

Introduction à PHP

Formulaires HTML et PHP, interactions avec le client

`monnerat@u-pec.fr`

12 avril 2019

IUT de Fontainebleau

Formlaire et traitement

Passage des paramètres depuis un formulaire

Transmission de paramètres

Encodage des données

Validation des données

Éléments HTML de formulaire

Texte

Cases à cocher

Listes

Upload de fichiers

Formlaire et traitement

Formulaire

Permet l'envoi de données entre le navigateur (client) et le serveur (l'application) par http(s).

Pourquoi faire ?

- Créer un espace sécurisé par authentification.
- Uploader des fichiers.
- Administrer le contenu d'un site.
- Payer ses impôts
- Gérer un panier d'articles.
- Etc.

En HTML, `<form>` et `</form>` déclare le début et la fin d'un formulaire.

- L'attribut `action` indique l'url destinatrice des données saisies après soumission.

Balise form

```
<form action="TraiterFormulaire.php">
  <!--
  le formulaire
  -->
  <input type="submit" value="Envoyer">
</form>
```

- La soumission des données (l'envoi) est déclenchée grâce à un élément `input` ou `button` de type `submit` (par un click), ou l'appuie sur la touche "Entrée".

Passage des paramètres depuis un formulaire

Transmission de paramètres

Encodage des données

Validation des données

Lorsqu'un formulaire est envoyé à un script php du serveur web, toutes ses données se retrouvent dans des superglobales. Il s'agit de tableaux associatifs accessibles dans tout le script. On utilisera :

`$_GET []` ~> contient toutes les données envoyées via l'URL (en GET)

`$_POST []` ~> contient les données envoyées via un formulaire en POST

`$_REQUEST []` ~> est la fusion des deux premiers.

Lorsqu'un formulaire est envoyé à un script php du serveur web, toutes ses données se retrouvent dans des superglobales. Il s'agit de tableaux associatifs accessibles dans tout le script. On utilisera :

`$_GET []` ~> contient toutes les données envoyées via l'URL (en GET)

`$_POST []` ~> contient les données envoyées via un formulaire en POST

`$_REQUEST []` ~> est la fusion des deux premiers.

Récupération des données

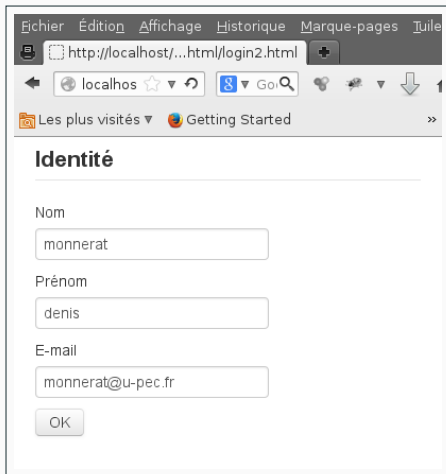
C'est la valeur de l'attribut **name** des éléments html du formulaire qui sert de clés dans les tableaux précédents.

Méthode POST

La méthode post permet entre autres de transmettre des "valeurs" saisies par l'utilisateur à un script (php), dans la **corps** de la requête http depuis un formulaire html.

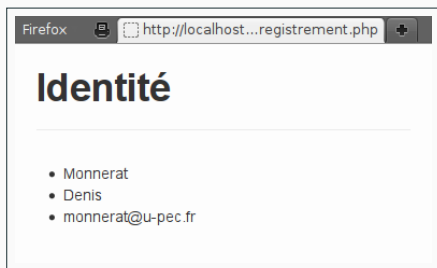
```
<form method="post" action="enregistrement.php">
  <fieldset>
    <legend><b>Identite</b></legend>
    <label for="e11">Nom</label> :
    <input id="e11" type="text" name="nom" />
    <label for="e12">Prenom</label> :
    <input id="e12" type="text" name="prenom" />
    <label for="e13">Email</label> :
    <input id="e13" type="email" name="email" />
  </fieldset>
  <input type="submit" name="accepter" value="OK">
</form>
```

L'exemple affiche un formulaire simple dans le navigateur : trois champs de saisie de texte.



Lorsque l'utilisateur valide et envoie les données au serveur, les variables du formulaire sont passées au script php destination au moyen du tableau `_POST`.

```
<?php
// on recupere les valeurs
// du formulaire
$nom=$_POST['nom'];
$prenom=$_POST['prenom'];
$email=$_POST['email'];
// on les traite
echo "<li>$nom</li>";
echo "<li>$prenom</li>";
echo "<li>$email</li>";
?>
```



Remarque : la fonction `extract` permet de créer des variables à partir des clés d'un tableau associatif. chaque case du tableau génère une variable du nom de sa clé.

```
<?php
// on recupere les valeurs du formulaire
extract($_POST);
// on les traite
echo "<li>$nom</li>";
echo "<li>$prenom</li>";
echo "<li>$email</li>";
?>
```

Méthode GET

L'attribut `method` vaut `get` (valeur par défaut)

```
<form method="get" action="enregistrement.php">
  <fieldset>
    <legend><b>Identite</b></legend>
    <label for="e11">Nom</label> :
    <input id="e11" type="text" name="nom" />
    <!-- etc -->
  </fieldset>
  <input type="submit" name="accepter" value="OK">
</form>
```

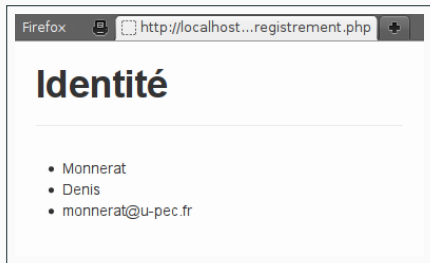
Les données sont passées directement via l'URL (elles sont encodées).

Méthode GET

hp?nom=Monnerat&prenom=Denis&email=monnerat%40u-pec.fr&accepter=OK

Les variables se récupèrent à l'aide du tableau `_GET`

```
<?php
// on recupere les valeurs
// du formulaire
$nom=$_GET['nom'];
$prenom=$_GET['prenom'];
$email=$_GET['email'];
// on les traite
echo "<li>$nom</li>";
echo "<li>$prenom</li>";
echo "<li>$email</li>";
?>
```



Passage des paramètres depuis un formulaire

Transmission de paramètres

Encodage des données

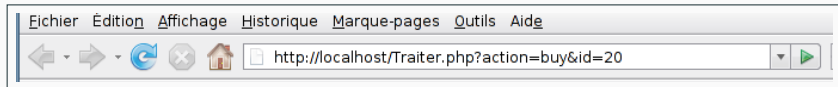
Validation des données

Passage de paramètres à un script

On peut passer des arguments à un script directement dans l'URL, comme le fait l'envoi des données avec la méthode GET.

```
$id=3;  
echo "<a href=\"Traiter.php?action=buy&id=$id\">Acheter</a>";
```

ce qui donnera l'URL :



La syntaxe pour passer des arguments à une page est la suivante

```
mon_url_ami?var1=valeur1&var2=valeur2 ...
```

Les variables transmises au script sont disponibles dans le tableau `_GET`. 14/38

Exemple

```
if (isset($_GET['rubrique']))
    $rubrique = $_GET['rubrique'];
else
    $rubrique = "";

switch($rubrique){
case "wim" :
    include('wim.php');
    break;
case "apl" :
    include('apl.php');
    break;
case "maths" :
    include('mathematique.php');
    break;
default :
    include ('accueil.php');
}
```

Exemple avec Google search

The screenshot shows a Google search for "iut de fontainebleau". The search results include several links to university pages, such as "IUT Sénart-Fontainebleau - Bienvenue à l'IUT" and "IUT de Sénart/Fontainebleau: Département Informatique". A knowledge panel for "IUT Sénart-Fontainebleau" is displayed on the right, featuring the UPEC logo, a map, and contact information: "Adresse : Route forestière Hurtault, 77300 Fontainebleau" and "Téléphone : 01 60 74 68 00". The panel also includes buttons for "Site Web" and "Itinéraire", and a section for "Avis" with "1 avis de Google". Below the knowledge panel, there are "Recherches associées" for other IUTs like "IUT d'Avignon" and "IUT de Paris-Villetaneuse".

Browser: Firefox
URL: https://www.google.fr/?gl=fr&cr=fr&rlz=1C1MDV8F2D6ex8wFVlK_IBw6gws_rd=ssl#q=iut-de-fontainebleau

Search: iut de fontainebleau

Results: Environ 206 000 résultats (0,48 secondes)

IUT Sénart-Fontainebleau - Bienvenue à l'IUT
www.iutsl.u-pec.fr/ | Bienvenue à l'IUT de Sénart Fontainebleau - découvrez nos DUT et nos licences professionnelles !

Nos formations
Toute l'offre de formation de l'IUT Sénart Fontainebleau : DUT ...

Contacts et organigramme
Contacts et organigramme. Version pdf; Imprimer la page ...

GEA
Gestion des entreprises et des administrations - Sénart ...

Autres résultats sur u-pec.fr »

Frédéric GERVAIS | Maître de conférences, Département ...
fredericgervais.com/ | Il y a 1 jour - Maître de conférences, Département Informatique, IUT Sénart Fontainebleau (IUEPC)

I.U.T. Fontainebleau - Ville de Fontainebleau
www.fontainebleau.fr/7-u-t-de-fontainebleau-deux- | L'IUT Sénart Fontainebleau, composante de l'UPEC est installé sur 2 sites, l'un en plein cœur de la forêt de Fontainebleau, l'autre à proximité du centre de la ...

IUT de Sénart/Fontainebleau: Département Informatique
www.iut-bleau.fr/ | IUT de Sénart Fontainebleau. Bienvenue sur le site interne du département informatique. Retenue 2015 - Portraits d'anciens étudiants - Fiche de ...

Images correspondant à iut de fontainebleau
Signaler des images inappropriées

IUT Sénart-Fontainebleau
Institut universitaire de technologie
Site Web | Itinéraire
Adresse : Route forestière Hurtault, 77300 Fontainebleau
Téléphone : 01 60 74 68 00
Suggérer une modification
Avis: 1 avis de Google
Donner un avis | Ajouter une photo
Envoyer sur votre téléphone | Envoyer

Recherches associées
Voir d'autres éléments (plus de 15)
IUT d'Avignon
Institut universitaire de technologie
IUT de Paris-Villetaneuse
Université Paris-Est de Technologie
IUT de Créteil-Villetaneuse
Institut universitaire de technologie
I.U.T. Paris Descartes
Institut universitaire de technologie

URL: https://www.google.fr/?gl=fr&cr=fr&rlz=1C1MDV8F2D6ex8wFVlK_IBw6gws_rd=ssl#q=iut-de-fontainebleau<[3/4] Top

Exemple avec Google search

rd=cr&ei=rmMDV8f2D6ex8wfVlK_IBw&gws_rd=ssl#q=iut+de+fontainebleau

ADE Expert [W3](#) The W3C Markup Val...

au



Images Actualités Vidéos Plus ▼ Outils de recherche

ats (0,48 secondes)

ainebleau - Bienvenue à l'IUT

Génart Fontainebleau - découvrez nos DUT et nos licences

ette page de nombreuses fois. Date de la dernière visite : 06/03/16



Passage des paramètres depuis un formulaire

Transmission de paramètres

Encodage des données

Validation des données

Encodage

L'attribut `enctype` de la balise `form` contrôle l'encodage des données. Par défaut, les données issues d'un formulaire sont de "type" `application/x-www-form-urlencoded`. Il s'agit d'une seule chaîne de caractères de la forme

```
var1=valeur1&var2=valeur2
```

qui est url encodée :

- Les caractères non alphanumériques sont remplacés par `%xx`, (`xx` est le code ascii du caractère en hexadécimal)
- Les caractères `; / ? : @ & = + $,` sont réservés :
 - `?` : début de `QUERY_STRING`.
 - `&` : séparateur de champs.
 - `=` : séparation entre le champ et sa valeur.
- Les espaces sont remplacés par des `+`.

Il existe d'autres encodages (notamment `multipart/form-data`).

URL	Statut	Domaine	Poids	Remot
POST enregistrer	200 OK	localhost	329 B	::1]:80
En-têtes Post Réponse HTML				
Paramètres		application/x-www-form-urlencoded		
accepter	OK			
email	monnerat@u-pec.fr			
nom	monnerat			
prenom	denis			
Source				
<code>nom=monnerat&prenom=denis&email=monnerat@u-pec.fr&accepter=OK</code>				

Attention

Seules les données au format `application/x-www-form-urlencoded` sont disponibles dans `_GET` ou `_POST`.

Quelle méthode utiliser ?

Dans les spécifications du protocole HTTP(s), rfc 2616

<http://www.ietf.org/rfc/rfc2616.txt> ↗,

- GET est sûre et idempotente : la requête ne modifie pas la ressource. Une répétition laisse le serveur dans le "même état".
- POST n'est pas idempotente. La requête change l'état du serveur. Ajout d'informations par exemple.

Pour résumer, GET doit être utilisée uniquement pour récupérer une ressource, POST pour en ajouter.

Évidemment, il s'agit de recommandations ...

1. Génération du premier formulaire.

1. Génération du premier formulaire.
2. Envoie des données.

1. Génération du premier formulaire.
2. Envoie des données.
3. Reception des données.

1. Génération du premier formulaire.
2. Envoie des données.
3. Reception des données.
4. Vérification de la validité des champs.

1. Génération du premier formulaire.
2. Envoie des données.
3. Reception des données.
4. Vérification de la validité des champs.
5. Si OK, on traite les données (écriture bd, etc.)

1. Génération du premier formulaire.
2. Envoie des données.
3. Reception des données.
4. Vérification de la validité des champs.
5. Si OK, on traite les données (écriture bd, etc.)
6. Sinon, on régénère le formulaire, en y insérant les données valides déjà saisies par l'utilisateur. (attribut `value`)

1. Génération du premier formulaire.
2. Envoie des données.
3. Reception des données.
4. Vérification de la validité des champs.
5. Si OK, on traite les données (écriture bd, etc.)
6. Sinon, on régénère le formulaire, en y insérant les données valides déjà saisies par l'utilisateur. (attribut `value`)

Règle d'or

Il ne faut jamais faire confiance aux données issues d'un formulaire. On reviendra de manière plus précise sur les bonnes pratiques en termes de sécurité.

Passage des paramètres depuis un formulaire

Transmission de paramètres

Encodage des données

Validation des données


```
<?php
mixed filter_var ( mixed $variable
[, int $filter = FILTER_DEFAULT
[, mixed $options ]] );

mixed filter_var_array ( array $data
[, mixed $definition
[, bool $add_empty = TRUE ]] );
?>
```

↪ Validation et nettoyage d'une donnée.

↪ Utilisation de filtres prédéfinis.

On a des fonctions analogues qui travaillent directement sur des **entrées** "externes" (get, post, cookie, etc.) avec `filter_input` et `filter_input_array`.

Filtres de validation

FILTER_VALIDATE_

- BOOLEAN
- INT
- EMAIL
- FLOAT
- IP
- URL
- etc.

Filtres de nettoyage

FILTER_SANITIZE_

- NUMBER_INT
- NUMBER_FLOAT
- STRING
- URL
- SPECIAL_CHARS

```
<?php
$email = "john.doe@example.com";

// Remove all illegal characters from email
$email = filter_var($email, FILTER_SANITIZE_EMAIL);

// Validate e-mail
if (!filter_var($email, FILTER_VALIDATE_EMAIL) === false) {
    echo("$email is a valid email address");
} else {
    echo("$email is not a valid email address");
}
?>
```

Eléments HTML de formulaire

Texte

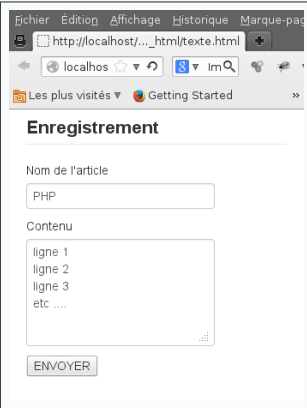
Cases à cocher

Listes

Upload de fichiers

- Champ texte : créer avec la balise `<input>` et l'attribut `type text`.
- Zone de texte (zone de texte sur plusieurs lignes) : créer avec la balise `<textarea>`

```
<form method="post" action="enregistrement.php">
  <fieldset>
    <legend><b>Enregistrement</b></legend>
    <label>Titre</label>
    <input type="text" name="titre">
    <label>Contenu</label>
    <textarea rows="5" name="contenu"></textarea>
  </fieldset>
  <button type="submit" name="accepter">
    ENVOYER
  </button>
</form>
```



The screenshot shows a web browser window with the address bar displaying `http://localhost/..._html/texte.html`. The browser's menu bar includes 'Fichier', 'Édition', 'Affichage', 'Historique', and 'Marque-page'. The page title is 'Enregistrement'. The form contains a text input field for 'Nom de l'article' with the value 'PHP', and a multi-line text area for 'Contenu' with the text 'ligne 1', 'ligne 2', 'ligne 3', and 'etc ...'. A button labeled 'ENVOYER' is at the bottom of the form.

```
<?php
echo $_POST['titre']."<br />";
echo nl2br($_POST['contenu']);
?>
```

Notez bien la fonction `nl2br` qui permet de convertir les sauts de lignes de la zone de texte en balise `
` pour l'affichage HTML.

Remarques

- champ texte qui permet de saisir un mot de passe : Il s'agit toujours de la balise `input` avec l'attribut `password`
- champ texte caché Il s'agit toujours de la balise `input` avec l'attribut `hidden`

```
<input type="password" name="nom" value="" />
<input type="hidden" name="nom" value="" />
```

HTML5 a rajouté de nouveaux type input

color	date	datetime
datetime-local	email	month
number	range	search
tel	time	url
	week	

Texte

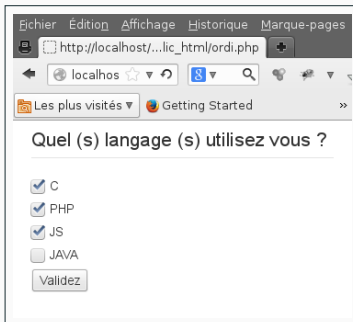
Cases à cocher

Listes

Upload de fichiers

balise input avec l'attribut checkbox.

```
<form method="post" action="traitement.php">
  <legend> Quel (s) langage (s) utilisez vous ?</legend>
  <input type="checkbox" name="langages[]" value="C"> C
  <input type="checkbox" name="langages[]" value="PHP"> PHP
  <input type="checkbox" name="langages[]" value="JS"> JS
  <input type="checkbox" name="langages[]" value="JAVA"> JAVA
  <button type="submit">
    Validez
  </button>
</form>
```



```
<?php
$langages = $_POST['langages'];
foreach ($langages as $unlangage)
    echo ('La case ' . $unlangage . 'a ete cochee <br>');
?>
```

balise input avec l'attribut radio : un seul choix possible.

```
<form action="reponse.php" methode=post>
  Qui etait Pythagore ?
  <input type="radio" name="question1" value=1>
  Un realisateur de film d'horreur.
  <input type="radio" name="question1" value=2>
  Un mathematicien grec.
  <input type="radio" name="question1" value=3>
  Un danseur mondain
  <input type=submit value=valider>
</form>
```

Il existe d'autres valeurs pour l'attribut type de la balise `input`

- Boutons
 - `reset` qui vide tous les champs d'un formulaire.
 - `submit` pour envoyer les données d'un formulaire.
 - `image` pour créer un bouton de validation graphique.

- Zone de données.
 - `file` pour soumettre un nom de fichier à uploader.
 - `hidden` pour envoyer une donnée cachée depuis un formulaire.

Texte

Cases à cocher

Listes

Upload de fichiers

Balise select

```
<form action="traiter.php" method=post>
  Quel est votre langage prefere ?
  <select name="langage">
    <option value=1>C</option>
    <option value=2>PHP</option>
    <option value=3>Bash</option>
    <option value=4>Assembleur Mips</option>
  </select>
  <input type=submit value=Valider>
</form>
```

Un seul choix possible. On peut néanmoins, en rajoutant `multiple` dans la balise, autorisé plusieurs choix.

Texte

Cases à cocher

Listes

Upload de fichiers


```
<form enctype="multipart/form-data"
  action="fileupload.php"
  method="post">
<input type="hidden" name="MAX_FILE_SIZE" value="100000" />
Transfere le fichier <input type="file" name="monfichier" />
<input type="submit" />
</form>
```

Le tableau `_FILE` contient un tableau pour chaque fichier transféré (clé correspondant à l'attribut `name`) contenant :

- `tmp_name` : le nom (et chemin) du fichier temporaire sous lequel le fichier a été stocké.
- `name` : le nom qu'avait le fichier dans l'espace de l'utilisateur
- `size` : la taille du fichier en octets
- `type` : le "MIME type" du fichier (ex : `text/html`, `image/gif`, etc.)