



Institut Supérieur des Technologies de l'Informations et de la Communication de Borj Cédria

Unité enseignement	Technologies et programmation web, Techniques Multimédia	Code U.E	520080
Elément d'enseignement	Programmation Web Dynamique	Code E.E	520101
Niveau	License Fondamentale en Sciences de l'Informatique	Classe	LFSI -3

CHAPITRE 4: FORMULAIRES HTML ET PHP

ELABORE PAR : AZZABI MOHAMED SEIFEDDINE



Année Universitaire 2017-2018

TABLE DES MATIERES

1	Introduction	3
2	Exemple pratique	3
2.1	Déclarer un formulaire	3
2.2	Afficher des données issues d'un formulaire (action.php).....	3
2.3	Résultat.....	3
3	A retenir	3
3.1	Dans la page HTML :	3
3.2	Dans la page PHP :	4
3.3	Exemple	4
4	La méthode POST	4
5	La méthode GET	5

1 INTRODUCTION

L'un des points forts de PHP est sa capacité à gérer les formulaires. Le concept de base qui est important à comprendre est que tous les champs d'un formulaire seront automatiquement disponibles dans le script PHP d'action. Lisez le chapitre du manuel concernant les [variables depuis des sources externes à PHP](#) pour plus d'informations et d'exemples sur la façon d'utiliser les formulaires. Voici un exemple de formulaire HTML :

2 EXEMPLE PRATIQUE

2.1 DECLARER UN FORMULAIRE

```
<form action="action.php" method="post">
  <p>Votre nom : <input type="text" name="nom" /></p>
  <p>Votre âge : <input type="text" name="age" /></p>
  <p><input type="submit" value="OK"></p>
</form>
```

Lorsque le visiteur remplit le formulaire, et clique sur le bouton OK, le fichier *action.php* est appelé. Dans ce fichier, vous pouvez écrire le script suivant :

2.2 AFFICHER DES DONNEES ISSUES D'UN FORMULAIRE (ACTION.PHP)

```
Bonjour, <?php echo $_POST['nom']; ?>.
Tu as <?php echo $_POST['age']; ?> ans.
```

2.3 RESULTAT

```
Bonjour Jean.
Tu as 29 ans.
```

3 A RETENIR

Pour pouvoir parser des informations depuis un formulaire HTML vers une page PHP il faut respecter les règles suivantes :

3.1 DANS LA PAGE HTML :


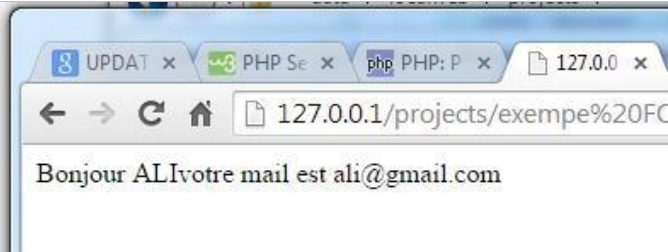
- Utiliser la balise **<form>** et renseigner les attributs suivants :
 - **action="fichier1.php"**
 - → C'est le nom du fichier qui contient le traitement PHP
 - **method="GET"** ou **method="POST"**
 - → GET pour envoyer les informations plus rapidement
 - → POST pour envoyer les informations plus sécurisé
- Utiliser la balise **<input type="submit">**

- → C'est en cliquant sur ce bouton que le contenu du formulaire est envoyé à la page fichier1.php

3.2 DANS LA PAGE PHP :

- Si vous avez utilisé la méthode **GET** lors de l'envoi du formulaire, les informations seront disponibles dans un tableau associatif nommé **\$_GET**
- Si vous avez utilisé la méthode **POST** lors de l'envoi du formulaire, les informations seront disponibles dans un tableau associatif nommé **\$_POST**

3.3 EXEMPLE

Page HTML fichier2.html	Page fichier1.php
<pre><form name="f1" method="get" action="fichier1.php"> Nom: <input id="nom" name="nom" type="text"> E-MAIL: <input id="mail" name="mail" type="email"> <input type="submit" value="Valider"> </form></pre>	<pre><?php \$n=\$_GET["nom"]; \$m=\$_GET["mail"]; echo "Bonjour ".\$n."votre mail est ".\$m; ?></pre>
 <p>Exemple Formulaire</p> <p>127.0.0.1:8020/PWD/exempe FORM/fichier2.html</p> <p>Nom: SALAH</p> <p>E-MAIL: salah@gmail.com</p> <p>Valider</p>	 <p>UPDAT x PHP Se x php PHP: P x 127.0.0 x</p> <p>127.0.0.1/projects/exempe%20FC</p> <p>Bonjour ALIvotre mail est ali@gmail.com</p>

4 LA METHODE POST

Un tableau associatif des valeurs passées au script courant via le protocole HTTP et la méthode POST lors de l'utilisation de la chaîne *application/x-www-form-urlencoded* ou *multipart/form-data* comme en-tête HTTP Content-Type dans la requête.

```
<?php
echo 'Bonjour ' . htmlspecialchars($_POST["name"]) . '!';
?>
```

En supposant que l'utilisateur a POSTÉ *name=Yannick*, l'exemple ci-dessus va afficher quelque chose de similaire à :

Bonjour Yannick !

5 LA METHODE GET

Un tableau associatif des valeurs passées au script courant via les paramètres d'URL.

```
<?php
echo 'Bonjour ' . htmlspecialchars($_GET["name"]) . '!';
?>
```

En assumant que l'utilisateur a entré *http://example.com/?name=Yannick* , l'exemple ci-dessus va afficher quelque chose de similaire à :

```
Bonjour Yannick!
```