou des créer des miniatures des images, tout en gardant le rapport largeur/hauteur (aspect ratio). Si vous voulez imposer un certain rapport largeur/hauteur, il faut aussi découper l'image pour éviter "d'étirer" l'image. La bibliothèque "gd" de PHP, qui est souvent préinstallé, propose des fonctions très intéressantes pour ce faire, par exemple `imagecopyresized` (<https://www.php.net/manual/fr/function.imagecopyresized.php>) et `imagecopyresampled` (<https://www.php.net/manual/fr/function.imagecopyresampled.php>).