

Secteur Tertiaire Informatique  
Filière étude - développement

Développer des pages Web

## Les formulaires en HTML

Accueil

**Apprentissage**

Période en  
entreprise

Evaluation



# SOMMAIRE

I Définition .....	4
II La balise <FORM> (type block) .....	4
III Les champs de saisie de texte .....	5
III.1 Saisie de texte : balise <input> attribut type="text" (balise de type inline-block)	5
III.2 Saisie de texte multi-lignes : balise <textarea> (block) .....	6
III.3 Saisie d'un mot de passe .....	6
IV Les cases à cocher et les boutons radio .....	6
V Les combobox (listes de choix) .....	7
VI Mettre un label au champ de saisie .....	9
VII Grouper les champs .....	10
VIII Soumission (envoi) d'un formulaire .....	10
IX Inclusion de données cachées .....	11
X Le transfert de fichier .....	11

## I DEFINITION

Les formulaires permettent à l'internaute d'effectuer une saisie, des choix et d'envoyer les informations vers le serveur web afin d'y être traitées (souvent grâce à un programme tournant sur le serveur, script CGI, Java, Asp ou PhP par exemple). Tout échange de données entre le poste client et le serveur du site se fait via des saisies effectuées dans des formulaires.

Les saisies dans un formulaire peuvent s'effectuer par des champs de saisie, des zones de texte multi ligne, des listes de choix (combobox), des cases à cocher ou des boutons radio.

## II LA BALISE <FORM> (TYPE BLOCK)

Les formulaires sont encadrés par les balises **<form>** et **</form>**. La balise **<form>** possède les attributs suivants :

Attribut	Utilité
<b>action</b>	Attribut obligatoire qui indique l'url du script coté serveur (cgi, asp, php...) qui doit s'exécuter lors de l'envoi du formulaire. Pour le renvoi sur la même page mettre action="#". Cette valeur peut être surchargée par un attribut « formaction » sur un élément <button> ou <input>.
<b>method</b>	Il détermine la méthode d'envoi des données du formulaire vers le serveur. Deux méthodes : <b>post</b> : les données saisies dans le formulaire sont envoyées avec l'en-tête du protocole http. Elles n'apparaissent donc pas dans l'url. <b>get</b> : les données saisies sont ajoutées à l'url. Attention le nombre de caractères d'une URL est limité.
<b>enctype</b>	Spécifie le format des données envoyées. Sa valeur par défaut est "application/x-www-form-urlencoded". Elle vaut " <b>text/plain</b> " dans le cas d'un formulaire dont action est égal à "mailto :". Pour l'envoi de fichier, enctype doit être égal à " <b>multipart/form-data</b> ".
<b>accept</b>	Contient un ou plusieurs types MIME (Multipurpose Internet Mail Extensions) de la forme : "text/html", "image/jpeg" qui correspond au type de fichier acceptés par le serveur. Cet attribut a été retiré dans HTML5 et ne doit plus être utilisé. À la place, il faut utiliser l'attribut « accept » de l'élément <input>.

Toutes les balises de saisie d'informations contenues dans un formulaire devront avoir un attribut <name> dont sa valeur servira du coté serveur à récupérer les informations lors de la soumission du formulaire.

### III LES CHAMPS DE SAISIE DE TEXTE

#### III.1 SAISIE DE TEXTE : BALISE <INPUT> ATTRIBUT TYPE="TEXT" (BALISE DE TYPE INLINE-BLOCK)

L'internaute peut saisir une information dans un champ de saisie par l'intermédiaire de la balise **<input type="text" ....>**. En xHtml et html4 cette balise possède les attributs suivants :

Attribut	Utilité
<b>size</b>	Permet de définir un nombre de caractère visible pour le champ. Ex : size="15"
<b>maxlength</b>	Permet de définir un nombre de caractère maximum saisissable. Ex : maxlength="25"
<b>value</b>	Texte définit comme valeur par défaut qui sera affichée dans le champ
<b>disabled</b>	Si cet attribut à la valeur "disabled", la zone de saisie est inactive.
<b>readonly</b>	Permet d'utiliser la zone de saisie comme affichage. La saisie est interdite.

Exemple d'un champ de saisie:

```
<input type="text" name="nom" size="25" maxlength="50" value="Julie DUBOIS" />
```

Depuis html5, d'autres attributs sont apparus pour <input type="text"> :

- placeholder : une indication pour l'utilisateur sur ce qu'il doit saisir
- pattern : une expression régulière servant à valider la valeur rentrée
- required : la valeur doit être renseignée avant l'envoi du formulaire

Il existe maintenant en html5 d'autres types pour la balise <input> qui permettent de saisir des informations plus précises :

- <input type="date"> : saisie d'une date en année, mois, jour
- <input type="email"> : saisie d'un mail
- <input type="number"> : saisie d'un nombre avec les attributs possibles min, max, value et step. Il peut aussi être lié à une balise <datalist>
- <input type="range"> : « slider » permettant la saisie d'un nombre avec les attributs possibles min, max, value et step
- <input type="tel"> : saisie d'un numéro de téléphone
- <input type="url"> : saisie d'une url

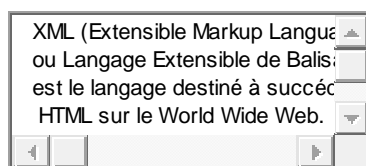
### III.2 SAISIE DE TEXTE MULTI-LIGNES : BALISE <TEXTAREA> (BLOCK)

Lorsque l'on veut saisir du texte sur plusieurs lignes, on passe par des zones de saisie multi-lignes à l'aide de la balise `<textarea>..</ textarea >`. Cette balise possède les attributs suivants :

Attribut	Utilité
<b>name</b>	Nom de la donnée. C'est le nom sous lequel l'information saisie ici sera connue et récupérée dans le script serveur déclenché par le formulaire.
<b>rows</b>	Nombre de lignes dans la zone de saisie.
<b>cols</b>	Nombre de caractères par ligne dans la zone de saisie
<b>readonly</b>	Indique que le contenu de la zone de saisie n'est pas modifiable

La valeur par défaut contenue dans la zone de saisie multi-ligne est le texte compris entre les balises `< textarea >` et `</ textarea >`.

Exemple d'une zone de saisie multi-ligne :



```
<textarea name="commentaire" cols="30" rows="5">
XML (Extensible Markup Language, ou Langage Extensible de Balisage)
est le langage destiné à succéder à HTML sur le World Wide Web.
Comme HTML (Hypertext Markup Language)...
</textarea>
```

### III.3 SAISIE D'UN MOT DE PASSE

Lorsque l'on veut définir un champ de saisie de type mot de passe, il faut passer par la balise :

`<input type="password" ....>`.

Les attributs sont quasi identiques à ceux du type "text". La seule différence est lors de la saisie dans le navigateur où les caractères tapés seront remplacés par des points.

## IV LES CASES A COCHER ET LES BOUTONS RADIO

Les données peuvent être sélectionnées par l'internaute dans des cases à cocher (plusieurs choix), ou dans des boutons radio (choix unique). Pour cela, on utilise les balises suivantes :

**<input type="checkbox" ....>**, pour les cases à cocher,  
**<input type="radio" ....>**, pour les boutons radio.

Ces balises possèdent les attributs suivants :

Attribut	Utilité
<b>name</b>	Nom de la donnée. C'est le nom sous lequel l'information saisie ici sera connue et récupérée dans le script serveur déclenché par le formulaire. Attention : pour créer des groupe de bouton radio (un seul coché possible) il faut que ces boutons portent le même nom.
<b>value</b>	Valeur envoyée lors de la soumission du formulaire lorsque cette case ou ce bouton est coché.
<b>checked</b>	Indique que l'option sera sélectionnée par défaut. La valeur de l'attribut est "checked".

Exemple de cases à cocher :

☐ Bleu      ☐ Rouge      ☐ Vert

```
<input type="checkbox" name="couleur1" value="bleu"> Bleu</input>
<input type="checkbox" name="couleur2" value="rouge"
checked="checked"> Rouge</input>
<input type="checkbox" name="couleur3" value="vert"> Vert</input>
```

Exemple de boutons radio :

☒ Bleu      ☐ Rouge      ☐ Vert

```
<input type="radio" name="couleur" value="bleu"> Bleu</input>
<input type="radio" name="couleur" value="rouge"> Rouge</input>
<input type="radio" name="couleur" value="vert" checked="checked">
Vert</input>
```

D'autres attributs existent et ne sont pas à négliger. Il s'agit des attributs permettant d'améliorer l'accessibilité :

- title
- accesskey
- tabindex

## V LES COMBOBOX (LISTES DE CHOIX)

L'internaute peut sélectionner les informations qu'il désire dans une liste pré-renseignée. La balise **<select>** permet de définir la liste des choix possibles.

Chaque choix sera défini par une balise **<option>**, incluse dans le **<select>**. C'est la première option qui est affichée par défaut.

Les attributs possibles de la balise **<select>** sont les suivants :

Attribut	Utilité
<b>name</b>	Nom de la donnée. C'est le nom sous lequel l'information saisie ici sera connue et récupérée dans le script serveur déclenché par le formulaire.
<b>size</b>	Si <b>size</b> vaut 1 (valeur par défaut), c'est une liste déroulante (un seul choix possible). Si <b>size</b> > 1, c'est une liste de choix avec ascenseur. (dans ce cas, on peut éventuellement avoir plusieurs choix : attribut multiple)
<b>multiple</b>	Dans le cas où size > 1 uniquement. Autorise plusieurs choix.

Les attributs possibles de la balise **<option>** sont les suivants :

Attribut	Utilité
<b>value</b>	Nom de la donnée pour l'option correspondante
<b>selected</b>	Détermine le choix par défaut

Exemple de liste déroulante :

```
<select name="couleur">
  <option value="bleu">Bleu</option>
  <option value="rouge">Rouge</option>
  <option value="vert">Vert</option>
</select>
```

Exemple de liste de choix :

```
<select name="couleur" size="3" multiple="multiple">
  <option value="bleu">Bleu</option>
  <option value="rouge">Rouge</option>
  <option value="vert">Vert</option>
</select>
```

## VI METTRE UN LABEL AU CHAMP DE SAISIE

La balise <label> permet de mettre un texte devant ou derrière le champ de saisie indiquant quelle est la nature de cette saisie.

```
<label>Entrer votre nom : </label><input type="text" name="nom" size="50"
maxlength="50" value="Julie DUBOIS" />
```

Il est possible d'associer ce label au champ de saisie permettant ainsi à l'internaute de cliquer sur le texte pour entrer dans le champ, sélectionner l'option voulue ou cocher la case choisie. Pour cela, la balise <input> doit avoir un attribut « id » qui l'identifie et la balise <label> doit avoir son attribut « for » égal à l'id.

```
<label for="zoneText1">Votre nom : </label><input id="zoneText1"
type="text" accesskey="s" name="nom" size="50" maxlength="50"
value="Julie DUBOIS" />
```

Pour l'accessibilité on respectera la position des labels par rapport au champ de saisie comme indiqué dans le tableau ci-dessous :

Élément HTML et son type	Ordre d'apparition	Exemple
input type="text"	Description du label, élément HTML	<label>Votre nom</label> <input type="text" name="txtNom" />
input type="password"	Description du label, élément HTML	<label>Votre mot de passe<</label> <input type="password" name="psw" />
input type="button"	Non applicable (utiliser l'attribut value à la place)	<input type="button" name="verifier" value="Vérifier la commande" />
input type="submit"	Non applicable (utiliser l'attribut value à la place)	<input type="submit" name="commande" value="Commander" />
input type="radio"	Élément HTML , description du label	<input type="radio" name="situation" value="marie" /><label> Oui, je suis marié</label> <input type="radio" name=" situation " value="celib" /><label> Non, je suis célibataire</label> 
input type="checkbox"	Élément HTML , description du label	<input type="checkbox" name="chkSouscrire" value="Souscrire" /><label> Souscrire à la newsletter</label>
select	Description du label, élément HTML	<label>Civilité</label> <select name="ddlCivilité"><option>M.</option>...</select>
textarea	Description du label, élément HTML	<label>Vos commentaires :</label> <textarea name="txtCommentaires"></textarea>
button	Non applicable (utiliser le contenu à la place)	<button name="cmdBouton">Envoyer, cliquer ici !</button>



## VII GROUPER LES CHAMPS

Dans un formulaire avec plusieurs zone de saisies il est généralement de bon ton de regrouper les champs par catégories. On peut par exemple séparer les champs relatifs à la situation de famille des champs qui demandent les coordonnées bancaires.

L'élément qui remplit ce rôle est **<fieldset>**.

Il suffit de faire délimiter les données d'un groupe par cette balise. Pour renseigner ce groupe on peut ajouter l'élément **<legend>**.

Exemple de groupement :

```
<form action="...."
  <fieldset>
    <legend>Vos coordonnées</legend>
    <p><label>Nom : </label><input type="text" /></p>
    <p><label>Prénom : </label><input type="text" /></p>
  </fieldset>
  <fieldset>
    <legend>Votre conjoint</legend>
    <p><label>Nom : <input type="text" /></label></p>
    <p><label>Prénom : <input type="text" /></label></p>
  </fieldset>
</form>
```

## VIII SOUMISSION (ENVOI) D'UN FORMULAIRE

L'envoi d'un formulaire, ou sa réinitialisation (remise à blanc des champs) s'effectue en cliquant sur deux boutons spéciaux : les boutons « submit » et « reset ». Ces boutons se déclarent à l'aide de la balise `<input type="submit" ....>`, pour l'envoi du formulaire, et `<input type="reset">` pour la réinitialisation des champs du formulaire.

Les attributs possibles sont les suivants :

Attribut	Utilité
<b>name</b>	Nom du bouton servant à le repérer dans du code javascript.
<b>value</b>	C'est le texte qui apparaît sur le bouton.
<b>title</b>	Le texte qui apparaîtra en info-bulle lorsque la souris sera sur le bouton

Exemple des 2 boutons :

Envoyer	Recommencer
---------	-------------

```
<input type="submit" value="Envoyer" />
<input type="reset" value="Recommencer" />
```

Pour soumettre un formulaire, on peut aussi utiliser une balise `<button type="submit">` qui permettra de créer un bouton de type texte ou image (voir exemple).

```
<button type="submit" value="Envoyer"
```

```
title="bouton d'envoi">Soumettre</button>
```

Soumettre

bouton d'envoi

Texte au  
passage de  
la souris

```
<button type="submit" value="Envoyer" title="bouton d'envoi">
</button>
```

commander

bouton d'envoi

## IX INCLUSION DE DONNEES CACHEES

Il est possible d'envoyer des données cachées au serveur (données non saisies transmises au serveur). Ceci se fait par l'intermédiaire de la balise

```
<input type="hidden" ....>.
```

Les attributs possibles sont les suivants :

Attribut	Utilité
<b>name</b>	Nom de la donnée. C'est le nom sous lequel l'information saisie ici sera connue et récupérée dans le script serveur déclenché par le formulaire.
<b>value</b>	C'est la valeur de la donnée.

## X LE TRANSFERT DE FICHIER

Le client peut vouloir transmettre au serveur un fichier texte ou image. Pour cela il faut un formulaire `<form>` dont l'attribut « `enctype` » est "multipart/form-data". La balise `<input>` aura son attribut « `type` » à "file".

Exemple :

```
<form action="#" method="post" enctype="multipart/form-data">
  <label>Sélectionner votre fichier image (.jpg) : </label>
  <input type="file" accept="image/jpeg" name="fichierImage" />
</form>
```

Sélectionner votre fichier image (.jpg) :  Parcourir...

**Etablissement référent**  
*Direction de l'ingénierie Neuilly*

**Equipe de conception**  
*Didier Bonneau*

**Remerciements :**

Documents de formation publiés  
sous [licence GFDL](#)

Date de mise à jour 08/09/2016



**afpa / Direction de l'Ingénierie** 13 place du Générale de Gaulle / 93108 Montreuil  
Cedex  
association nationale pour la formation professionnelle des  
adultes  
Ministère des Affaires sociales du Travail et de la  
Solidarité