

# **Gestion des formulaires en php**

# Formulaire HTML

Un formulaire permet de retourner des informations saisies par un utilisateur vers une application serveur. La création d'un formulaire nécessite la connaissance de quelques balises HTML indispensables :

- ◆ La balise form: `<form> </form>`
- ◆ Champ de saisie de text en ligne :  
`<input type = "text" name ="nom_du_champ" value="chaîne">`
- ◆ Boutons d'envoi et d'effacement :  
`<input type=" submit " value = "Envoyer">`  
`<input type = "reset" name ="efface" value = "Effacer">`
- ◆ Case à cocher et bouton radio :  
`<input type = "checkbox" name ="case1" value="valeur_case">`  
`<input type = "radio" name ="radio1" value ="valeur_radio">`
- ◆ Liste de sélection avec options à choix unique :  
`<select name ="select" size="1">`  
`<option value = "un"> choix </option>`  
`<option value ="deux"> choix2 </option>`  
`</select>`

# Méthodes d'envoi Get et Post

## Attributs de la balise <form>

- **action** : page qui traite les informations introduites dans le formulaire
- **method**: ("get" ou "post")

### => La méthode GET

#### Inconvénients :

- rendre visibles les données dans la barre d'adresse du navigateur  
-> envoi dans l'url des paires key/value : ?toto=val&titi=val2&...
- la longueur totale est limitée à 255 caractères, ce qui rend impossible la transmission d'un volume de données important

=> La méthode POST assure une confidentialité efficace des données

# Récupération des paramètres en PHP

Les tableaux associatifs **\$\_GET** et **\$\_POST** contiennent toutes les variables envoyées par un formulaire. L'index représente ici l'attribut « name » de chaque champ.

Lorsqu'on soumet un formulaire avec PHP via la méthode **POST**, celui-ci crée des variables **\$\_POST["X"]** avec 'X' correspondant aux noms des différents champs. Ceci est valable pour tout formulaire.

## Exemple:

**//Lecture d'une variable appartenant au tableau \$\_GET**

```
<?php  
echo $_GET["prenom"];  
?>
```

**//Lecture d'une variable appartenant au tableau \$\_POST**

```
<?php  
echo $_POST["prenom"];  
?>
```

# Récupération des paramètres en PHP

Exemple:

**Le code HTML du formulaire**

```
<html>
<body>
<form action="affiche.php" method="post">
Nom: <input type="text" name="nom"><br>
Prénom: <input type="text" name="prenom"><br>
<input type="submit" >
</form>
</body>
</html>
```

**Le code PHP de la page affiche.php**

```
<html>
<body>
Bienvenue
<?php echo $_POST["nom"]. " ". $_POST["prenom"]; ?>
</body>
</html>
```

# Récupération des paramètres en PHP

Pour tout champ de formulaire, lorsque le visiteur ne touche à rien, les valeurs par défaut seront conservées et utilisées de la même manière par PHP (qui n'a aucun moyen de savoir si il s'agit d'une valeur par défaut ou d'une valeur remplie).

⇒ **il faudra toujours vérifier les informations pour éviter les failles de sécurité**

Exemple1: Tester si les champs sont vides

**Le code HTML du formulaire de la page form.php**

```
<html>
<head>
</head>
<body>
  <form method="post" action="verif.php">
    Votre nom : <input type="text" name="nom"> <br>
    Votre prénom : <input type="text" name="prenom"> <br>
    <input type="submit" value="Envoyer">
  </form>
</body>
</html>
```

# Récupération des paramètres en PHP

## Exemple1: Tester si les champs sont vides

Le code PHP de la page `verif.php`

`<?php`

```
if (empty($_POST['nom']))
    {exit("ERREUR : le champ nom doit être rempli.");}
if(empty($_POST['prenom']))
    {exit("ERREUR : le champ prenom doit être rempli.");}
if( !empty($_POST['prenom']) && !empty($_POST['nom']) )
{
    // Tout est OK
    echo "tous les champs sont remplis";
    echo $_POST['nom'];
    echo $_POST['prenom'];
}

?>
```

# Récupération des paramètres en PHP

## Exemple1: Tester si les champs sont vides (version2)

### Le code PHP de la page `verif.php`

```
<?php
```

```
    $champ_vide=0;
    if (empty($_POST['nom']))    $champ_vide++;
    if (empty($_POST['prenom'])) $champ_vide++;
    if ($champ_vide==2)
        {exit("ERREUR : tous les champs ne sont pas remplis.");}
    elseif(empty($_POST['nom']))
        {exit("ERREUR : le champ nom doit être rempli.");}
    elseif(empty($_POST['prenom']))
        {exit("ERREUR : le champ prenom doit être rempli.");}
```

```
else {
    // Tout est OK
    echo "tous les champs sont remplis";
    echo $_POST['nom'];
    echo $_POST['prenom'];
}
```

```
?>
```



# Récupération des paramètres en PHP

## Exemple2: Tester la validité d'une adresse email

### Le code HTML du formulaire de la page form.php

```
<html>
<body>
<form action="verif.php" method="post">
  <p>
    <label for="email">eMail : </label>
    <input type="text" name="email" />
  </p>
  <p><input name="submit" type="submit" value="Envoyer"/></p>
</form>
</body>
</html>
```

# Récupération des paramètres en PHP

## Exemple2: Tester la validité d'une adresse email

### Le code PHP de la page `verif.php`

```
<php
//tester la validité du mail
if (isset($_POST['email']))
{
    $_POST['email'] = htmlspecialchars($_POST['email']);
    if (preg_match("#^[a-z0-9._-]+@[a-z0-9._-]{2,}\.[a-z]{2,4}$#i", $_POST['email']))
    { $email = $_POST['email']; }
    else
    { $email = "Adresse eMail invalide"; }
}
else
{
    $email = "";
}

echo "$email";

?>
```

# Récupération des paramètres en PHP

## **Exercice d'application:**

Ecrire un script PHP permettant de vérifier si les champs d'un formulaire sont vides ou non (Nom, Prénom, Age). Si c'est le cas un message d'avertissement sera affiché, sinon si tous les champs sont remplis alors on doit vérifier les conditions suivantes:

- **Si  $\text{age} \leq 30$  alors on affiche le message suivant « vous êtes adolescent »**
- **Si  $\text{age} > 30$  alors on affiche le message suivant « vous êtes adulte »**

# Récupération des paramètres en PHP

## Exemple3:

Le code HTML du formulaire de la page form.php

```
<form id="test" action="verif.php" method="post">
<p>Votre sexe : <label for="sexe_M">Masculin</label>
<input type="radio" name="sexe" value="M" />
<label for="sexe_F">Féminin</label>
<input type="radio" name="sexe" value="F" /> </p>
<p>Les acteurs que vous appréciez :<br />
<input type="checkbox" id="richard_gere" name="richard_gere" />
<label for="richard_gere">Richard Gere</label>
<input type="checkbox" name="bruce_willis" />
<label for="bruce_willis">Bruce Willis</label>
<input type="checkbox" name="harrison_ford" />
<label for="harrison_ford">Harrison Ford</label>
<input type="checkbox" name="richard_dean_anderson" />
<label for="richard_dean_anderson">Richard Dean Anderson</label>
<input type="checkbox" name="jean_reno" />
<label for="jean_reno">Jean Reno</label> <p>
<p><label for="inscrire">M'inscrire à la newsletter ?</label>
<input type="checkbox" id="inscrire" name="inscrire" /></p>
<p><input type="submit" name="submit" value="Valider" /></p>
</form>
```

# Récupération des paramètres en PHP

## Exemple3:

Le code PHP de la page `verif.php`

```
<?php
$sexe = $_POST['sexe'];
echo '<p>Votre sexe : '.$sexe.'</p>';
echo '<hr />';
echo '<p> Vos acteurs préférés :</p>';
echo '<ul>';
if (isset($_POST['richard_gere'])) {echo '<li>Richard Gere</li>';}
if (isset($_POST['bruce_willis'])) {echo '<li>Bruce Willis</li>';}
if (isset($_POST['harrison_ford'])) {echo '<li>Harrison Ford</li>';}
if (isset($_POST['richard_dean_anderson'])) {echo '<li>Richard Dean Anderson</li>';}
if (isset($_POST['jean_reno'])) {echo '<li>Jean Reno</li>';}
echo '</ul>';
if (isset($_POST['inscrire'])) // la case inscription a été cochée
{ echo '<p>Votre adresse email a été ajoutée à la liste !</p>'; }
?>
```

# Récupération des paramètres en PHP

## Exemple4:

### Le code HTML / PHP du index.php

```
<html>
<body>
<form name="monform" method="post">
choix 1 : <input type="checkbox" name="choix1" value="1"><br />
choix 2 : <input type="checkbox" name="choix2" value="2"><br />
choix 3 : <input type="checkbox" name="choix3" value="3"><br />
<input type="submit" name="go" value="go">
</form>
```

# Récupération des paramètres en PHP

## Exemple4:

Le code HTML / PHP du index.php

```
<?php
if(isset($_POST['go']))
{
    //on déclare une variable
    $choix = "";
    //on teste
    if(!empty($_POST['choix1'])) $choix .= $_POST['choix1'];
    if(!empty($_POST['choix2'])) $choix .= $_POST['choix2'];
    if(!empty($_POST['choix3'])) $choix .= $_POST['choix3'];
    echo $choix;
}
?>
</body>
</html>
```

# Récupération des paramètres en PHP

## Exemple5:

### Le code HTML / PHP du index.php

```
<html>
<body>
<form name="monform" method="post">
choix 1 : <input type="checkbox" name="choix[]" value="1"><br />
choix 2 : <input type="checkbox" name="choix[]" value="2"><br />
choix 3 : <input type="checkbox" name="choix[]" value="3"><br />
<input type="submit" name="go" value="go">
</form>
```



# Récupération des paramètres en PHP

## Exemple5:

Le code HTML / PHP du index.php

```
<?php
if(isset($_POST['go']) && !empty($_POST['choix']))
{
    //on déclare une variable
    $choix = '';
    //on boucle
    for ($i=0;$i<count($_POST['choix']);$i++)
    {
        //on concatène
        $choix .= $_POST['choix'][$i];
    }
    echo $choix;
}
else { echo 'Sélectionner un choix!'; }
?>
</body>
</html>
```