

HTML5 : bases

- Rappel : structure d'un document HTML5
- Balises d'entête
- Balises de texte
- Balises génériques et attributs globaux
- Balises de structuration du document
- Balises de lien (URL, HTTP, arborescence,...)
- Balises de liste
- Balises pour le multimédia (image, son, vidéo,...)
- Balises pour les tableaux

Structure d'un document HTML

- Le document commence par le « **doctype** » qui indique quelle version de HTML est utilisée. En HTML5, il a été simplifié : **<!doctype html>**
- Il y a ensuite les balises **<html>** et **</html>** qui encadrent l'ensemble du document
- Le document est composé de 2 parties :
 - **L'entête** délimité par **<head>** et **</head>** avec les propriétés globales du document
 - **Le corps** délimité par **<body>** et **</body>** avec le contenu du document qui doit s'afficher

Une page HTML de base

```
<!doctype html>
<html>
  <head>
    <title>Titre du document</title>
    <meta charset="utf-8" />
  </head>
  <body>
    <h1>Titre principal de la page</h1>
    <p>Ceci est un paragraphe</p>
  </body>
</html>
```

Balises d'entête

- L'entête **<head>** peut contenir un certain nombre de balises, seule **<title>** est obligatoire
- **<title>** contient le titre du document et apparaît dans l'onglet ou le titre de fenêtre du navigateur
- **<meta>** contient des informations sur le document qui ne sont pas affichées à l'écran
- **<style>** et **<script>** contiennent respectivement le code CSS et le code JavaScript pour la page
- **<link>** sert de lien avec une ressource externe

Balises de texte

- On peut organiser son texte, définir des titres et des paragraphes, passer une ligne, mettre en valeur des passages,...
- Pour les **titres**, il y a 6 niveaux d'importance : **<h1>**, **<h2>**, **<h3>**, **<h4>**, **<h5>** et **<h6>**
- Pour les **paragraphes**, il y a **<p>**
- On peut avoir plusieurs titres et paragraphes dans une page, pour les distinguer, on utilise des **attributs**, soit **id** (valeur unique) soit **class**

Balises de texte

- `
` réalise un saut de ligne
- `` permet de mettre en valeur une portion de texte (affiché en italique, par défaut)
- `` permet de mettre en valeur une portion de texte (affiché en gras, par défaut)
- `<hr />` crée une séparation horizontale
- `<pre>` affiche le texte préformaté (dans le code)
- `<sub>` et `<sup>` affichent en indice ou exposant
- `<address>` contient des informations de contact

Balises de texte

- **<blockquote>** contient un bloc de citation
- **<q>** est plutôt utilisé pour les courtes citations
- **<abbr>** décrit des abréviations et acronymes
- **<mark>** écrit le texte comme s'il était surligné
- **<ins>** écrit le texte comme s'il était inséré
- **** écrit le texte comme s'il était supprimé
- Il en existe encore d'autres : **<small>**, ****, **<i>**, **<dfn>**, **<code>**, **<var>**, **<kbd>**, **<samp>**, **<time>**, **<wbr>**, **<s>**, **<ruby>**, **<rt>**, **<rp>**, **<bdo>**, **<bdi>**,...

Balises génériques

- Il existe deux **balises génériques** qu'on peut utiliser quand aucune autre balise ne convient au niveau de la sémantique.
 - **<div>** : balise de type «block» (utilisée pour créer un regroupement d'éléments ou une division)
 - **** : balise de type «inline» (portion de texte)
- Les balises `<div>` sont souvent utilisées pour effectuer la mise en page du document et diviser en zones. Des balises de structuration ont été ajoutées en HTML5 (voir plus loin)

Attributs globaux

- En général, les **attributs** sont spécifiques à une ou plusieurs balises et permettent de décrire les **propriétés** de l'élément (taille, références,...)
- Mais certains attributs, les **attributs globaux**, peuvent se retrouver dans toutes les balises : **class, id, title, style,...**
- Pour les éléments génériques comme `<div>` ou ``, vu qu'ils n'ont pas de sémantique, il faut employer ces attributs pour les distinguer

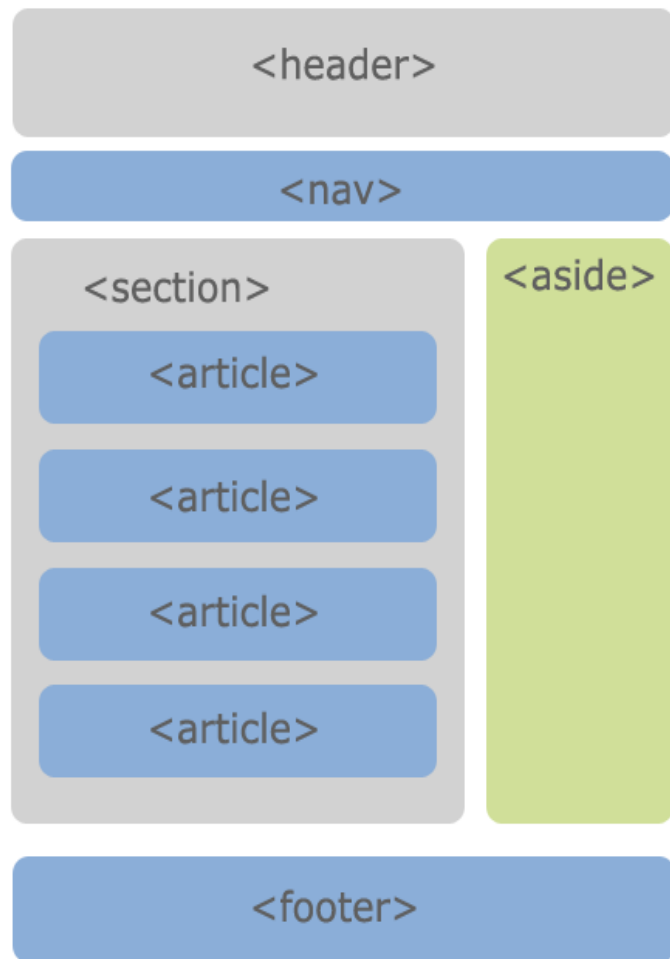
Block ou inline

- Avec les balises de type « **block** », les éléments prennent toute la **largeur de la page** et se disposent les uns **au-dessus** des autres. On peut les dimensionner.
Par exemple : <p>, <h1> à <h6>, <div>, ,...
- Avec les balises de type « **inline** », les éléments prennent la **largeur de leur contenu** et se disposent les uns **à côté** des autres. On ne peut pas les dimensionner.
Par exemple : <a>, , , ,...

Balises de structuration

- Avec HTML5, il y a des balises pour délimiter les zones d'un document et remplacer `<div>`
 - `<header>` : introduction, entête
 - `<nav>` : liens de navigation
 - `<section>` : regroupement de contenu thématique
 - `<article>` : contenu dans un document
 - `<aside>` : annexe au contenu principal
 - `<footer>` : pied de page

Mise en page de base



```
<!doctype html>
<html>
<head>
  <title>titre </title>
  <meta charset="utf-8">
</head>
<body>
<header>
  <nav></nav>
</header>
<section>
  <article></article>
  <article></article>
  <article></article>
  <article></article>
</section>
<aside></aside>
<footer></footer>
</body>
</html>
```

NB : la mise en page et le positionnement des éléments doit être réglé avec CSS

Balises de lien

- Sur Internet, pour passer d'un document à l'autre, on clique sur des **liens hypertexte**
- La balise **<a>** permet de définir un **hyperlien**
- Des attributs sont nécessaires pour préciser :
 - **href** : définit la cible du lien (une **URL**)
 - **target** : définit le mode d'affichage de la cible
 - D'autres attributs sont disponibles (hreflang, media, rel, type, download)

Types de lien

- Pour atteindre une autre page ou une ressource, on doit fournir **l'adresse ou URL**
- On distingue **lien absolu** et **lien relatif** :
 - **absolu** : on donne l'adresse complète (page hébergée sur un site dans un autre domaine)
 - **relatif** : on donne l'adresse par rapport à la page en cours (page ou ressource hébergée sur le même site dans un autre répertoire)
- Si le lien pointe vers une ressource extérieure, on précise aussi le **protocole** (HTTP, FTP,...)

URL ou adresse

- Pour accéder à une ressource (texte, image, son, vidéo, etc) on utilise l'**URL** (Unique Resource Locator) ou **URI** (Uniform Resource Identifier)
- L'URL contient **le protocole**, **le nom du serveur**, **le chemin** (arborescence des dossiers/répertoires) **vers la ressource**, **le port** (si différent de la valeur par défaut 80) et **le nom du fichier**
- Par exemple :

<http://www.cf2m.be/formations/web/description.html:8080>

Protocole HTTP

- **HTTP** (HyperText Transfer Protocol) est un **protocole** de communication utilisé pour envoyer des requêtes à un serveur et recevoir des pages web. Il fonctionne grâce à TCP/IP
- Il existe d'autres protocoles utilisés sur Internet :
 - **HTTPS** pour un transfert HTTP sécurisé
 - **FTP** pour le transfert de fichiers
 - **SMTP**, **POP3** et **IMAP** pour l'e-mail
- Dans l'URL, on indique : **http://www.monsite.be**

Exemples de lien

- **Lien relatif** vers une autre page de son site (la page *contact.html* est dans le **même répertoire** que la page actuellement visitée)

```
<a href="contact.html">Contactez-nous</a>
```

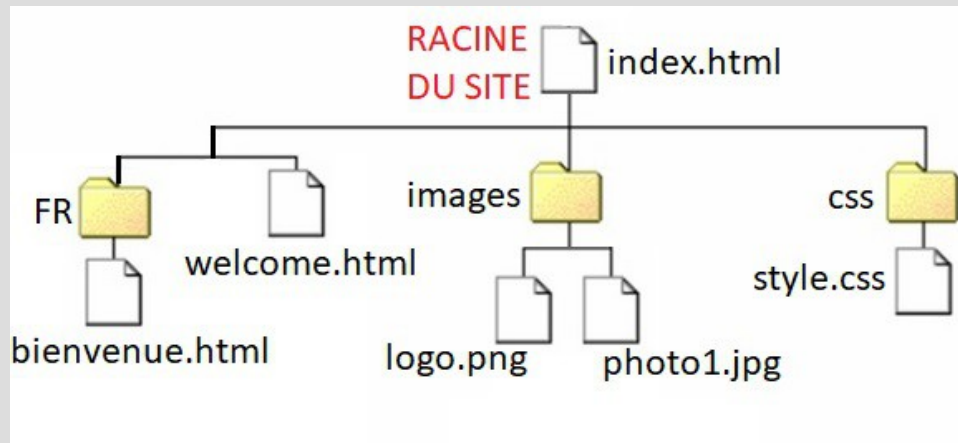
- **Lien absolu** vers un autre site

```
<a href="http://www.cf2m.be">CF2m</a>
```

- Lien pour une **adresse e-mail**

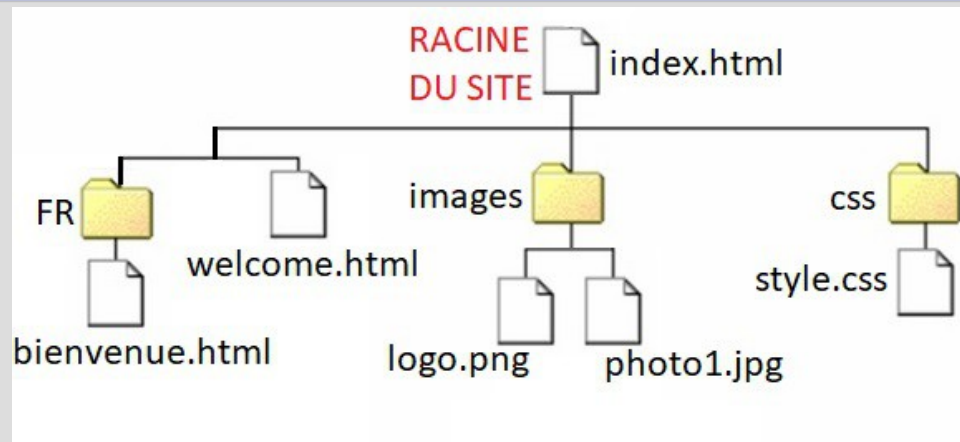
```
<a href="mailto:john@site.com">Ecrivez-moi</a>
```

Répertoires et fichiers



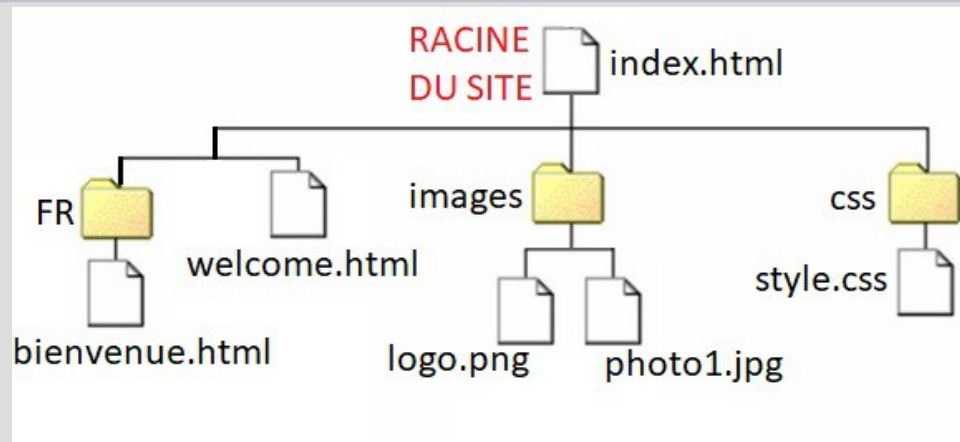
- Un **site Web** est constitué de plusieurs **fichiers** organisés dans des **répertoires**
- Selon **l'arborescence**, la manière d'écrire les URL répond à certaines **règles** et **notations**
- Le schéma ci-dessus est un **plan du site**

Répertoires et fichiers



- Si on est à la **racine du site** : `index.html`
`Welcome`
`Bienvenue`
`Ma photo`
- On indique le **nom du fichier** cible avec son chemin (suite des **répertoires séparés par /**)

Répertoires et fichiers



- Si on est sur Bienvenue : `bienvenue.html`
`Welcome`
`Racine du site`
`Ma photo`
- Pour **remonter vers le répertoire parent**, on indique le nom du fichier cible précédé de `../`

Ancre et id

- La balise **<a>** peut servir à atteindre une **ancre**, soit un point de repère fixe dans la page, pour se déplacer dans un document très long
- On indique l'ancre en plaçant une valeur dans l'**attribut id** d'une balise (exemple : id="top")
- On atteint la cible avec une balise **<a>** dont le href est le **nom de l'ancre précédé du signe #**

```
<header id="top"><h1>Chapitre 1</h1></header>  
<p>Ici se trouve un texte très long...bla bla bla</p>  
<a href="#top">Retour en haut de la page</a>
```

Balises de liste

- Avec les listes on peut créer des **énumérations**
- La **liste ordonnée** (numérotée) est composée d'un élément **** contenant des éléments ****
- La **liste non ordonnée** est composée d'un élément **** contenant des éléments ****
- La **liste de définition** est composée d'un élément **<dl>** contenant des éléments **<dt>** (terme à définir) et **<dd>** (définition)
- On utilise souvent les listes pour les **menus de navigation** qui sont des **énumérations de lien**

Attributs des listes

- La balise **** accepte les attributs :
 - start : pour définir la 1ère valeur de numérotation
 - reversed : pour une numérotation inversée
 - type : type de numérotation (numérique, lettres, chiffres romains,...)
- Les autres balises n'ont pas d'attributs spécifiques mais elles peuvent être stylées avec **CSS** et les propriétés du genre « **list-style-...** »

Balises pour le multimédia

- Un fichier **HTML** ne contient **que du texte** mais on peut afficher dans une page Web des images, des vidéos, du son,... **Comment ?**
- On va établir des relations avec des **fichiers externes** contenant ces ressources
- On va pouvoir les **manipuler avec JavaScript**

Les images

- L'élément **** est utilisé pour placer une image dans la page
- Les **formats** les plus utilisés sont **GIF**, **PNG**, **JPEG** et **SVG** (voir annexe pour plus de détails)
- **** est utilisé surtout pour des **logos** ou des **icônes cliquables**, des **photos d'illustration** d'un article,...
- Pour des **images de fond**, des **décorations**, des **séparations**,..., on préfère utiliser **CSS**

Attributs de

- **src** indique l'URL de l'image à afficher
- **alt** contient un texte alternatif à afficher si l'image ne peut pas être visualisée
- **title** affiche une infobulle au survol de l'image
- **width** et **height** indiquent la largeur et la hauteur de l'image (voir aussi les propriétés CSS)
- **usemap** et **ismap** permettent d'utiliser des zones cliquables dans les images (à voir avec les éléments <map> et <area>)

Les figures

- On a ajouté avec HTML5, les éléments **<figure>** et **<figcaption>** pour regrouper une image () et une légende (comme dans un livre) sous forme de texte ou de lien
- L'élément **<figure>** peut contenir aussi des éléments vidéo, audio, du code, des tableaux, des dessins (avec <canvas>),...
- La **légende** est placée dans **<figcaption>**

Éléments embarqués

- L'élément **<iframe>** place dans la page Web un cadre avec un autre document HTML
- L'élément **<embed>** intègre du contenu externe dans le document HTML (utilisé pour le Flash, ActiveX,...)
- Les éléments **<object>** et **<param>** jouent un rôle similaire à **<embed>**
- **Attention à la sécurité** avec tous ces éléments qui intègrent un contenu externe !

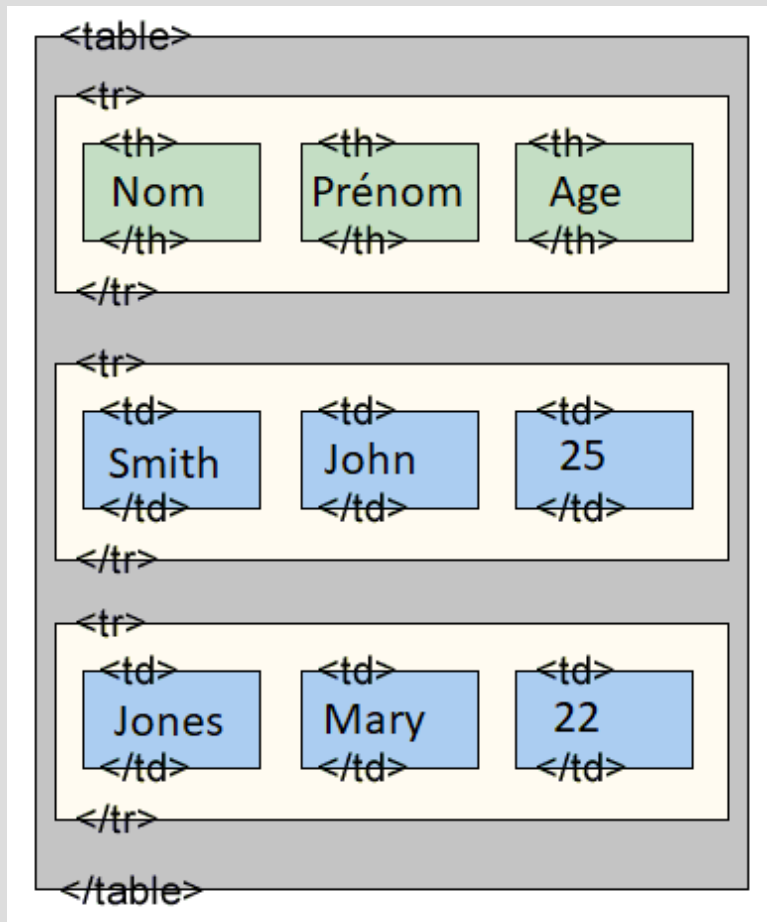
Son, Vidéo, Dessin

- **<audio>** permet d'ajouter du **son** dans une page Web
- **<video>** permet d'ajouter une **vidéo** dans une page Web
- **<canvas>** permet d'ajouter un **zone de dessin ou d'animation** dans une page Web
- Ces différents éléments sont **pilotables** avec du code **JavaScript**

Balises pour les tableaux

- L'élément **<table>** regroupe les données du tableau constitué de 1 ou plusieurs lignes **<tr>** contenant une ou plusieurs cellules **<td>**
- **<th>** peut remplacer **<td>** comme entête de ligne ou de colonne
- **<caption>** permet d'ajouter un **titre** au tableau
- **<thead>**, **<tfoot>**, **<tbody>** représentent respectivement **l'entête**, **le pied** et **le corps** du tableau. Ensemble, ils doivent être utilisés dans cet ordre. Mais ils ne sont **pas obligatoires**

Exemple de tableau minimaliste



Nom	Prénom	Age
Smith	John	25
Jones	Mary	22

- **<table>**, **<tr>** et **<td>** sont les éléments de base d'un tableau
- Les autres éléments sont optionnels

Autres balises et attributs

- Les éléments **<colgroup>** et **<col>** permettent de spécifier des propriétés aux colonnes
- Les attributs **rowspan** et **colspan** permettent de fusionner des lignes et des colonnes pour créer des tableaux plus complexes
- La **mise en forme** peut être améliorée avec des propriétés de **CSS**