

HTML

APPRENDRE À CODER EN HTML

Marine ABADI / UTOPIOS / M2I Formation



Notions de bases du HTML

Les balises HTML

A quoi ça sert ?

- Ce sont des **éléments** qui permettent d'insérer du contenu sur notre page : un paragraphe, un titre, insérer une image, une vidéo...

- Les **balises** ouvrantes et fermantes
< p > J'apprends à coder </ p >

- les **balises** orphelines
*< p > J'apprends à coder </ p >
< p > du HTML </ p >*

- Exemple de balises : *< h1 >, < h2 >, < h3 >, < h4 >, < h5 >, < h6 >, < p >, < a >, < img >, < div >, < span >, < ul >, < ol > , < li >*

```
<h1>PREMIERE SEMAINE</h1>
<p>Voici les cours :
  <ol>
    <li>Algorithme et programmation structurée</li>
    <li><em>HTML, HyperText <br> Markup Language </em></li>
    <li>CSS</li>
  </ol>
</p>
```

Les attributs

A quoi ça sert ?

- Ils sont **obligatoires ou facultatifs**
- Ils viennent **compléter les éléments** en les définissant plus précisément

- Un **attribut** contient toujours une valeur

`UTOPIOS`

- Exemple d'attribut : *href, src, alt*

```
RE SEMAINE</H1>
ci les cours :
<ol>
    <li>Algorithme et programmation structurée</li>
    <li><em>HTML, HyperText <br> Markup Language </em></li>
    <li>CSS</li>
</ol>
<a href="http://utopios.solutions">Mon site UTOPIOS</a>

```

La structure d'une page HTML

Comment s'y prendre ?

- ➔ Une **page HTML** doit être **valide**, elle doit comporter certains éléments dans un ordre précis
- ➔ Le **DOCTYPE**
- ➔ **<html>**
- ➔ **<head>, <title> et <meta>**
- ➔ **<body>**
- ➔ Il faut respecter l'**imbrication**

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>MCPD .NET</title>
5          <meta charset="utf-8" />
6      </head>
7      <body>
8
9      </body>
10     </html>
```

C'est parti pour
coder...A complex network graph composed of numerous small, semi-transparent blue dots connected by thin, light blue lines, forming a dense web of triangles and polygons. This pattern repeats across the entire slide, creating a sense of digital connectivity and complexity.

Créer une page HTML

■ Les étapes

- ➔ Créer un dossier sur l'ordinateur
- ➔ Ouverture et enregistrement d'un fichier .html
- ➔ L'auto-complétion
- ➔ Les raccourcis *Utiliser « !tab » pour créer la structure de page*
- ➔ Ouvrir le fichier HTML avec un navigateur

Indentation et commentaire

Comment mais surtout pourquoi ?

- Indenter correspond à créer des retraits en début de ligne dans votre éditeur de façon cohérente et logique. *Ctrl+K+F en sélectionnant la partie de code OU Shift+Alt+F*
- Pour un code **plus propre et plus lisible**
- Les **commentaires** sont des lignes de texte écrite au milieu du code pour **donner des indications** sur ce que fait le code *Ctrl + K + C / Ctrl + K + U*
- **Attention** : ne rien mettre de confidentiel dans les commentaires.
- Le code est **visible par tous** !

Titres / paragraphes / Espace / Retour à la ligne

■ Hiérarchiser le contenu d'une page

- ➔ Il existe **six niveaux hiérarchiques** de titres (heading) définis par les éléments : `h1, h2, h3, h4, h5, h6`.
- ➔ Attention ils n'ont rien avoir avec la balise `<title>`
- ➔ Pour les paragraphes, on utilise `<p> Voici mon paragraphe </p>`
- ➔ Les espaces dans le code ne sont pas interprétés par le navigateur
- ➔ On utilise `
` comme break ou `<hr>` (horizontal rule)
- ➔ Plusieurs solutions `<pre>`

Entités HTML & caractères réservés

■ Les entités HTML

→ Une **entité HTML** est une suite de caractère qui est utilisée pour afficher un caractère réservé ou un **caractère invisible** (comme un espace) en HTML.

→ Un **caractère réservé** est un caractère qui possède déjà une signification particulière en HTML.

Par exemple le caractère <

- L'entité HTML **&nbsp** (« non breaking space ») simple espace
- L'entité HTML **&ensp** (« en space ») espace double ;
- L'entité HTML **&emsp** (« em space ») espace quadruple ;
- L'entité HTML **&thinsp** (« thin space ») demi-espace;
- L'entité HTML **&lt** < (chevron ouvrant)
- L'entité HTML **&gt** > (chevron fermant)
- L'entité HTML **&amp** & (esperluette)

```
→  
<h1>Mon titre  
principal</h1>  
<h2>Je suis      un      titre  
important</h2>  
  
<p>abcdef</p>  
<p>a b c d e f</p>  
<p>a&thinsp;b&thinsp;c&thinsp;d&thinsp;e&thinsp;f</p>  
  
<p>a&ampnbspb&ampnbspc&ampnbspd&ampnbspe&ampnbspf</p>  
<p>a&ampenspb&ampenspc&ampenspd&ampenspe&ampenspf</p>  
<p>a&ampemspb&ampemspc&ampemspd&ampemspe&ampemspf</p>
```

Mon titre principal

Je suis un titre important

abcdef

a b c d e f

a b c d e f

a b c d e f

a b c d e f

Importance des textes

■ Les éléments à utiliser

- ➔ L'élément ``
- ➔ L'élément `` (pour emphasis ; « emphase » ou « accentuation » en français)
- ➔ L'élément `<mark>`

Sémantique et structure de page

■ Les éléments à utiliser

→ Le HTML5 intègre des éléments dit « structurants » qui permettent de préciser le sens de certains contenus. Google par ex. peut maintenant comprendre les rôles et l'importance de chaque élément de la page.

Voici la manière dont on va structurer une page :

- header : en-tête ou le haut d'une page
- nav : menu de navigation dans une page
- main : contenu principal de l'élément body
- section : section dans une page, groupement thématique
- article : partie de page comme un post sur un blog par exemple
- aside : partie de contenu non lié au reste
- footer : pied d'une page

Sémantique et structure de page

■ Imbrication simple

<header>

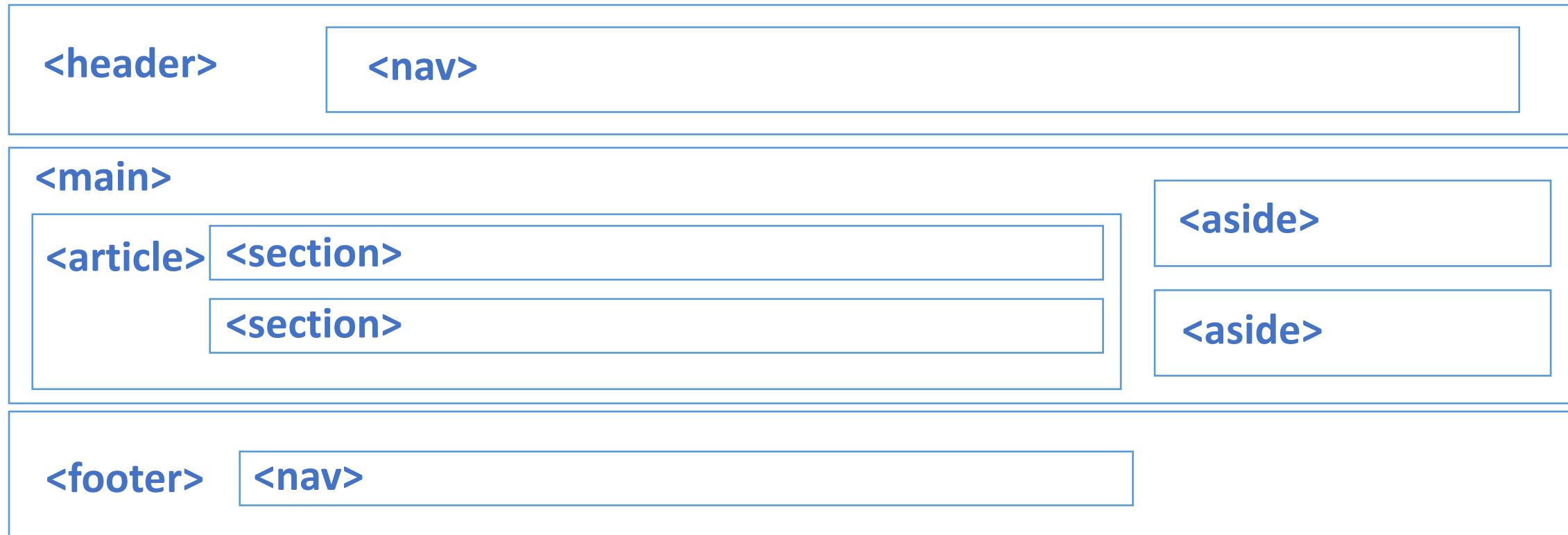
<nav>

<main>

<footer>

Sémantique et structure de page

■ Imbrication complexe



Sémantique et structure de page

■ Imbrication complexe



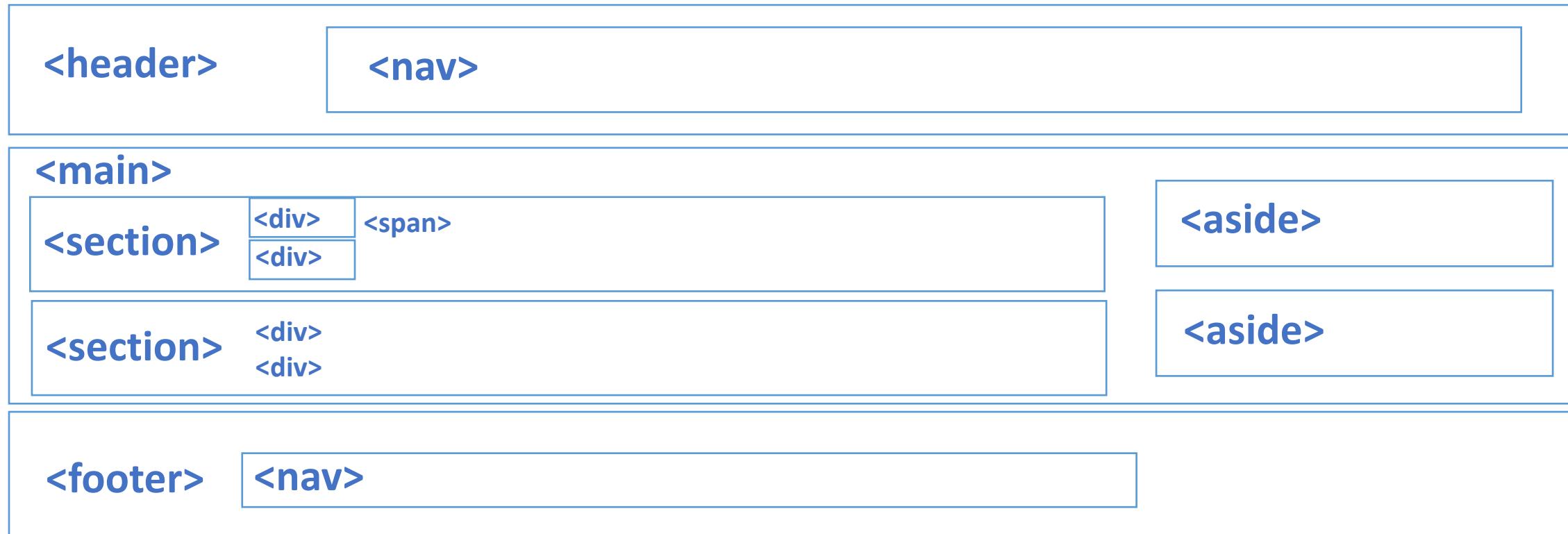
Balises non sémantiques

■ Div et Span

- ➡ Ces deux éléments sont en effet des **conteneurs génériques** qui ont été créés pour nous permettre d'ordonner nos pages plus simplement ensuite en CSS.
- ➡ On utiliser l'élément **div** comme conteneur pour différents éléments afin de pouvoir ensuite facilement appliquer les mêmes styles CSS
- ➡ L'élément **span** sert de conteneur à un autre niveau : il va servir de conteneur interne à un élément plutôt que de conteneur pour plusieurs éléments.
- ➡ Lien intéressant : liste des balises => <https://jaetheme.com/balises-html5/>

Sémantique et structure de page

■ Imbrication complexe avec Div et Span



Liste HTML

■ Les types de liste

 Les **listes non-ordonnées** vont être utiles pour lister des éléments sans hiérarchie ni ordre logique.

```
<h1>Les listes</h1>
<!--Un exemple de liste non-ordonnée-->
<ul>
    <li>Bureau</li>
    <li>Manger</li>
    <li>Se promener</li>
</ul>
```

Liste HTML

■ Les types de liste

- ➡ Les **listes ordonnées** seront utiles lorsqu'il y aura une notion d'ordre ou de progression logique ou encore de hiérarchie entre les éléments de notre liste.

```
<h1>Les listes</h1>
<!--Deux exemples de listes ordonnées-->
<p>Liste n°1 :</p>
<ol>
    <li>Entrée</li>
    <li>Plat</li>
    <li>Dessert</li>
</ol>
```

Liens HTML

A quoi servent les liens ?

- ➔ Les **liens** en HTML vont nous servir à créer des ponts entre différentes pages d'un même site ou de sites différents.
- ➔ En cliquant sur une **ancre** nos utilisateurs vont être redirigés vers une page cible.
- ➔ Il existe **deux types de liens** hypertextes :
 - Les **liens internes** qui vont servir à naviguer d'une page à l'autre dans un même site
 - Les **liens externes** qui vont envoyer les utilisateurs vers une page d'un autre site

Liens HTML

Comment créer un lien ?

→ L'élément HTML `a` accompagné de son attribut `href` pour créer un **lien** en HTML

→ **Créer un lien externe**

<p>Pour créer des liens en HTML, on utilise l'élément `a` et son attribut `href`.</p>

<p>Ex : lien vers UTOPIOS</p>



→ **Créer un lien interne**

- les deux pages (source et destination) se trouvent dans un même dossier `href=< nom_page.html >`
- la page de destination se trouve dans un sous dossier par rapport à la page source `href=<dossier/nom_page.html >`
- la page de destination se trouve dans un dossier parent par rapport à la page source `href=<.../nom_page.html >`

Liens HTML

L'attribut target

- On utilise l'attribut target pour que le lien s'ouvre dans un nouvel onglet afin que le visiteur ne perde pas la page de votre site.
- Souvent on utilise la valeur _blank qui spécifie que la nouvelle page doit s'ouvrir dans un nouvel onglet.

<p>Un lien vers la page d'accueil de <a href="https://utopios.solutions/" target="_blank

Envoyer un mail

■ Ouvrir la messagerie par défaut

- Pour permettre l'envoi d'un mail en HTML, on va indiquer en valeur de l'attribut href de notre élément a la valeur mailto : suivie de notre adresse email.

```
<h1>Envoyer un mail</h1>
<p>M'envoyer un <a href="mailto:pierre.giraud@edhec.com">mail</a></p>
```

Télécharger un fichier

■ Permettre le téléchargement d'un fichier PDF

- ➡ Pour la plupart des fichiers, il va falloir indiquer leur adresse (relative ou leur URL complète) en valeur de l'attribut `href`
- ➡ On va également pouvoir « forcer » le téléchargement d'un fichier en ajoutant un attribut `download` dans l'élément `a` tout en indiquant l'adresse du fichier en question en valeur de l'attribut `href`

Créer des tableaux

■ Les éléments de base

- ➔ Un tableau est un ensemble de lignes et de colonnes. L'intersection entre une ligne et une colonne est une cellule de tableau. On ne les utilise pas pour faire de la mise en page
- ➔ Un élément `table`
- ➔ Un élément `th` pour l'entête
- ➔ Des éléments `tr` « table row » qui vont nous permettre d'ajouter des lignes
- ➔ Des éléments `td` « table data » qui vont nous permettre d'ajouter des cellules dans nos lignes et donc de créer de nouvelles colonnes.

Créer des tableaux

■ Structurer les tableaux

- ➡ Lorsque les tableaux sont plus complexes, on va utiliser les éléments spécifiques :
 - ➡ thead pour la tête du tableau
 - ➡ tbody pour le corps du tableau
 - ➡ tfoot pour le pied du tableau.

Insérer des médias

■ Insérer des images

- ➔ L'insertion d'images en HTML va se faire au moyen de l'élément HTML `img`.
- ➔ C'est une balise orpheline
- ➔ On va préciser deux attributs : les attributs `src (source)` et `alt`
- ➔ Les éléments `figure` et `figcaption` vont marquer sémantiquement du contenu média comme des images, de l'audio ou des vidéos.

```
<figure>

<figcaption>C'est le logo de M2I</figcaption>
</figure>
```

Insérer des médias

■ Insérer de l'audio

- On peut utiliser un élément **audio** avec deux attributs **src** et **controls** (bouton de lecture etc...)
- Mais pour mettre différentes versions, on utilise l'élément **audio** et l'élément **source** avec
 - un attribut **src**
 - un attribut **type** permet d'indiquer au navigateur le type de codec audio utilisé
- Les autres attributs
 - L'attribut **preload** permet d'indiquer au navigateur si le fichier doit être préchargé
 - L'attribut **autoplay** va nous permettre de lancer automatiquement la lecture du fichier audio
 - L'attribut **loop** permet de lire le fichier audio en boucle
 - L'attribut **muted** sert à définir si le son doit être initialement coupé

Insérer des médias

■ Insérer des vidéos

- On utilise un élément **video** avec deux attributs **src** et **controls**
- Ou bien l'élément **vidéo** puis l'élément **source**, nous allons devoir inclure aussi :
 - Un attribut **src**
 - Un attribut **type** permet d'indiquer au navigateur le type de codec audio utilisé
- Les autres attributs
 - Un attribut **preload** permet d'indiquer au navigateur si le fichier doit être préchargé
 - Un attribut **autoplay** va nous permettre de lancer automatiquement la lecture du fichier audio dès qu'il sera chargé.
 - Un attribut **loop** permet de lire le fichier audio en boucle
 - Un attribut **muted** sert à définir si le son doit être initialement coupé.
 - Un attribut **poster** va nous permettre de renseigner l'adresse d'une image qui devra être utilisée comme image d'illustration de fond

Insérer un fichier web dans un autre

■ Insérer des iframe

- ➔ L'élément **iframe** permet d'intégrer un fichier web entier dans un autre.
- ➔ On l'utilise pour intégrer des vidéos YouTube ou des cartes Google Maps.
- ➔ Atribut **src** : Cet attribut va nous permettre d'indiquer l'adresse du document à intégrer`
- ➔ Atribut **width** et **height** : permet de définir les dimensions de notre élément iframe
- ➔ **allow** : permet de choisir quelles fonctionnalités de l'iframe activer ou désactiver. Cet attribut est utilisé pour renforcer la sécurité du code
- ➔ **sandbox** : permet de limiter les permissions de notre iframe c'est-à-dire de limiter ses possibilités. Cet attribut est utilisé pour renforcer la sécurité de notre code.

Les formulaires

Insérer un formulaire

- Les formulaires HTML vont permettre aux visiteurs d'envoyer des données que nous pourrons manipuler et ou stocker.
- Il faudra un langage PHP pour manipuler les données et MYSQL pour les stocker
- On utilise l'élément HTML **form** avec l'attribut **method** (valeur : get et post).
- Puis l'attribut **action** : sert à préciser l'adresse de la page qui va traiter les données.

```
<form method="post" action="form.php">
  <div class="champ">
    <label for="pseudo">Entrez un pseudonyme : </label>
    <input type="text" id="pseudo" name="pseudo">
  </div>
  <div class="champ">
    <label for="pass">Entrez un mot de passe :</label>
    <input type="password" id="pass" name="pass">
  </div>
  <div class="champ">
    <label for="mail">Entrez un mail :</label>
    <input type="email" id="mail" name="mail">
  </div>
  <div>
    <input type="submit" value="Envoyer">
  </div>
</form>
```

Les formulaires

Insérer un formulaire

- Pour créer ces champs, nous allons utiliser des éléments **input** et son attribut **type** (**text**, **email**, **date**)
- Pour donner des indications sur les données attendues, on utilise des éléments **label**. On fait une liaison entre son **label** et son **input** (avec un **id unique**)
- Pour créer un bouton d'envoi, il faut utiliser un élément **input** mais cette fois-ci de type **input type="submit"**.

```
<form method="post" action="form.php">
  <div class="champ">
    <label for="pseudo">Entrez un pseudonyme : </label>
    <input type="text" id="pseudo" name="pseudo">
  </div>
  <div class="champ">
    <label for="pass">Entrez un mot de passe :</label>
    <input type="password" id="pass" name="pass">
  </div>
  <div class="champ">
    <label for="mail">Entrez un mail :</label>
    <input type="email" id="mail" name="mail">
  </div>
  <div>
    <input type="submit" value="Envoyer">
  </div>
</form>
```

Les formulaires

Eléments d'un formulaire

Elément	Définition
form	Définit un formulaire
input	Définit un champ de données pour l'utilisateur
label	Définit une légende pour un élément input
textarea	Définit un champ de texte long
select	Définit une liste de choix
optgroup	Définit un groupe d'options dans une liste
option	Définit une option dans une liste
fieldset	Permet de regrouper les éléments d'un formulaire en différentes parties
legend	Ajoute une légende à un élément fieldset

Les formulaires

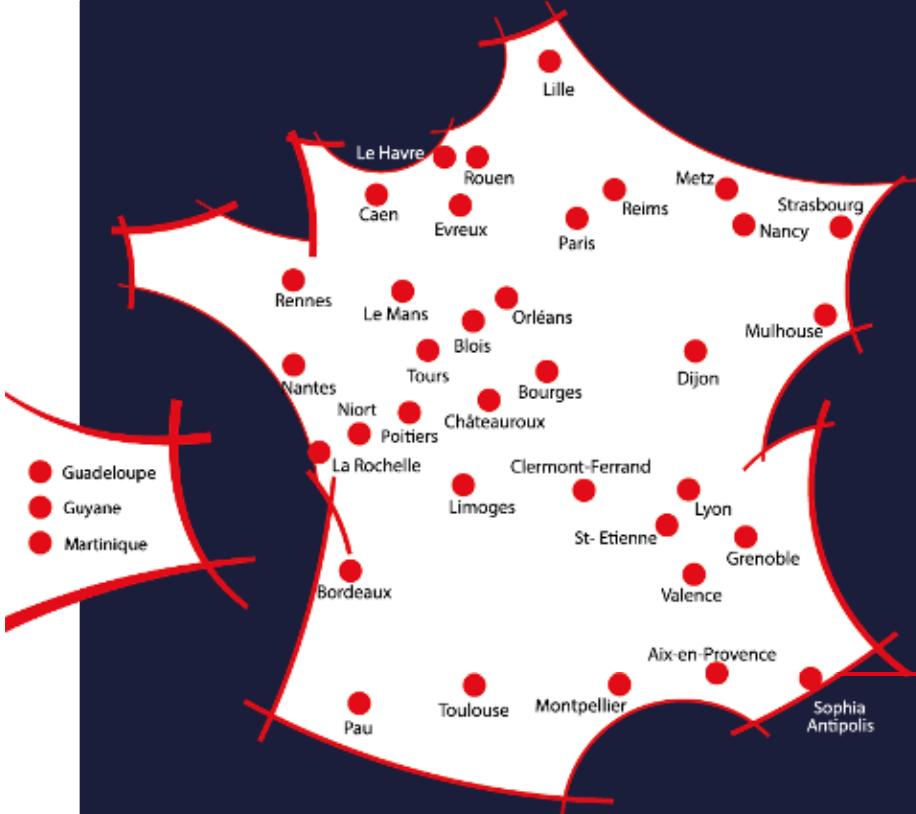
Type d'input

Type d'input	Définition
text	Définit un champ de saisie monoligne qui accepte du texte
number	Définit un champ qui accepte un nombre décimal
email	Crée un champ qui permet de renseigner une adresse mail
date	Permet à l'utilisateur d'envoyer une date
password	Créer un champ de saisie monoligne dont la valeur va être cachée
checkbox	Permet de créer une case à cocher. L'utilisateur peut cocher une ou plusieurs cases d'un coup
radio	Permet de créer un bouton radio. Par définition, un seul bouton radio peut être coché dans un ensemble
url	Crée un champ qui accepte une URL
file	Permet à un utilisateur de télécharger un fichier
submit	Crée un bouton d'envoi des données du formulaire

Ressources intéressantes

■ Liens pratiques

- ➔ Pour vérifier la validité d'un code HTML, le w3c (World Wide Web Consortium) met à disposition
<http://validator.w3.org>
- ➔ Pour accéder à de la documentation (toutes les balises, attributs etc...)
<https://developer.mozilla.org/fr/docs/Web/HTML>



Découvrez également
l'ensemble des stages à votre disposition
sur notre site m2iformation.fr

m2iformation.fr

