

Développement Web



Avant de commencer ...

En quoi consiste le développement web ?

Le développement web repose sur l'utilisation des langages (HTML/CSS, JavaScript, PHP...) pour écrire des programmes qui sont ensuite exécutés par les ordinateurs. Les instructions sont mises en place sur Internet et sont effectuées sur des serveurs. En fonction des besoins des propriétaires du site ou des pages web, ces dernières peuvent être constituées uniquement de textes et d'éléments graphiques ressemblant à un document ou être interactives en affichant des informations qui évoluent.

Qu'est-ce qu'un développeur web ?

Un développeur web est un informaticien spécialisé dans la programmation ou expressément impliqué dans le développement des applications du World Wide Web, ou des applications qui sont exécutées à partir d'un serveur web sur un navigateur web et qui utilisent le protocole HTTP comme vecteur de transmission de l'information.

Qui ce qu'un site Web ?

Un site web est un ensemble de pages web visualisables dans un navigateur. Ces pages web sont reliées entre elles par des liens qui permettent de passer de l'une à l'autre. L'ensemble des pages d'un site web est en général accessible sous une adresse au même nom de domaine.

Pourquoi le langage HTML ?

HTML est un langage de description de document utilisé sur Internet pour faire des pages Web. Son sigle signifie « HyperText Markup Language » en anglais, littéralement « langage de marquage hypertexte ». Le balisage HTML est incorporé dans le texte du document et est interprété par un navigateur Web.

Pourquoi on utilise le CSS ?

CSS est l'acronyme de « Cascading Style Sheets » ce qui signifie « feuille de style en cascade ». Le CSS est un langage informatique utilisé pour mettre en forme les fichiers HTML sur un site web. Ainsi, les fichiers CSS permettent de gérer l'aspect graphique d'une page en HTML.

Les différentes versions de HTML et CSS ?

Au fil du temps, les langages HTML et CSS ont beaucoup évolué. Dans la toute première version de HTML (HTML 1.0) il n'était même pas possible d'afficher des images !

Comment fonctionne un site Web ?

1. Premièrement, le client commande une **page Web** au serveur. Il saisit l'URL d'un site dans son navigateur. Celui-ci envoie immédiatement une **requête web** au serveur.
2. Ensuite, le **serveur** prépare cette commande c'est-à-dire la page Web en question. Le serveur va se charger de traiter la requête et renvoyer les données demandées (page web, image, vidéo...).
3. Et enfin, le **navigateur** interprète les données reçues et les renvoie au client qui va les afficher directement sur notre écran.



Quels sont les meilleurs éditeurs HTML pour 2019 ?

1. Atom
2. NotePad ++
3. Sublime Text
4. Adobe Dreamweaver CC



Le langage HTML5

I. Les balises et leurs attributs :

Il ne suffit pas d'écrire simplement du texte dans l'éditeur, il faut aussi donner des instructions à l'ordinateur. En HTML, on utilise pour cela des balises. Celles-ci sont invisibles à l'écran pour vos visiteurs, mais elles permettent à l'ordinateur de comprendre ce qu'il doit afficher. Elles sont entourées de "chevrons", c'est-à-dire des symboles < et >, comme ceci : <balise>

1. Types de balises

On distingue deux types de balises : les balises en paires et les balises orphelines.

- Les balises en paires

Elles s'ouvrent, contiennent du texte, et se ferment plus loin. Voici à quoi elles ressemblent : **<h1>Ceci est un titre</h1>**

- Les balises orphelines

Ce sont des balises qui servent le plus souvent à insérer un élément à un endroit précis (par exemple une image).

Une balise orpheline s'écrit comme ceci : ****

2. Les attributs

Les attributs sont un peu les options des balises. Ils viennent les compléter pour donner des informations supplémentaires. L'attribut se place après le nom de la balise ouvrante et a le plus souvent une valeur, comme ceci : **<balise attribut = "valeur">**

Prenons la balise **** que nous venons de voir. Seule, elle ne sert pas à grand-chose. On pourrait rajouter un attribut qui indique le nom de l'image à afficher : ****

II. Structure de base d'une page HTML5

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <!-- En-tête de la page -->
5     <meta charset="utf-8" />
6     <title>Titre</title>
7   </head>
8
9   <body>
10    <!-- Corps de la page -->
11  </body>
12 </html>
```

1. Le doctype

Cette ligne du doctype était autrefois incroyablement complexe. Il était impossible de la retenir de tête. Pour XHTML 1.0, il fallait écrire : `<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">`.

Dans le cadre de HTML5, il a été décidé de la simplifier, pour le plus grand bonheur des webmasters. Quand vous voyez une balise doctype courte `<!DOCTYPE html>` cela signifie que la page est écrite en HTML5.

2. La balise `<html>`

C'est la balise principale du code. Elle englobe tout le contenu de votre page. Comme vous pouvez le voir, la balise fermante `</html>` se trouve tout à la fin du code !

3. L'en-tête `<head>` et le corps `<body>`

Une page web est constituée de deux parties :

- L'en-tête : cette section donne quelques informations générales sur la page comme son titre, l'encodage, etc.
- Le corps : c'est là que se trouve la partie principale de la page. Tout ce que nous écrivons ici sera affiché à l'écran.

4. L'encodage (charset)

`<meta charset="utf-8" />`

Cette balise indique l'encodage utilisé dans votre fichier `.html`.

L'encodage indique la façon dont le fichier est enregistré. C'est lui qui détermine comment les caractères spéciaux vont s'afficher (accents, idéogrammes chinois et japonais, caractères arabes, etc.).

5. Le titre principal de la page `<title>`

C'est le titre de votre page, toute page doit avoir un titre qui décrit ce qu'elle contient. Il est conseillé de garder le titre assez court (moins de 100 caractères en général).

6. Les commentaires

`<!-- Ceci est un commentaire -->`

Un commentaire en HTML est un texte qui sert simplement de mémo. Il n'est pas affiché, il n'est pas lu par l'ordinateur, cela ne change rien à l'affichage de la page.

III. Quelques Balises d'organisation du texte

Balise	Rôle
<code><p> Votre paragraphe </p></code>	Délimiter les paragraphes.
<code>
</code>	Sauter une ligne
<code><h1> Votre titre </h1></code>	Crée des titres, on a le droit d'utiliser six niveaux de titres différents de h1 à h6
<code> votre texte </code>	Mettre un peu en valeur (emphase faible) un texte
<code> Votre texte </code>	Mettre bien en valeur (emphase forte) un texte
<code><mark> Votre texte </mark></code>	Permet de faire ressortir visuellement un texte.
<code> ligne 1 ligne N </code>	Créer une liste non ordonnée
<code> ligne 1 ligne N </code>	Créer une liste ordonnée

1. Remarques

- Avec les balises , , <mark> la plupart des navigateurs affichent les textes en italique, en gras et surlignés mais rien ne les y oblige. Et pourtant, ce n'est pas à cela que servent ces balises, Le rôle des balises est d'indiquer le sens du texte.
- De nombreux programmes analysent le code source des pages web, à commencer par les robots de moteurs de recherche, ces robots parcourent le Web en lisant le code HTML de tous les sites. Les mots-clés ont tendance à avoir plus de valeur à leurs yeux, donc si quelqu'un fait une recherche sur ces mots, il a plus de chances de tomber sur votre site.

IV. Créez des liens

Il est facile de reconnaître les liens sur une page : ils sont écrits d'une façon différente (par défaut, en bleu et soulignés). Pour faire un lien, la balise que nous allons utiliser est très simple à retenir :<a>. Il faut cependant lui ajouter un attribut, href, pour indiquer vers quelle page le lien doit conduire.

1. Lien externe :

Le code ci-dessous est un lien qui amène vers OpenClassrooms, situé à l'adresse <https://openclassrooms.com> :

```
<a href = "https://openclassrooms.com">Aller à OpenClassrooms</a>
```

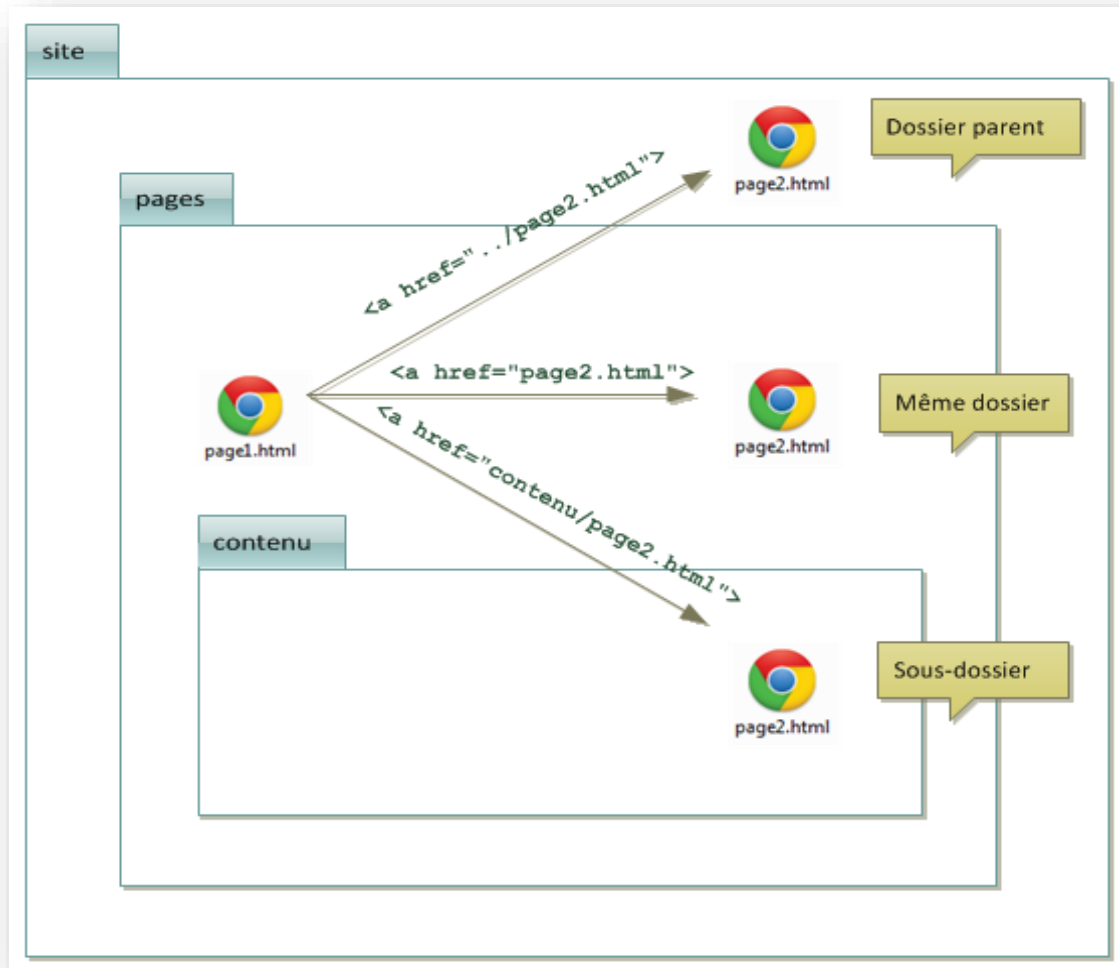
2. Lien interne :

Pour faire un lien de la page 1 vers la page 2 : si les deux fichiers sont situés dans le même dossier, il suffit d'écrire comme cible du lien le nom du fichier vers lequel on veut amener.

```
<a href = "page2.html"> Texte </a>. On dit que c'est un lien relatif.
```

- Les liens relatifs

Il suffit de regarder dans quel niveau de dossier se trouve votre fichier cible pour savoir comment écrire votre lien. La figure suivante fait la synthèse des différents liens relatifs possibles.



- Un lien vers une ancre :

Une ancre est une sorte de point de repère que vous pouvez mettre dans vos pages HTML lorsqu'elles sont très longues, il peut alors être utile de faire un lien amenant plus loin dans la même page. Pour créer une ancre, il suffit de rajouter l'attribut **id** à une balise qui va alors servir de repère. Par exemple : `<h2 id = "mon_ancre"> Titre </h2>`

Ensuite, il suffit de créer un lien comme d'habitude, mais cette fois l'attribut **href** contiendra un dièse (#) suivi du nom de l'ancre. Exemple :

` Aller vers l'ancre `

- Cas pratiques d'utilisation des liens

- Un lien qui affiche une infobulle au survol

Vous pouvez utiliser l'attribut `title` qui affiche une bulle d'aide lorsqu'on pointe sur le lien. Cet attribut est facultatif. La bulle d'aide peut être utile pour informer le visiteur avant même qu'il n'ait cliqué sur le lien.

Par exemple :

```
<a href = "https://openclassrooms.com" title = "Vous ne le regretterez pas !" > OpenClassrooms </a>
```

- Un lien qui ouvre une nouvelle fenêtre

Il est possible de forcer l'ouverture d'un lien dans une nouvelle fenêtre. Pour cela, on rajoutera `target = "_blank"` à la balise `<a>`.

- Un lien pour envoyer un e-mail

Si vous voulez que vos visiteurs puissent vous envoyer un e-mail, vous pouvez utiliser des liens de type **mailto**. Rien ne change au niveau de la balise, vous devez simplement modifier la valeur de l'attribut **href** comme ceci :

```
<a href="mailto:exemple@exemple.com"> Envoyez-moi un e-mail !</a>
```

- Un lien pour télécharger un fichier

Il faut procéder exactement comme si vous faisiez un lien vers une page web, mais en indiquant cette fois le nom du fichier à télécharger. Par exemple : ` Télécharger le fichier `

V. Insérez des images

Il existe plusieurs formats d'images adaptées au Web :

- JPEG : pour les photos
- PNG : pour toutes les autres illustrations
- GIF : similaire au PNG, plus limité en nombre de couleurs mais qui peut être animé.

On insère une image avec la balise ``, C'est une balise de type orpheline (comme `
`). Elle doit obligatoirement comporter au moins ces deux attributs : `src` (nom de l'image) et `alt` (courte description de l'image). ``

Si une image illustre le texte (a un rôle avant tout sémantique et n'est pas seulement décorative), il est conseillé de la placer au sein d'une balise **<figure>**. La balise **<figcaption>** permet d'écrire la légende de l'image.

Exemple :

```
<figure>  
  <img src = "images/ie.png" alt = "Internet Explorer" />  
  <img src = "images/m.png" alt = "Mozilla Firefox" />  
  <img src = "images/chro.png" alt = "Google Chrome" />  
  <figcaption>Logos des navigateurs </figcaption>  
</figure>
```

VI. Insérez une table

Les tables servent à représenter des collections de données.

1. **Table** : Le conteneur global
2. **Thead** : Contient les titres de colonnes. Contient une ligne <tr> et des cellules <td>.
3. **Tbody** : contient les données à proprement parlé.
4. **Ttfoot** : réuni les informations relatives au pied de page.
5. **Tr** : Ligne de table, contenant un ensemble de cellules, soit des balises <td>.
6. **Td** : Cellules de tableau. L'attribut colspan permet d'en regrouper plusieurs horizontalement sans changer l'agencement global de la table. On peut les fusionner verticalement, sur plusieurs lignes avec l'attribut rowspan.

Exemple :

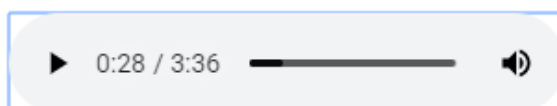
	<td colspan="2"> </td>	
<td rowspan="2"></td>		

VII. Lire un fichier audio

Il était assez compliqué de lire un fichier audio ou mp3 depuis votre site web, le HTML5 a simplifié sa mise en place. A partir d'une simple balise à insérer dans votre page web, la lecture de votre fichier musicale s'effectuera simplement. Pour utiliser ce lecteur HTML5, vous devez utiliser la balise <audio>. Cette balise a été implanté dans le HTML5 afin de faciliter la lecture de fichier musicaux ou sonore.

1. Exemple d'utilisation de la balise <audio>

```
<audio controls="controls">  
  <source src="music.mp3" type="audio/mp3" />  
  Votre navigateur n'est pas compatible  
</audio>
```



2. Attribut spécifique à la balise <audio>

Attribut	Valeur	Description
autoplay	autoplay	Lecture automatique au chargement de la page
controls	controls	Affiche les boutons de contrôle du lecteur
loop	loop	Lecture en boucle
preload	auto metadata none	Définis si le fichier musicale ou sonore est pré-chargé en même temps que la page ou uniquement lorsque l'utilisateur clic sur play
src	url	Définissez l'adresse du fichier musicale ou sonore

VIII. Lire un fichier vidéo

Une des nouveautés majeures de HTML5 et l'une des plus remarquables est la balise **video**, elle offre en HTML5 une solution simple, native pour les navigateurs pour l'intégration d'une vidéo dans une page web.

1. Les attributs de la balise <video>

- L'attribut controls donne accès aux contrôles de lecture (boutons de navigation, volume, etc, selon les possibilités du navigateur), ou les masque s'il est omis.
- L'attribut preload="auto" permet de de spécifier au navigateur de **débuter le téléchargement** de la vidéo tout de suite, en anticipant sur le fait que l'utilisateur lira la vidéo. Attention, cette option est à manier avec prudence (il est préférable que ce soit la seule raison d'être de la page).
- L'attribut autoplay="true" comme son nom l'indique, permet de lancer la **lecture automatiquement**. Cela peut également être problématique avec une connexion à faible bande passante ou sur un terminal mobile. De manière générale, évitez d'imposer vos choix à l'utilisateur... et à sa connexion internet.
- L'attribut poster="image.jpg" permet d'indiquer une **image à afficher par défaut** dans l'espace réservé par la vidéo, avant que la lecture de celle-ci ne soit lancée.
- L'attribut loop indique que la lecture doit s'effectuer **en boucle**.

2. Exemple :

```
<video width="400" height="222" controls="controls">
```

```
<source src="video.mp4" type="video/mp4" />
```

Ici l'alternative à la vidéo

```
</video>
```

IX. Créez des formulaires

Toute page HTML peut être enrichie de formulaires interactifs, qui invitent vos visiteurs à renseigner des informations : saisir du texte, sélectionner des options, valider avec un bouton... tout est possible !

Nous arrivons cependant aux limites du langage HTML car il faut ensuite pouvoir analyser les informations que le visiteur a saisies... et cela ne peut pas se faire en langage HTML. Comme nous allons le voir, le traitement des résultats doit s'effectuer dans un autre langage, par exemple le PHP.

1. La balise <form>

<form> </form> C'est la balise principale du formulaire, elle permet d'en indiquer le début et la fin.

2. Les attributs

- **Method** : cet attribut indique par quel moyen les données vont être envoyées. Il existe deux solutions pour envoyer des données sur le Web :
 - `method="GET"` : c'est une méthode en général assez peu adaptée car elle est limitée à 255 caractères. La particularité vient du fait que les informations seront envoyées dans l'adresse de la page (`http://...`), mais ce détail ne nous intéresse pas vraiment pour le moment. La plupart du temps, je vous recommande d'utiliser l'autre méthode `post`.
 - `method="POST"` : c'est la méthode la plus utilisée pour les formulaires car elle permet d'envoyer un grand nombre d'informations. Les données saisies dans le formulaire ne transitent pas par la barre d'adresse.
- **action** : c'est l'adresse de la page ou du programme qui va *traiter* les informations. Cette page se chargera de vous envoyer un e-mail avec le message si c'est ce que vous voulez, ou bien d'enregistrer le message avec tous les autres dans une base de données. Cela ne peut pas se faire en HTML et CSS, on utilisera en général un autre langage.
- **Exemple** :

```
<form method="post" action="traitement.php">
```

```
</form>
```

3. Zone de texte mono-ligne

```
<form method="GET" action="traitement.php">
```

```
  Votre pseudo : </label><input type="text" name="pseudo" />
```

```
</form>
```

Votre pseudo :

- **Quelques attributs supplémentaires** :
 - On peut agrandir le champ avec **size**.
 - On peut limiter le nombre de caractères que l'on peut saisir avec **maxlength**.
 - On peut pré-remplir le champ avec une valeur par défaut à l'aide de **value**.

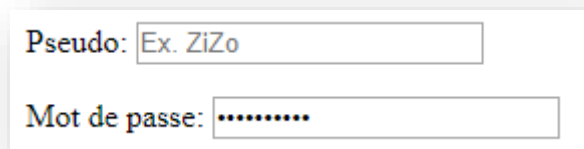
- On peut donner une indication sur le contenu du champ avec **placeholder**. Cette indication disparaîtra dès que le visiteur aura cliqué à l'intérieur du champ.

4. Zone mot de passe

Vous pouvez facilement faire en sorte que la zone de texte se comporte comme une « zone de mot de passe », c'est-à-dire une zone où on ne voit pas à l'écran les caractères saisis. Pour créer ce type de zone de saisie, utilisez l'attribut `type="password"`.

- **Exemple :**

```
<p>Pseudo: <input type="text"
name="pseudo"placeholder="Ex. ZiZo"/></p>
<p>Mot de passe: <input type="password" name="pw" /></p>
```



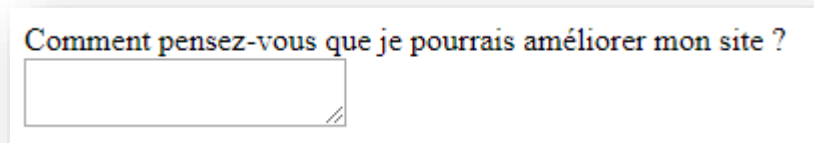
The image shows a screenshot of a web form. It contains two input fields. The first is a text input field with the label 'Pseudo:' and a placeholder text 'Ex. ZiZo'. The second is a password input field with the label 'Mot de passe:' and masked characters represented by dots.

5. Zone de texte multiligne

Pour créer une zone de texte multiligne, on change de balise : nous allons utiliser `<textarea> </textarea>`.

- **Exemple :**

```
<label>Comment pensez-vous que je pourrais améliorer
mon site ?</label><br /><textarea
name="ameliorer"></textarea>
```



The image shows a screenshot of a web form. It contains a multi-line text input field with the label 'Comment pensez-vous que je pourrais améliorer mon site ?'. The input field is empty and has a small cursor icon at the bottom right corner.

- **Attributs :**

On peut ajouter les attributs `rows` et `cols` à la balise `<textarea>`. Le premier indique le nombre de lignes de texte qui peuvent être affichées simultanément, et le second le nombre de colonnes.

6. Les zones de saisie enrichies

HTML5 apporte de nombreuses fonctionnalités nouvelles relatives aux formulaires. De nouveaux types de champs sont en effet apparus avec cette version. Il suffit de donner à l'attribut `type` de la balise `<input />` l'une des nouvelles valeurs disponibles.

- **email**
- **url**
- **tel**
- **number**

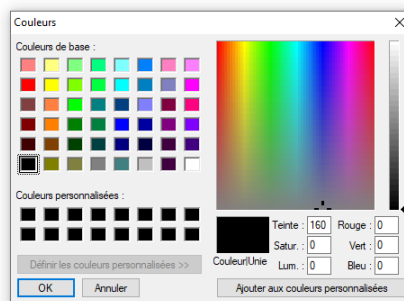
Vous pouvez personnaliser le fonctionnement du champ avec les attributs suivants :

- min: valeur minimale autorisée.
- max: valeur maximale autorisée.
- step: c'est le « pas » de déplacement

- **range**



- **color**



- **date**

Différents types de champs de sélection de date existent :

- date : pour la date (05/08/1985 par exemple) ;
- time : pour l'heure (13:37 par exemple) ;
- week : pour la semaine ;
- month : pour le mois ;
- datetime-local : pour la date et l'heure.

7. Les éléments d'options

HTML vous offre une ribambelle d'éléments d'options à utiliser dans votre formulaire. Ce sont des éléments qui demandent au visiteur de faire un choix parmi une liste de possibilités. Nous allons passer en revue :

- **Les cases à cocher**

```
<form method="post" action="traitement.php">
  <p>
    Cochez les aliments que vous aimez manger : <br />
    <input type="checkbox" name="choix"/> <label>Frites</label><br />
    <input type="checkbox" name="choix" /> <label>Steak haché</label><br />
    <input type="checkbox" name="choix"/> <label>Epinards</label><br />
    <input type="checkbox" name="choix"/> <label>Huitres</label>
  </p>
</form>
```

On peut utiliser l'attribut checked pour sélectionner une valeur par défaut.

Cochez les aliments que vous aimez manger :

- Frites
- Steak haché
- Epinards
- Huitres

- **Les zones d'options**

```
<form method="post" action="traitement.php">
  <p>
    Veuillez indiquer la tranche d'âge dans laquelle vous vous situez :<br />
    <input type="radio" name="age" /> <label>Moins de 15 ans</label><br />
    <input type="radio" name="age" /> <label>15-25 ans</label><br />
    <input type="radio" name="age" /> <label>25-40 ans</label><br />
    <input type="radio" name="age" /> <label>Encore plus vieux que ça ?!</label>
  </p>
</form>
```

L'attribut checked est, là aussi, disponible pour sélectionner une valeur par défaut.

Veuillez indiquer la tranche d'âge dans laquelle vous vous situez :

- Moins de 15 ans
- 15-25 ans
- 25-40 ans
- Encore plus vieux que ça ?!

- **Les listes déroulantes**

Les listes déroulantes sont un autre moyen élégant de faire un choix parmi plusieurs possibilités. Le fonctionnement est un peu différent. On va utiliser la balise <select> </select> qui indique le début et la fin de la liste déroulante. On ajoute l'attribut name à la balise pour donner un nom à la liste.

Puis, à l'intérieur du <select> </select>, nous allons placer plusieurs balises <option> </option> (une par choix possible). On ajoute à chacune d'elles un attribut value pour pouvoir identifier ce que le visiteur a choisi.

Voici un exemple d'utilisation :

```
<label>Dans quel pays habitez-vous ?</label><br />
<select name="pays">
  <option value="france">France</option>
  <option value="espagne" selected>Espagne</option>
  <option value="italie">Italie</option>
  <option value="royaume-uni">Royaume-Uni</option>
  <option value="canada">Canada</option>
  <option value="etats-unis">États-Unis</option>
  <option value="chine">Chine</option>
  <option value="japon">Japon</option>
</select>
```

Dans quel pays habitez-vous ?



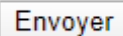
Espagne
France
Espagne
Italie
Royaume-Uni
Canada
Etats-Unis
Chine
Japon

8. Le bouton d'envoi

Il ne nous reste plus qu'à créer le bouton d'envoi. Là encore, la balise `<input />` vient à notre secours. Elle existe en quatre versions :

- `type="submit"` : le principal bouton d'envoi de formulaire. C'est celui que vous utiliserez le plus souvent. Le visiteur sera conduit à la page indiquée dans l'attribut `action` du formulaire.
- `type="reset"` : remise à zéro du formulaire.
- `type="image"` : équivalent du bouton `submit`, présenté cette fois sous forme d'image. Rajoutez l'attribut `src` pour indiquer l'URL de l'image.
- `type="button"` : bouton générique, qui n'aura (par défaut) aucun effet. En général, ce bouton est géré en JavaScript pour exécuter des actions sur la page. Nous ne l'utiliserons pas ici.

On peut changer le texte affiché à l'intérieur des boutons avec l'attribut `value`. Pour créer un bouton d'envoi on écrira donc par exemple : `<input type="submit" value="Envoyer" />`



Envoyer

Lorsque vous cliquez sur le bouton « Envoyer », le formulaire vous amène alors à la page indiquée dans l'attribut `action`.