

Tableau – Ligne *tr*, titre *th*, cellule *td*

```
<table border="1">
  <caption>Légende</caption>
  <tr>
    <th>Titre 1</th>
    <th>Titre 2</th>
  </tr>
  <tr>
    <td>case 1.1</td>
    <td>case 1.2</td>
  </tr>
  <tr>
    <td>case 2.1</td>
    <td>case 2.2</td>
  </tr>
</table>
```



Titre 1	Titre 2
case 1.1	case 1.2
case 2.1	case 2.2

L'attribut `border="1"` est utilisé uniquement à des fins pédagogiques. Il sert à visualiser les bordures dans l'exemple (sinon elles seraient invisibles).

Dans un développement réel, les bordures sont affichées à l'aide de styles CSS.

Tableau – Fusion de colonnes – *colspan*

```
<table border="1">
  <caption>Légende</caption>
  <tr>
    <th>Titre 1</th>
    <th>Titre 2</th>
    <th>Titre 3</th>
  </tr>
  <tr>
    <td>case 1.1</td>
    <td>case 1.2</td>
    <td>case 1.3</td>
  </tr>
  <tr>
    <td colspan="2">case 2.1 + 2.2</td>
    <td>case 2.3</td>
  </tr>
  <tr>
    <td>case 3.1</td>
    <td>case 3.2</td>
    <td>case 3.3</td>
  </tr>
</table>
```



Titre 1	Titre 2	Titre 3
case 1.1	case 1.2	case 1.3
case 2.1 + 2.2		case 2.3
case 3.1	case 3.2	case 3.3

Tableau – Fusion de lignes – *rowspan*

```
<table border="1">
  <caption>Légende</caption>
  <tr>
    <th>Titre 1</th>
    <th>Titre 2</th>
    <th>Titre 3</th>
  </tr>
  <tr>
    <td>case 1.1</td>
    <td>case 1.2</td>
    <td rowspan="2">case<br>1.3 + 2.3</td>
  </tr>
  <tr>
    <td>case 2.1</td>
    <td>case 2.2</td>
  </tr>
  <tr>
    <td>case 3.1</td>
    <td>case 3.2</td>
    <td>case 3.3</td>
  </tr>
</table>
```



Titre 1	Titre 2	Titre 3
case 1.1	case 1.2	case 1.3 + 2.3
case 2.1	case 2.2	
case 3.1	case 3.2	case 3.3

Tableau – *thead*, *tbody*, *tfoot*

```
<table border="1">
  <caption>Légende</caption>
  <thead>
    <tr>
      <th>Titre 1</th>
      <th>Titre 2</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>case 1.1</td>
      <td>case 1.2</td>
    </tr>
    <tr>
      <td>case 2.1</td>
      <td>case 2.2</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <th>Titre 1</th>
      <th>Titre 2</th>
    </tr>
  </tfoot>
</table>
```



Titre 1	Titre 2
case 1.1	case 1.2
case 2.1	case 2.2
Titre 1	Titre 2

Ces trois éléments ne sont pas obligatoires. Ils permettent généralement de mieux structurer les gros tableaux.

Ils ne sont pas utiles sur les petits tableaux.

Audio et vidéo

Jusqu'à HTML4, nécessité d'un *plug-in* (ex. flash)

HTML5 intègre directement audio et vidéo

Attention aux formats !

Formats de **conteneurs** : Ogg, WebM, MP3, MP4, Wav



→ **flux audio/vidéo compressés (MPEG, Vorbis, VP8, etc.)**

→ **métadonnées (titre, sous-titres, descripteurs divers)**

Voir : http://fr.wikipedia.org/wiki/Format_conteneur

Vidéo

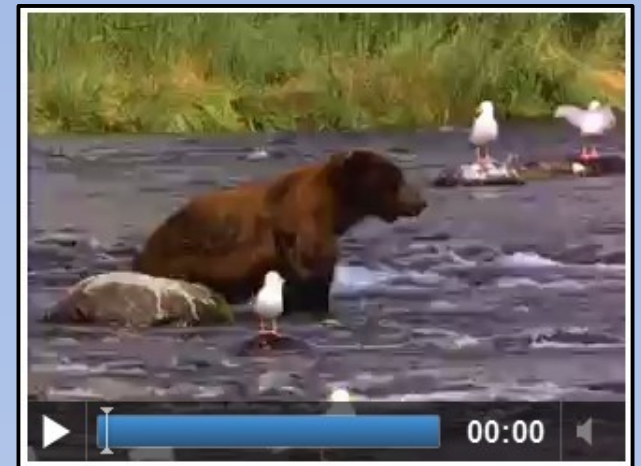
3 formats de conteneur vidéo pris en charge :

Browser	MP4	WebM	Ogg
Internet Explorer 9+	YES	NO	NO
Chrome 6+	YES	YES	YES
Firefox 3.6+	NO	YES	YES
Safari 5+	YES	NO	NO
Opera 10.6+	NO	YES	YES

H264
Propriétaire
e

VP8
ouvert

Theora



```
5 <video width="320" height="240" controls="controls">
6   <source src="movie.mp4">
7   <source src="movie.ogg">
8   <source src="movie.webm">
9   Votre navigateur ne prend pas en charge cette balise.
10 </video>
```

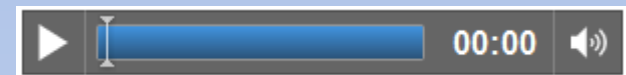
Attributs de l'élément *video*

- **poster** = image à la place de la vidéo (avant lancement)
- **controls** = barre de contrôle
- **width, height** = largeur, hauteur
- **loop** = jouer en boucle
- **autoplay** = jouer dès lchargement de la page
- **preload** = préchargement de la vidéo
 - **auto** (par défaut) : laisse le navigateur décider
 - **metadata** : charge uniquement les métadonnées (durée,dimensions,...)
 - **none** : pas de préchargement. Utile pour ne pas gaspiller de bande passante sur votre site.

Audio

3 formats de conteneur audio pris en charge :

Navigateur	MP3	Wav	Ogg
Internet Explorer 9+	YES	NO	NO
Chrome 6+	YES	YES	YES
Firefox 3.6+	NO	YES	YES
Safari 5+	YES	YES	NO
Opera 10+	NO	YES	YES



```
<audio controls="controls">
  <source src="song.ogg" type="audio/ogg">
  <source src="song.mp3" type="audio/mpeg">
  Votre navigateur ne prend pas en charge cette balise.
</audio>
```

Attributs : *preload, autoplay, loop et controls*

Les formulaires

- Un formulaire permet au visiteur d'un site web de saisir des informations.
- Ces informations ne peuvent pas être traitées par le langage HTML. Elles devront être traitées par un autre langage, par exemple le PHP.

Le HTML ne sert qu'à la saisie et à l'envoi des données !!!

Exemple de formulaire

Vos commentaires

Aidez-nous à améliorer notre site:

Votre nom

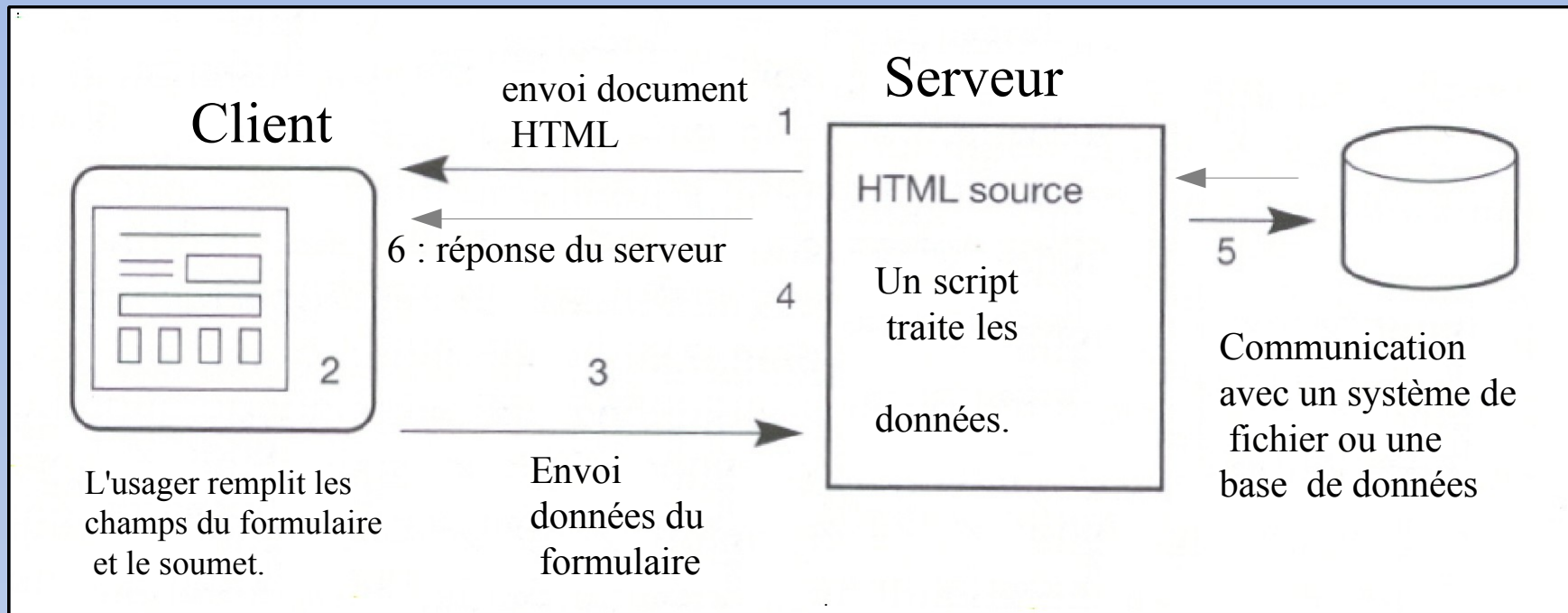
Vos commentaires:

Votre adresse de courrier électronique:

Indiquez les éléments du site que vous aimez:

Design du site ☐ Les liens ☐ Facilité de navigation ☐ Les images ☐

Exemple de formulaire



L'élément *form*

Tout formulaire doit être contenu dans un élément *form*.

Les deux principaux attributs d'un élément *form* sont :

L'attribut *method*

L'attribut *action*

```
7 <form method="post" action="traitement.php">  
8   <!-- zone de saisie du formulaire -->  
9 </form>
```

L'attribut *method*

- L'attribut *method* définit la méthode à utiliser pour envoyer les données saisies par le visiteur.
- Seules deux méthodes sont possibles :
 - La méthode *get*
 - La méthode *post*

La méthode *get*

- Les informations sont envoyées à travers l'URL de la page.
- Les données sont visibles dans l'URL.
- Le maximum de caractères qui peut être envoyé est de 255.

Il est préférable de ne pas utiliser cette méthode !!!

La méthode *post*

- Les données ne sont pas envoyées à travers l'URL de la page, mais dans un bloc de données.
- Les données ne sont pas visibles dans l'URL.
- Le nombre de caractères qui peut être envoyé n'est pas limité.

Il est préférable d'utiliser cette méthode !!!

L'attribut *action*

- L'attribut *action* définit le fichier qui recevra les données.
- Ce fichier est généralement un fichier PHP.

L'élément *input*

- L'élément *input* permet à un visiteur d'effectuer des saisies de données à travers différents types de zones de saisie.
- Le type de zone de saisie se définit à l'aide de l'attribut *type*.

L'attribut *type* (1)

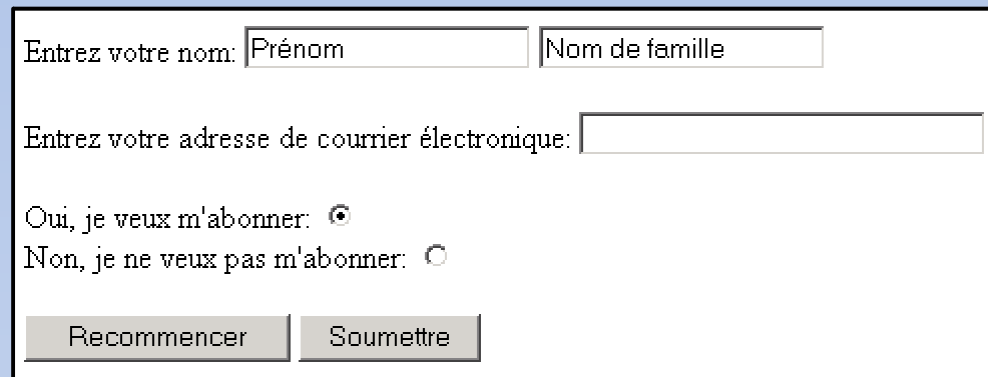
- Texte `<input type="text" ...>`
- Mot de passe `<input type="password" ...`
`>`
- Boîte à cocher `<input type="checkbox" ...`
`>`
- Bouton radio `<input type="radio" ...>`
- Texte caché `<input type="hidden" ...>`
- Soumettre (bouton) `<input type="submit" ...>`
- Soumettre (image) `<input type="image" ...>`
- Réinitialisation `<input type="reset" ...>`
- Fichier `<input type="file" ...>`
- Bouton `<input type="button" ...>`

L'attribut *type* (2)

■ Couleur	<code><input type="color" ...></code>
■ Date	<code><input type="date" ...></code>
■ Date et heure	<code><input type="datetime" ...</code>
■	<code>></code>
■ Heure	<code><input type="time" ...></code>
■ Mois	<code><input type="month" ...></code>
■ Semaine	<code><input type="week" ...></code>
■ E-mail	<code><input type="email" ...></code>
■ Nombre entier	<code><input type="number" ...></code>
■ Curseur	<code><input type="range" ...></code>
■ Recherche	<code><input type="search" ...></code>
■ Téléphone	<code><input type="tel" ...></code>
■ URL	<code><input type="url" ...></code>

Les types *submit*, *image* et *reset*

- Le type *submit* est un bouton qui permet d'envoyer les données du formulaire vers le fichier de traitement. Ce dernier a été défini dans l'attribut *action* de l'élément *form*.
- Le type *image* est équivalent au bouton *submit*, mais le bouton s'affiche sous forme d'image.
- Le type *reset* est un bouton qui permet de réinitialiser le formulaire.



Entrez votre nom:

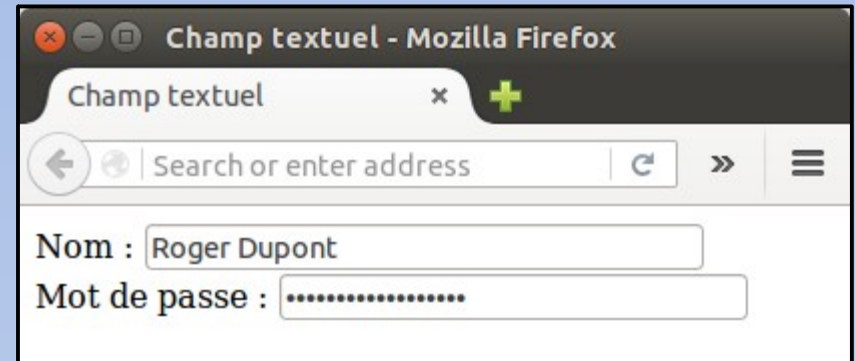
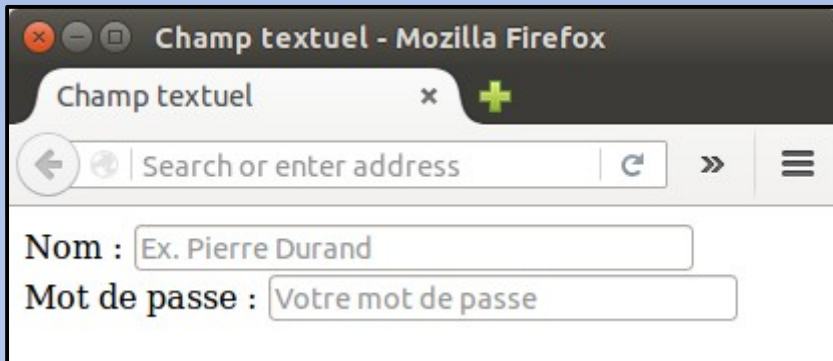
Entrez votre adresse de courrier électronique:

Oui, je veux m'abonner: ☒

Non, je ne veux pas m'abonner: ☐

Champs textuels et libellés – *text*, *label*

```
<label for="nom">Nom :</label>
<input type="text" id="nom" name="nom" size="25" maxlength="100" placeholder="Ex. Pierre Durand">
<br>
<label for="pass">Mot de passe :</label>
<input type="password" id="pass" name="pass" placeholder="Votre mot de passe">
```

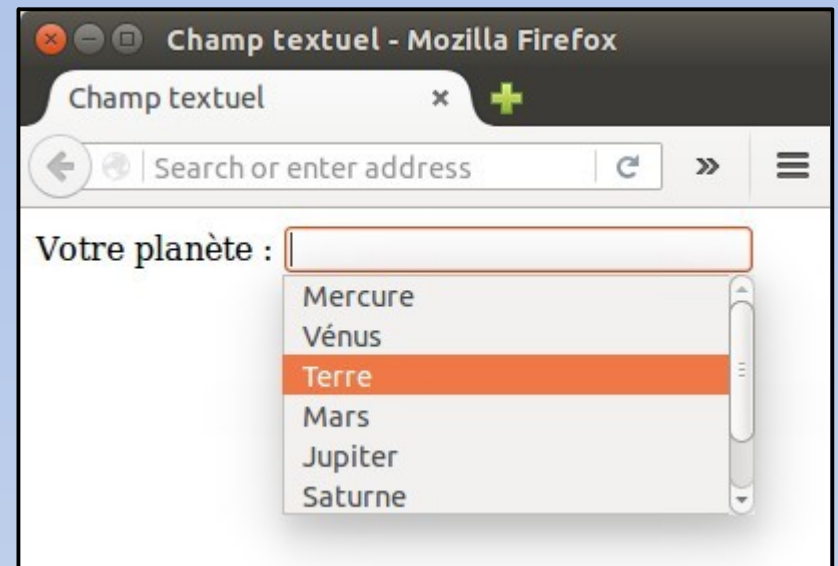
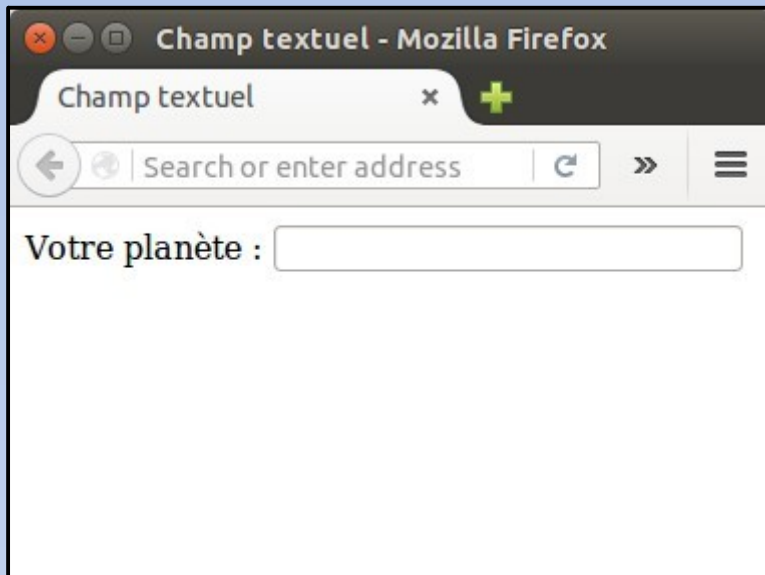


Il est également possible de pré-remplir le champ avec l'attribut *value*.

Ne pas confondre *value* et *placeholder*

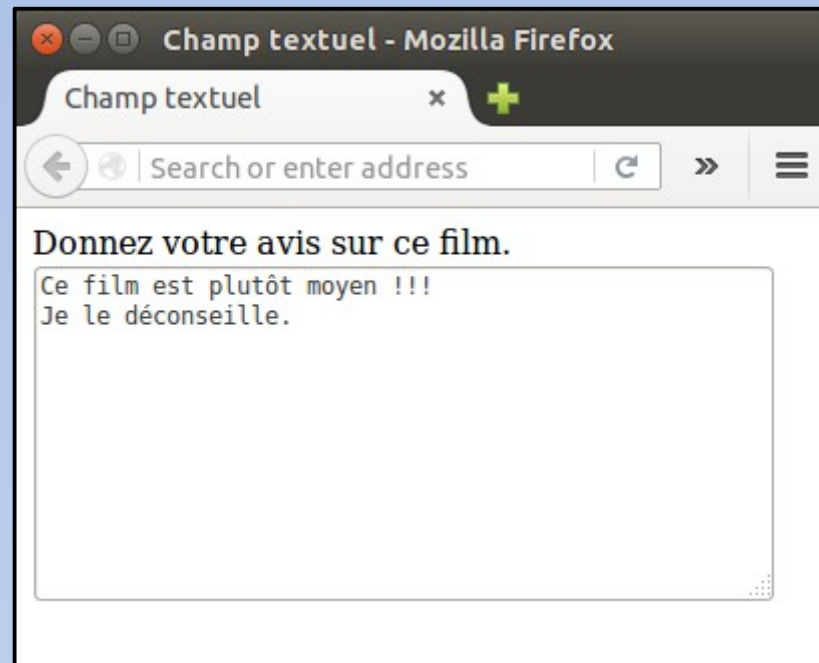
Champs textuels et complétion automatique

```
<label for="homeworld">Votre planète :</label>
<input type="text" name="homeworld" id="homeworld" list="planets">
<datalist id="planets">
  <option value="Mercure">
  <option value="Vénus">
  <option value="Terre">
  <option value="Mars">
  <option value="Jupiter">
  <option value="Saturne">
  <option value="Uranus">
  <option value="Neptune">
</datalist>
```



Zone de texte multi-ligne - *textarea*

```
<label for="multiligne">
    Donnez votre avis sur ce film.
</label>
<br>
<textarea name="multiligne" id="multiligne" rows="10" cols="50">
    Ce film est plutôt moyen !!!
    Je le déconseille.
</textarea>
```



E-mail, URL, Téléphone – *email*, *url*, *tel*

```
<label for="email">Adresse e-mail :</label>
<input type="email" id="email" name="email">
<br>

<label for="website">Site Web :</label>
<input type="url" id="website" name="website">
<br>

<label for="phone">Téléphone :</label>
<input type="tel" id="phone" name="phone">
<br>
```

Adresse e-mail :

Site Web :

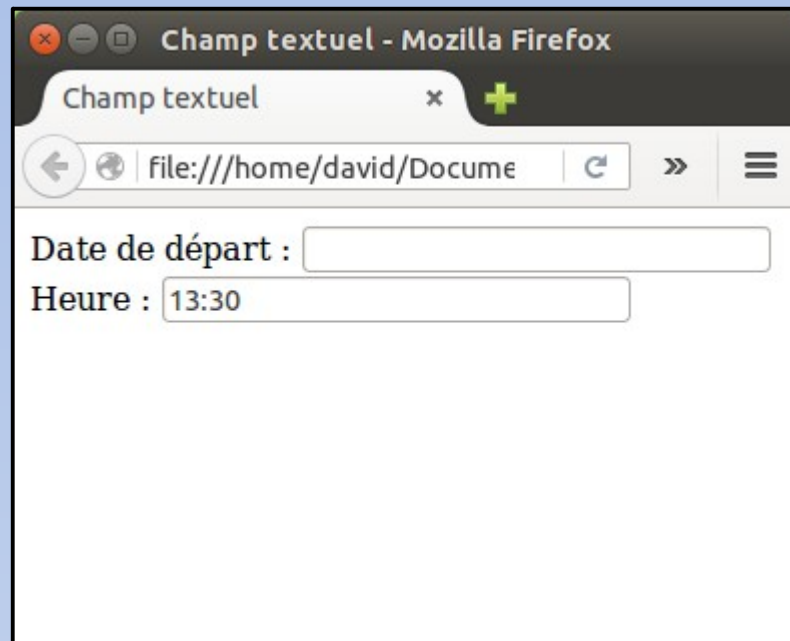
Téléphone :

Sur les *smartphones*,
l'affichage du clavier s'adapte
au type de contrôle.



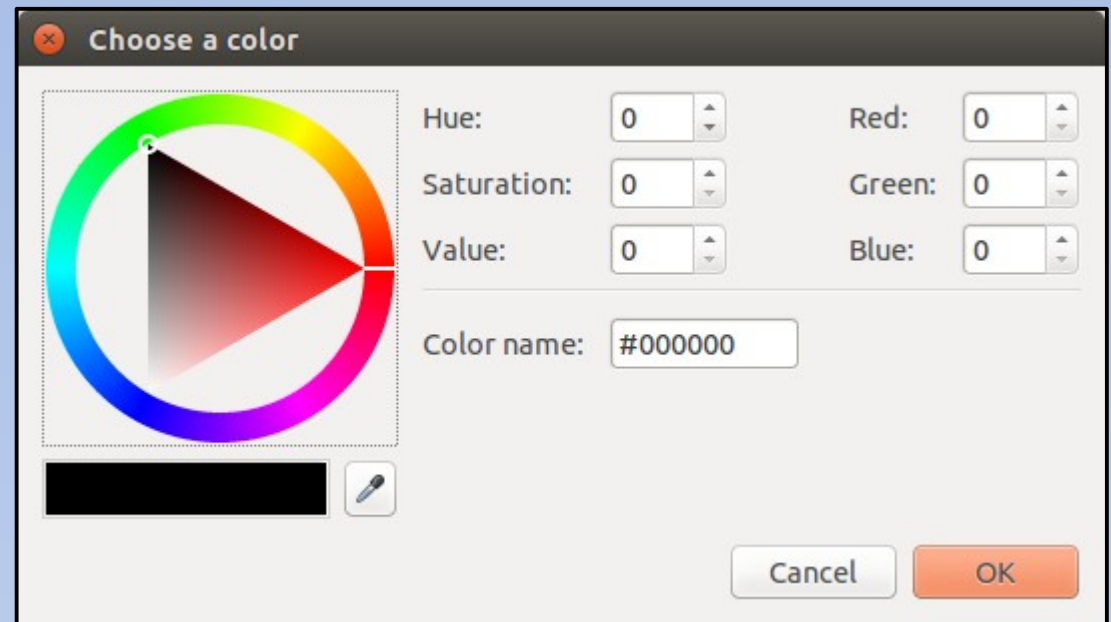
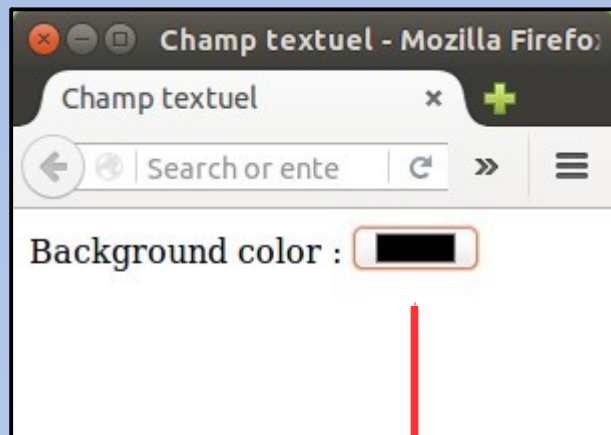
Date et heure – *date, time*

```
<label for="dtstart">Date de départ :</label>  
<input type="date" id="dtstart" name="dtstart"> <br>  
<label for="t">Heure :</label>  
<input type="time" id="t" min="11:00" max="21:00" value="13:30">
```



Couleur – *color*

```
<label for="bgcolor">Background color :</label>  
<input id="bgcolor" name="bgcolor" type="color">
```



Les cases à cocher – *checkbox*

Cochez le style de film que vous aimez :

```
<br>
<input type="checkbox" name="horreur" id="horreur">
<label for="horreur">Horreur</label>
<br>
<input type="checkbox" name="romantique" id="romantique">
<label for="romantique">Romantique</label>
<br>
<input type="checkbox" name="thriller" id="thriller">
<label for="thriller">Thriller</label>
<br>
<input type="checkbox" name="comedie" id="comedie">
<label for="comedie">Comédie</label>
```



Les zones d'options – *radio*

```
<form action="traitement.php" method="post">
  <label for="email">Adresse e-mail :</label><br>
  <input type="text" name="email" id="email" size="20"><br>

  Indiquez le format sous lequel vous aimeriez recevoir notre bulletin :<br>
  <input type="radio" name="format" value="texte" checked>Texte<br>
  <input type="radio" name="format" value="html">HTML<br>
  <input type="radio" name="format" value="pdf">PDF<br>
  <input type="submit" value="Soumettre">
</form>
```



The screenshot shows a Mozilla Firefox browser window titled "Champ textuel". The address bar displays "file:///home/david/Documents/En:". The form content is as follows:

Adresse e-mail :

Indiquez le format sous lequel vous aimeriez recevoir notre bulletin :

☒ Texte

☐ HTML

☐ PDF

Liste déroulante – *select* (1)

- Menu déroulant : l'utilisateur fait un ou plusieurs choix dans une liste.
- Élément *select* qui contient plusieurs éléments *option* qui définissent les choix...

`<select>...<option>...</select>`

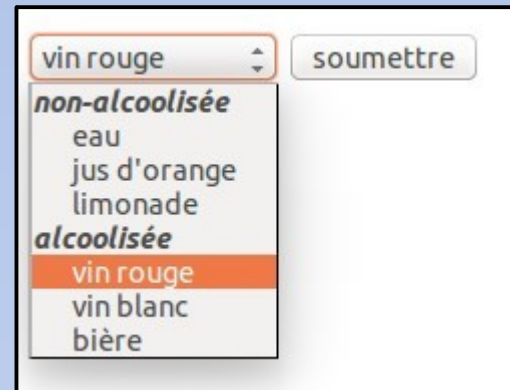
- Attributs de *select* :
 - ♦ *name* : nom du menu ;
 - ♦ *size* : nombre de choix visibles à la fois ;
 - ♦ *multiple* : s'il est présent, plusieurs choix simultanés possibles.

Liste déroulante – *select* (2)

- L'élément *option* ne peut contenir que du texte.
`<option></option>`
- Les attributs de *option* :
 - ♦ *value* : valeur utilisée si ce choix est fait.
 - ♦ *selected* : si présent, cette option est choisie par défaut.

Liste déroulante – *select* (3)

```
<select name="boisson">
  <optgroup label="non-alcoolisée">
    <option value="eau"> eau </option>
    <option value="jus_orange"> jus d'orange </option>
    <option value="limo"> limonade </option>
  </optgroup>
  <optgroup label="alcoolisée">
    <option value="rouge" selected> vin rouge </option>
    <option value="blanc"> vin blanc </option>
    <option value="biere"> bière </option>
  </optgroup>
</select>
<input type="submit" value="soumettre">
```



The screenshot shows a web form with a dropdown menu and a submit button. The dropdown menu is open, displaying a list of options. The options are grouped into two categories: 'non-alcoolisée' and 'alcoolisée'. The 'non-alcoolisée' group includes 'eau', 'jus d'orange', and 'limonade'. The 'alcoolisée' group includes 'vin rouge', 'vin blanc', and 'bière'. The 'vin rouge' option is currently selected and highlighted in orange. To the right of the dropdown menu is a submit button labeled 'soumettre'.

Category	Option
non-alcoolisée	eau
	jus d'orange
	limonade
alcoolisée	vin rouge (selected)
	vin blanc
	bière

Liste à choix multiples (*select + multiple*)

```
<form action="" method="post">
  <p>Quels sont les serveurs qui vous intéressent ?</p>
  <select name="produits" multiple size="4">
    <option value="ibmnt">IBM NT</option>
    <option value="aix">IBM AIX</option>
    <option value="hpux">HP UX</option>
    <option value="compaqnt">Compaq NT</option>
    <option value="win98">Microsoft Win98</option>
    <option value="ibmlinux">IBM linux</option>
    <option value="hplinux">HP LINUX</option>
    <option value="compaqlinux">Compaq LINUX</option>
  </select>
  <input type="submit">
</form>
```

Quels sont les serveurs qui vous intéressent ?

IBM NT	
IBM AIX	
HP UX	
Compaq NT	



Quels sont les serveurs qui vous intéressent ?

IBM NT	
IBM AIX	
HP UX	
Compaq NT	

Soumettre un fichier – *file*

Le type *file* permet à l'utilisateur d'envoyer un fichier au serveur à partir d'un formulaire.

```
<h2>Concours de photographies numériques</h2>
<form method="POST" action="traitement.php" enctype="multipart/form-data">
  <label for="nom">Entrez votre nom :</label>
  <br>
  <input type="text" name="nom" id="nom" size="30">
  <br><br>
  Vous devez soumettre un fichier de format JPG ou GIF :
  <br>
  <input type="file" name="fichier">
  <br>
  <input type="submit" value="Soumettre">
</form>
```

Concours de photographies numériques

Entrez votre nom :

Vous devez soumettre un fichier de format JPG ou GIF :

Browse...

No file selected.

Soumettre

Éléments *fieldset* et *legend*

Ces deux éléments permettent de structurer les formulaires.

```
<fieldset>
  <legend>Informations personnelles</legend>
  <label for="nom">Nom :</label><input type='text' id="nom">
  <br>
  <label for="prenom">Prénom :</label><input type='text' id="prenom">
</fieldset>
<fieldset>
  <legend>Favoris</legend>
  <label for="auteur">Auteur favori :<input type='text' id="auteur">
  <br>
  <label for="livre">Livre favori :<input type='text' id="livre">
</fieldset>
```

The image shows a visual representation of the HTML form structure. It consists of two distinct sections, each enclosed in a rectangular box. The first section is titled 'Informations personnelles' and contains two input fields: 'Nom :' and 'Prénom :'. The second section is titled 'Favoris' and contains two input fields: 'Auteur favori :' and 'Livre favori :'. The visual layout demonstrates how the `fieldset` and `legend` elements are used to group related form elements.

Attributs *required* et *autofocus*

Il est possible de rendre un champ obligatoire en lui ajoutant l'attribut *required*.

Par exemple :

```
<input type="text" name="film" id="film" required>
```

Il est possible de placer le curseur dans un champ dès l'ouverture d'une page en lui ajoutant l'attribut *autofocus*.

Par exemple :

```
<input type="text" name="film" id="film" autofocus>
```

Éléments de type *Block* et *Inline*

Il existe deux catégories d'élément :

1. Les éléments de type *block*
2. Les éléments de type *inline*

Éléments de type *Block* (1)

- Les éléments de type *block* peuvent produire un retour à la ligne.
- Lorsque deux éléments de type *block* se succèdent sur une page, ils sont positionnés **par défaut** l'un sous l'autre.
- Deux éléments de type *block* peuvent également se positionner l'un à côté de l'autre.

Éléments de type *Block* (2)

- La plupart du temps, un élément de type *block* peut contenir un autre élément de type *block*.
- La plupart du temps, un élément de type *block* peut contenir un élément de type *inline*.

Éléments de type *Block* (3)

Quelques éléments de type *block* :

- *h1...h6* : Titre
- *p* : Paragraphe
- *ul, ol, dl* : Listes (non ordonnée, ordonnée ou avec définition)
- *li, dt, dd* : Élément de liste, définition, et titre de définition
- *table* : Tableau
- *pre* : Texte dont la mise en page est préservée
- *form* : Formulaire

Éléments de type *Inline* (1)

- Les éléments de type *inline* ne produisent jamais de retour à la ligne.
- Lorsque deux éléments de type *inline* se succèdent sur une page, ils sont positionnés l'un à côté de l'autre.
- Deux éléments de type *inline* ne peuvent pas se positionner l'un sous l'autre.

Éléments de type *Inline* (2)

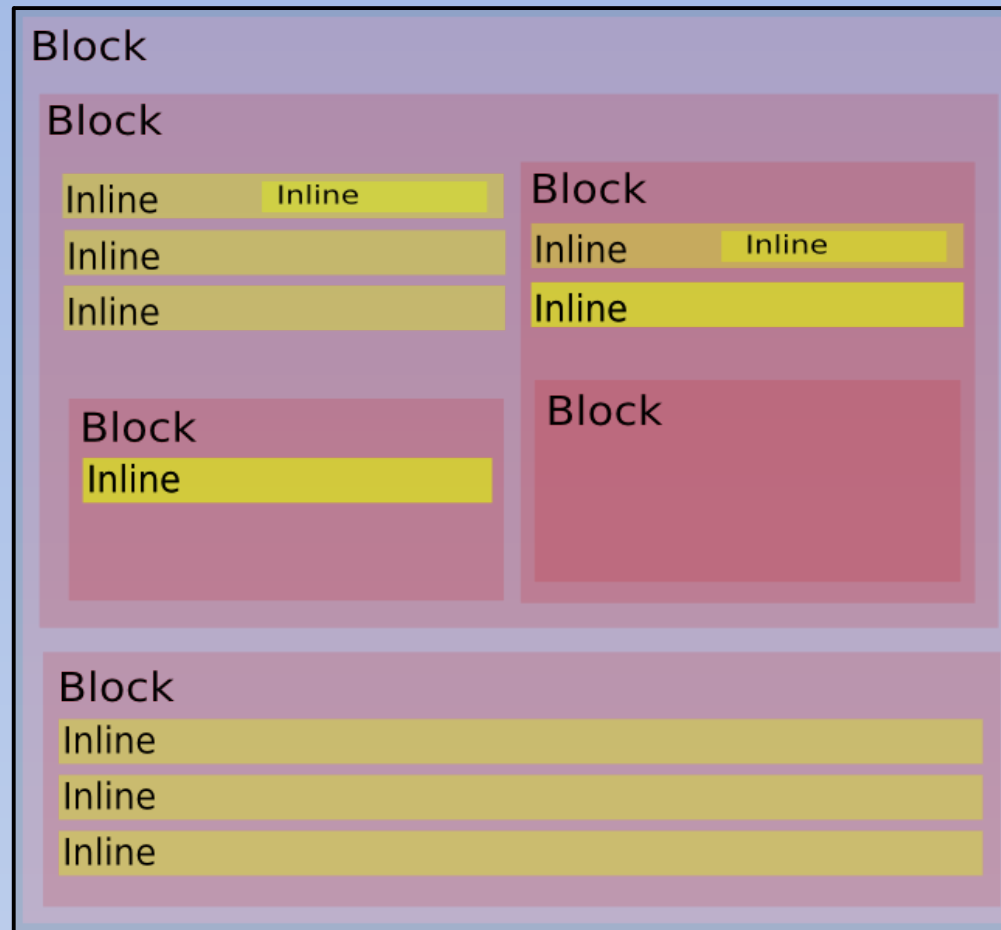
- La plupart du temps, un élément de type *inline* peut contenir un autre élément de type *inline*.
- Un élément de type *inline* ne peut jamais contenir un élément de type *block*.

Éléments de type *Inline* (3)

Quelques éléments de type *inline* :

- a : Hyperlien
- strong : Définit du contenu important
- em : Définit une emphase
- mark : Définit du contenu qui doit être repéré facilement
- img : Image
- br : Saut de ligne
- input : Zone de saisie

Encastrements *Block / Inline* (1)



Encastrements *Block / Inline* (2)

Possible :

```
<p><strong>Hello World</strong></p>
```

Impossible :

```
<strong><p>Hello World</p></strong>
```

Compatibilité IE < 9

- Certaines balises du HTML 5 ne sont pas compatibles avec les versions antérieures à *Internet Explorer 9*.
- Pour rendre ces balises compatibles avec les anciennes versions, il faut ajouter les commentaires ci-dessous dans son code HTML 5.
- Ces commentaires doivent se placer dans l'élément *head*.

```
<head>  
  <meta charset="utf-8">  
  <!--[if lt IE 9]>  
  <script src="http://html5shiv.googlecode.com/svn/trunk/html5.js"></script>  
  <![endif]-->  
  <title>Hello World</title>  
</head>
```

Référencer son site

- **Pour faire connaître son site :**
 - **Enregistrer son site** manuellement sur les principaux moteurs de recherche (google, yahoo, Bing, ...)
 - Utiliser des produits commerciaux ou gratuits qui font le **référencement automatiquement**, voire passer par des **agences spécialisées**
 - **Faire inscrire des liens vers son propre site** dans des sites déjà référencés.
- **Pour que le site soit correctement référencé :**
 - Les moteurs de recherche utilisent le **nom de domaine**, l'en-tête et d'autres métadonnées.
 - Renseigner les balises **title** et **meta**.
 - Utiliser les attributs **title** et **alt** dans les images et vidéos.

Complément cours HTML

Les images cliquable-réactives

QUELQUES BALISES MULTIMEDIA : Images entre <body> et </body>

- Charger une image du serveur nyx à votre document

```
<IMG SRC="http://nyx.unice.fr/images/chato.jpg">
```

- Charger une image de votre répertoire courant en spécifiant la taille

```
<IMG SRC="theatre.jpg" WIDTH=200 HEIGHT=500 Border = 10>
```

- Afficher en grand d'une image en cliquant sur l'image :

```
<A HREF="images/verdure.jpg">
```

```
<IMG SRC="images/verdure.jpg" WIDTH=200 HEIGHT=500 Border = 0  
ISMAP>
```

```
</A>
```

Complément cours HTML

Les images cliquable-réactives

IMAGES CLIQUABLES entre <body> et

- Une image cliquable est une image contenant des zones servant les liens vers les autres documents

- Pour la mise en place d'une image cliquable, on doit composer 2 balises :

une balise Image IMG pour charger l'image

Une balise MAP pour définir les zones et les liens concernés
la référence à la balise Map est faite par attribut **USEMAP**

- Balise MAP :

Syntaxe : <MAP NAME=Nom>

```
<AREA SHAPE=valeur COORDS= "valeur_coordonnées" HREF=URL>
```

```
<AREA SHAPE=valeur COORDS= "valeur_coordonnées" HREF=URL>
```

```
<AREA SHAPE=valeur COORDS= "valeur_coordonnées" HREF=URL>
```

```
<AREA SHAPE=valeur COORDS= "valeur_coordonnées" HREF=URL>
```

```
</MAP>
```

Complément cours HTML Les images cliquable-réactives

IMAGES CLIQUABLES entre <body> et </body>

□ Syntaxe de balise MAP (suite)

Les attributs de AREA prennent les valeurs suivantes :

SHAPE= rect|circle|poly|default la zone de définition est un rectangle, un cercle, un polygone ou le reste de la figure qui n'a pas été décrite. Ainsi :

- rect : **est défini par les coins opposés** (par exemple "130,10,170,90")
- circle : **est défini par son centre et par le rayon** (par exemple "50,50,10")
- poly : **est défini par un ensemble de points** (par exemple "250,10 210,90 290,90")
- default : **est défini par les points non définis précédemment.**

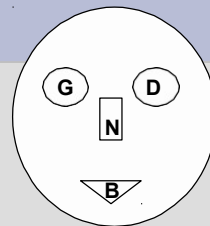
COORDS= des valeurs entre côtes et séparées par des virgules comme décrit précédemment.

HREF= l'URL qui sera atteinte après le clic.

Complément cours HTML Les images cliquable-réactives

IMAGES CLIQUABLES : Exemple

- G est un cercle et défini par 1 point (g1,g2) et un rayon r1
- D est un cercle et défini par 1 point (d1,d2) et un rayon r2
- N est un rectangle et défini par 2 points opposés : (n1;n2) et (n3,n4)
- B est un polygone et défini par 3 sommets : (b1,b2), (b3,b4),(b5,b6)



<http://www.serveur.fr/images/tete.gif>

```
<IMG SRC="http://www.serveur.fr/images/tete.gif" USEMAP="#unmap" >
<MAP Name="unmap"
<AREA SHAP=circle coords=" g1,g2,r1 " href=" œil-gauche.html >
<AREA SHAP=circle coords=" d1,d2,r2 " href=" œil-droit.html >
<AREA SHAP=rect coords=" n1,n2,n3,n4 " href=" nez.html >
<AREA SHAP=poly coords=" b1,b2,b3,b4,b5,b6 " href="bouche.html >
</MAP>
```