

# FORMULAIRES ET PHP

(méthodes GET et POST)

## Introduction :

L'écriture du code d'un formulaire s'effectue avec le langage html, ces formulaires permettent à un utilisateur de transmettre des données.

Pour récupérer et traiter les données issues d'un formulaire, on utilisera dans ce cours le langage PHP.

## Remarque

Il faut relire attentivement le guide d'installation et d'utilisation du logiciel easyPhp devserver 17 (l'utilisation du serveur apache a été déjà abordé dans le chapitre précédent (initiation au php))

## I) Les formulaires:

### 11) L'élément form:

Les balises html `<form> ....</form>` permettent de mettre en place le bloc "formulaire":

```
1 <form method="post" action="fichier.php">
2
3 ICI LES ELEMENTS DU FORMULAIRE
4
5 </form>
6
```

Nous avons dans le code ci-dessus deux attributs qui sont *action* et *method*:

- L'attribut *action* a pour valeur le nom du fichier vers lequel les données à traiter seront envoyées.
- L'attribut *method* qui a pour valeur "post", il nous indique que la méthode utilisée pour effectuer cette requête http est la méthode post.

## Exemple 1 : un exemple de formulaire

Le fichier "Exemple1\_Whats\_your\_name.html" ci-dessous contient un formulaire permettant de saisir votre nom.

Quand l'envoi de cette donnée sera effectué vers le fichier Exemple1\_WhatsYourName.php, le serveur devra exécuter ce programme php pour répondre à la requête du client.

- Le fichier "Exemple1\_Whats\_your\_name.html"

```
1  <!DOCTYPE html>
2  <html lang="fr">
3
4  <head>
5      <meta charset="utf-8">
6      <title> formulaire </title>
7      <style>
8      form{
9          margin: 20px;
10         border: 6px ridge #aaa;
11         border-radius: 20px;
12         display: inline-block;
13         padding: 20px;
14     }
15     </style>
16 </head>
```

```
20 <form method="post" action="traitement.php">
21     <p>
22         <label for="votreNom">Entrez votre nom:</label>
23         <input type="text" name="nom" id="votreNom" />
24     </p>
25
26     <p>
27         <input type="submit" value="Cliquez pour valider" />
28     </p>
29 </form>
30
31 </body>
32 </html>
```

- Le fichier "Exemple1\_WhatsYourName.php"

```
1  <!DOCTYPE html>
2  <html lang="fr">
3
4
5  <head>
6      <meta charset="utf-8">
7      <title> traitement du formulaire </title>
8  </head>
9
10 <body>
11
12
13 <p> Bonjour
14 <?php
15     $reponse = $_POST['nom'];
16     echo $reponse;
17 ?>
18 </p>
19 </body>
20 </html>
```

Remarque: la ligne `$reponse=$_POST['nom']` permet d'attribuer à la variable `$reponse` la chaîne de caractères qui a été saisie par l'utilisateur dans le formulaire ( la balise `<input>` ayant l'attribut `name="nom"`). La variable `$_POST` est appelé variable super globale.

### TRAVAIL A FAIRE

Déposer ces deux fichiers dans le répertoire `eds-www` de voter serveur apache et tester le fonctionnement.

A retenir:

On crée un formulaire avec le code suivant:

```
<form method="post" action="cible.php">
  <p>
    on inserera ici les éléments de notre formulaire
  </p>
</form>
```

Ce formulaire enverra des informations à la page cible.php.

La balise input est la balise essentielle des formulaires car elle permet de créer un bon nombre d'éléments interactifs, la syntaxe de cette balise est la suivante:

```
<input type="text" name="nom" id="votreNom"/>
```

On peut l'utiliser avec d'autre type

## II) Les différents éléments de formulaire:

### 21) zone de texte : (lire la documentation sur MDM)

<https://developer.mozilla.org/fr/docs/Web/HTML/Element/Input/text>

### 22) Zone de saisie d'un nombre:

<https://developer.mozilla.org/fr/docs/Web/HTML/Element/Input/number>

### 23) Zone de texte étendue:

<https://developer.mozilla.org/fr/docs/Web/HTML/Element/Textarea>

### 24) Cases à cocher:

<https://developer.mozilla.org/en-US/docs/Web/HTML/Element/Input/checkbox>

### 25) Liste déroulante:

<https://developer.mozilla.org/fr/docs/Web/HTML/Element/Select>

### 26) Les boutons d'options:

<https://developer.mozilla.org/fr/docs/Web/HTML/Element/Input/radio>

### **III)Exercices:**

#### **31)Exercice 1 :**

Après avoir consulter attentivement la documentation proposée au paragraphe 25, contruire un élément form en utilisant la balise <select> permettant de changer la couleur de fond, la couleur du texte et la couleur du cadre en proposant un choix un choix de couleurs (ces modifications impacteront le texte du fichier intitulé II\_31\_Exercice1.php donc il faut consulter ce fichier).

Le fichier html à compléter est intitulé III\_31\_Exercice1\_A\_completer.html

Réponse : fichier cible intitulé III\_31\_Exercice1\_reponse.html dans le dossier Formulaires\_php\_Version\_eleve.

#### **32)Exercice2 :**

**Ecrire** un fichier intitulé «III\_32\_Exercice22\_GET .html » contenant un élément form.

Les éléments de ce formulaire doivent permettre d'envoyer des données personnelles (nom, prénom et age) vers un fichier php intitulé « traitement\_GET.php » **à créer**, la méthode http utilisée sera évidemment la méthode GET.

Correction : fichiers III\_32\_Exercice2\_GET .html et traitement\_GET.php dans le dossier Formulaires\_php (dans l'ENT)

#### **33)Exercice 3 :** mêmes questions avec la méthode http POST

Quelle est la différences entre les deux méthodes ?

(pour trouver la réponse, observer attentivement la barre d'adresse de votre navigateur pour les deux méthodes)

Correction : fichiers III\_33\_Exercice3\_POST.html et Traitement33.POST.php

### 34) Exercice 4 :

La page intitulée "III\_34\_Exercice4.html" du site à réaliser doit contenir un formulaire permettant d'identifier l'utilisateur pour accéder à une page intitulée acces34.php fichier cible de la page html.

Le fichier html doit permettre d'obtenir la présentation ci-dessous:

### Méthode GET

accès à une page php

identifiant:

Mot de passe:

(on ne se soucie pas de l'alignement de différents blocs)

### Ecrire le fichier III\_34\_Exercice4.html

Vérifier votre travail, avec votre navigateur exécuter votre fichier intitulée "III\_34\_Exercice4.html" pour contrôler l'affichage.

Ensuite charger dans le répertoire eds-www de votre serveur apache, le fichier intitulée "III\_34\_Exercice4.html" ainsi que le fichier cible acces34.php puis tester le fichier intitulée "III\_34\_Exercice4.html" qui doit vous permettre d'accéder à la page php avec l'identifiant admin et le mot de passe 1234.

La méthode d'envoi des information au serveur est-elle bien adaptée ?

(pour répondre observer attentivement la barre d'adresse de votre navigateur une fois que vous avez cliqué sur envoyer)

### Conclusion:

De façon générale il faut donc éviter d'utiliser la méthode GET pour transmettre les données issues d'un formulaire vers un serveur.

Il est important de bien comprendre que la méthode POST n'offre pas non plus une sécurité absolue car les informations transmises ne sont pas cryptées.

Seule l'utilisation du protocole sécurisé https garantit un transfert des données entre le client et le serveur.



