



Voyons maintenant comment créer des formulaires, et surtout comment bien les utiliser. Les formulaires vont permettre à vos visiteurs de soumettre des informations, que ce soit un nom, un prénom, un chiffre, etc...

Prenons le code suivant :

exemple1

```
<html>
<head>
<title>Ma page de test</title>
</head>
<body>
<form action = "traitement.php" method="post">
Votre nom : <input type = "text" name = "nom"><br />
Votre fonction : <input type = "text" name = "fonction"><br />
<input type = "submit" value = "Envoyer">
</form>
</body>
</html>
```

Puis, lorsque l'utilisateur cliquera sur le bouton "Envoyer", les données du formulaire seront envoyées sur la page traitement.php.

Et dans la page traitement.php, nous allons récupérer une variable de type tableau (\$_POST : car notre formulaire a comme method la valeur post).

En clair, dans la page traitement.php, on aura une variable \$_POST['nom'] qui contiendra la chaîne de caractères qu'aura saisi le visiteur dans le champ "Votre nom : " (on a la variable \$_POST['nom'], car dans l'attribut name de notre formulaire pour le champ concernant le nom).

De même, on aura une variable \$_POST['fonction'] qui contiendra la chaîne de caractères qu'aura saisi le visiteur dans le champ "Votre fonction : " (encore une fois, on a la variable \$_POST['fonction'] car l'attribut name du champ prend la valeur fonction).

Prenons ensuite le code suivant pour la page traitement.php :

exemple2

```
<html>
<head>
<title>Ma page de traitement</title>
</head>
<body>
<?
// on teste la déclaration de nos variables
if (isset($_POST['nom']) && isset($_POST['fonction'])) {
    // on affiche nos résultats
    echo 'Votre nom est ' . $_POST['nom'] . ' et votre fonction est ' . $_POST['fonction'];
}
?>
</body>
</html>
```

En supposant que l'on écrive "LA GLOBULE" dans le champ "Votre nom" et "Webmaster" dans le champ "Votre fonction", on verra alors s'afficher à l'écran :

```
Votre nom est LA GLOBULE et votre fonction est Webmaster
```

PS : dans le cas où le formulaire utilise une méthode get, nous utilisons la variable tableau \$_GET.

Voyons maintenant le cas des formulaires munis d'un champ de type file (formulaire permettant le téléchargement de fichiers sur votre site).

Imaginons que l'on ai le formulaire suivant :

exemple3

```
<html>
<head>
<title>Ma page de test</title>
</head>
<body>
<form action = "traitement.php" method="post" enctype="multipart/form-data">
Votre fichier : <input type = "file" name = "mon_fichier"><br />
<input type = "hidden" name="MAX_FILE_SIZE" value="20000">
<input type = "submit" value = "Envoyer">
</form>
</body>
</html>
```

Pour récupérer votre fichier, vous avez à votre disposition le tableau \$_FILES qui aura plusieurs entrées :

- \$_FILES['mon_fichier']['tmp_name'] : le nom temporaire du fichier sur le serveur
- \$_FILES['mon_fichier']['name'] : le nom original du fichier sur la machine cliente
- \$_FILES['mon_fichier']['type'] : le type MIME du fichier
- \$_FILES['mon_fichier']['size'] : la taille du fichier

Naturellement, vous pourrez utiliser ces valeurs pour tester votre fichier.

Si il correspond à vos attentes, vous pourrez finaliser votre téléchargement à l'aides des fonctions copy ou move_uploaded_file (afin de copier le fichier téléchargé sur le disque dur du serveur).

Auteur : LA GLOBULE

Dernière révision du cours : 2007-06-18 15:11:57