

Qualiente

DÉVELOPPEMENT ET MISE EN LIGNE  
D'UN SITE WEB EN HTML / CSS

Formateur | Pierre Vélon



# Le langage HTML

# Le langage HTML

- HTML signifie HyperText Markup Language (langage de balises affichant du texte enrichi)
- Langage créé en 1991 par Tim Berners-Lee et permettant de présenter à un utilisateur les pages d'un site Internet
  - La version 5 est en cours de finalisation
- Organisation sous forme de balises, d'attributs et de texte
  - Une balise est un élément commençant par « < », se terminant par « > » et signifiant une instruction de présentation
  - Un attribut est une propriété d'une balise. Il peut y avoir plusieurs attributs pour une même balise ou aucun
    - <balise attribut="valeur">
  - Les balises HTML fonctionnent généralement par paires (ouverture et fermeture) et ne sont pas sensibles à la casse
    - L'instruction de présentation s'applique au texte entre les balises
- Exemple : <p align="center">Paragraphe 1</p>

# Structure d'un fichier HTML

- ❑ Un fichier HTML commence par la balise <html>, contient une entête et un corps, se termine par la balise </html>
- ❑ L'entête commence par la balise <head>, se termine par la balise </head>
  - ❑ Elle contient notamment le titre de la page et des informations non visibles de l'internaute
- ❑ Le corps commence par la balise <body>, se termine par la balise </body>
  - ❑ Il permet de préciser le contenu visible de la page
- ❑ Un fichier HTML vide est donc de la forme :  
<html>  
<head></head>  
<body></body>  
</html>

# Structure d'un fichier HTML

- Les sauts de ligne, les espaces et les tabulations au sein d'un fichier HTML n'influent pas sur la présentation et sont même souhaités pour la lisibilité du document
- Dans le cas où une balise n'est pas fermée, l'instruction de présentation est appliquée jusqu'à la fin du document ou jusqu'à ce qu'une autre balise vienne contredire cette instruction
  - Exemple : texte de taille 14 jusqu'à une instruction demandant un texte en taille 10
- Des commentaires non interprétés peuvent être insérés pour améliorer la lecture du code
  - <!-- commentaires -->
- Un fichier HTML a comme extension « .htm » ou « .html » et s'exécute dans un navigateur Web

# Les navigateurs Web

- Les principaux navigateurs du marché sont les suivants (statistiques de juillet 2012)
  - Internet Explorer (Microsoft) – 29 % de part de marché
  - Firefox (Mozilla) – 31 % de part de marché
  - Chrome (Google) – 29 % de part de marché
  - Safari (Apple) – 9 % de part de marché
  - Opera (Opera) – 1 % de part de marché
- Le navigateur Web est un logiciel très complexe
  - Les navigateurs n'affichent pas tous les sites de la même façon
    - Le développeur doit prendre l'habitude de vérifier le bon fonctionnement d'un site Web sur l'ensemble des navigateurs (y compris les navigateurs mobiles)

# L'entête d'un fichier HTML

- La première ligne correspond au type de document
  - `<!DOCTYPE html>`
- La balise `<title>` permet de définir le titre de la page (moins de 100 caractères en général)
- La description de la page (affichée par les moteurs de recherche) est précisée dans une balise `<meta>` avec les attributs suivants :
  - `<meta name="description" content="votre description">`
- Les mots-clés de la page (ou du site associé) sont précisés dans une balise `<meta>` avec les attributs suivants :
  - `<meta name="keywords" content="mot-clé1,mot-clé2">`
- Pour gérer tous les caractères spéciaux, y compris les accents de la langue française, il est nécessaire d'ajouter une ligne relative au jeu de caractères
  - `<meta charset="utf-8">`

# Corps HTML : balises textes

- Gras : `<b>`, Italique : `<i>`, Souligné : `<u>`, Barré : `<strike>`, Exposant : `<sup>`, Indice : `<sub>`
  - Exemple : `<b><u>Texte en gras et souligné</u></b>`
- Position du texte : balise `<div>` et attribut « align »
  - Gauche (par défaut)
    - `<div align="left">Texte à gauche</div>`
  - Centré
    - `<div align="center">Texte centré</div>`
  - Droit
    - `<div align="right">Texte à droite</div>`
- Couleur du texte : balise `<font>` et attribut « color »
  - Les couleurs sont codées selon la notation hexadécimale
    - 6 caractères de 0 à 9 et/ou de A à F
    - Exemple : `<font color="FF0000">Texte rouge</font>`



# Corps HTML : balises textes

- Taille du texte : balise `<font>` et attribut « size »
  - Les tailles HTML sont définies sur une échelle de 1 à 7
    - Il est toutefois possible d'utiliser des tailles en pixels (1 pixel = 0,025 cms) ou en points
    - Exemple : `<font size="2">Texte de taille HTML 2</font>`
- Style du texte : balise `<font>` et attribut « face »
  - Les polices pouvant être utilisées sont les mêmes que dans un outil de traitement de texte
    - Dans le cas où l'utilisateur ne possède pas celle-ci, une police par défaut sera appliquée)
    - Exemple : `<font face="Verdana">Texte en Verdana</font>`
- La balise `<p>` permet de créer un paragraphe
- Les sauts de ligne entre les différentes zones de texte sont assurés par la balise `<br>`, qui ne nécessite pas de balise de fermeture

# Corps HTML : images

- Les formats possibles pour une image au format Web sont : png, jpg (ou jpeg), gif, bmp (à éviter, taille des fichiers souvent trop importante)
  - Le PNG est le format le plus récent, il peut être rendu transparent et n'altère pas la qualité de l'image
- Une image est insérée dans une page HTML grâce à la balise orpheline `<img>`
  - Attributs obligatoires
    - « src » : chemin de l'image
    - « alt » : texte alternatif, affiché à la place de l'image si problème
  - Attributs facultatifs
    - « title » : infobulle (légende affichée au passage de la souris)
    - « border » : taille de la bordure encadrant l'image
    - « width » : largeur de l'image
    - « height » : hauteur de l'image
- Une image sert d'illustration ou de lien vers une autre page
- Exemple : ``

# Corps HTML : liens hypertexte

- Un lien hypertexte est un élément permettant d'ouvrir une autre ressource HTML (section de la même page, autre page, document, image, vidéo...)
- Un lien est créé avec la balise `<a>`, associée à l'attribut « href »
  - Exemple : `<a href="page2.html">Page 2</a>`
- L'attribut « target » permet de préciser si le navigateur Internet doit ouvrir une nouvelle page (ou un nouvel onglet au sein des navigateurs les plus récents)
  - Par défaut, l'ouverture se fait au sein de la même page
  - Exemple d'ouverture dans une nouvelle page :  
`<a href="page2.html" target="_blank">Page 2</a>`

# Corps HTML : tableaux

- Un tableau HTML permet d'améliorer la forme du contenu proposé dans une page Web
- Il s'ouvre avec la balise <table>
  - Une ligne est définie avec la balise <tr>
  - Une colonne est définie avec la balise <td>
  - Les colonnes de la ligne d'entête sont définies avec <th> et non <td>
  - Exemple d'un tableau d'une ligne / une colonne  
`<table><tr><td>Cellule 1</td></tr></table>`
- Les attributs suivants sont facultatifs
  - « width » : largeur du tableau, de la ligne ou de la colonne
  - « height » : hauteur du tableau, de la ligne ou de la colonne
  - « border » : taille de la bordure du tableau
  - « bgcolor » : couleur de fond du tableau , de la ligne ou de la colonne
  - « cellpadding » : espacement entre les cellules du tableau
  - « cellspacing » : retrait à gauche du texte dans une cellule du tableau
  - « align » : positionnement du tableau (gauche, droite, centré ou justifié), de la ligne ou de la colonne
- Un titre peut être ajouté avec la balise <caption>

# Corps HTML : formulaires

- Un formulaire permet aux internautes de renseigner des informations
- Il est géré par la balise `<form>` avec les attributs suivants :
  - « name » : nom du formulaire
  - « method » : get (par défaut, informations affichées dans la barre d'adresse) ou post (informations non visibles)
  - « action » : adresse de la page qui va traiter les informations
- Les 3 éléments principaux d'un formulaire sont :
  - `<input>` : zone de texte court, bouton radio, case à cocher (balise orpheline)
    - Exemple : `<input type="password">`, champ de type mot de passe
  - `<select>` : liste déroulante
    - Chaque valeur de la liste est encadrée par la balise `<option>`
  - `<textarea>` : zone de texte long
    - Exemple : `<textarea>Insérez vos commentaires</textarea>`
- Chaque zone de formulaire doit contenir un attribut « name » pour que les informations puissent ensuite être récupérées dans la page de traitement
  - Exemple : `<input type="text" name="email">` E-mail
- HTML 5 apporte de nouvelles valeurs possibles pour l'attribut type de la balise input : email, url, tel, number, range, color, date, search...



# Le langage CSS

# Le langage CSS

- Les fichiers CSS (extension « .css ») ont été créés afin de regrouper toutes les instructions de forme (couleur, taille, police, décoration, marge...) initialement déclarées dans les fichiers HTML
  - La structure d'une page et sa présentation sont gérées dans des fichiers différents mais complémentaires
  - La conception d'une page se fait dans un premier temps sans se soucier de la présentation, ce qui permet de gagner en efficacité
  - Le code HTML est considérablement réduit en taille et en complexité puisqu'il ne contient plus de balises ni d'attributs de présentation
  - Les pages HTML font référence à la même feuille de styles, la présentation d'un site Web est donc uniformisée et peut être modifiée en très peu de temps
  - Un même document peut avoir plusieurs feuilles de style associées, une pour la lecture à l'écran par exemple et une pour l'impression papier
- CSS signifie Cascading Style Sheet (feuille de style en cascade)
  - Publication des premières spécifications du langage fin 1996
  - La variante CSS 2.1 est initiée en 2001 et validée en 2007, les navigateurs Web ont profité de cette longue période pour intégrer un certain nombre de syntaxes CSS non clairement spécifiées
  - La version 3 du langage CSS est en cours de finalisation
- L'appel du fichier CSS se fait dans l'entête du fichier HTML
  - `<link rel="stylesheet" type="text/css" href="mafeuilledestyle.css">`
  - Il est également possible de déclarer des styles dans l'entête HTML ou au niveau d'une balise

# Propriétés, valeurs et déclarations

- Une propriété est une syntaxe définie par les spécifications CSS et indiquant la mise en forme de l'élément auquel elle s'applique
  - Les propriétés se regroupent par type (typographie, positionnement...) mais aussi par média (écran, imprimante, mobile, auditif, projection...)
  - CSS 1 comportait 53 propriétés, CSS 2.1 en propose 115 et CSS 3 en admet 245 pour le moment
- Les normes CSS spécifient les valeurs pour chaque propriété
  - Exemple : la propriété « color » admet « #000000 » comme valeur
- Le terme déclaration désigne un ensemble formé par la propriété et sa valeur associée sous la forme : « propriété: valeur »
  - Exemples : « color: #000000 », « font-style: italic »
- Il est possible de regrouper plusieurs déclarations en les séparant par des points-virgules
- Un bloc de déclarations commence par « { » et se termine par « } » (même s'il n'y a qu'une déclaration)



# Les propriétés les plus utilisées

## □ Texte

- font-family : nom de la police à utiliser
- font-size : taille du texte
- font-weight : poids du texte (normal ou gras)
- font-style : inclinaison du texte (normal ou italique)
- text-decoration : décoration du texte (normal, souligné, surligné, souligné et surligné ou barré)
- text-transform : modification de la casse (minuscules ou majuscules)
- text-align : alignement horizontal du texte (gauche, droite, centré ou justifié)
- color : couleur du texte
- opacity : transparence du texte (valeur entre 0 et 1)

# Les propriétés les plus utilisées

## □ Fond

- background-color : couleur de fond
- background-image : image de fond
- background-attachment : position du fond (fixe ou défilant)

## □ Boîtes

- display : type de boîte (les plus fréquents : block et inline)
- width : largeur du contenu de la boîte, hors retrait
- height : hauteur du contenu de la boîte, hors retrait
- margin : marge autour de la boîte
- padding : retrait du texte au sein de la boîte
- border : épaisseur, type et couleur de la bordure
- border-radius : arrondi de la bordure
- visibility : visibilité de la boîte (visible ou masqué)
- z-index : ordre d'affichage en cas de superposition (+ > -)

# Les propriétés les plus utilisées

## □ Liste

- list-style-type : type de puce (point noir, point creux...)
- list-style-image : puce personnalisée
- list-style-position : position du texte en cas de retour automatique à la ligne
  - inside : le texte s'aligne au début de la puce
  - outside : le texte s'aligne à la fin de la puce

## □ Tableaux

- border-collapse : type de bordure (fusion, séparation)
- empty-cells : affichage des cellules vides (oui/non)
- caption-side : position du titre du tableau (haut ou bas)

## □ Divers

- cursor : curseur de la souris (pointeur, chargement...)

# Les sélecteurs CSS

- Le terme de sélecteur désigne la partie précédant l'accolade ouvrante du bloc de déclaration
  - ▣ Il indique l'élément sur lequel vont s'appliquer les déclarations CSS présentes dans le bloc qui lui est associé
  - ▣ Il peut caractériser plusieurs types d'éléments
    - Sélecteur de balise : « p », « h1 », « h2 », « div », « a »...
    - Sélecteur de classe (une classe est la valeur de l'attribut « class » de balises HTML d'un même groupe) : « .sport », « .fruit »...
    - Sélecteur d'identifiant (un identifiant est la valeur de l'attribut « id » d'une unique balise HTML) : « #tennis », « #kiwi »...
    - Sélecteur d'attribut : « [href] », « [align] »...
- Plusieurs sélecteurs peuvent précéder un bloc de déclaration en les séparant par une virgule
  - ▣ Exemple : « h1, h2, h3 { font-weight: bold; } »
- Un sélecteur ne peut être composé que de caractères alphanumériques, de « - » et de « \_ »
  - ▣ Il ne peut pas commencer par un chiffre ou un « - »
  - ▣ Les caractères spéciaux ne sont pas valides

# Sélecteurs de classe

- Partie HTML

- `<span class="erreur">Champ obligatoire</span>`

- Partie CSS

- `.erreur { color: red; }`

- Le langage CSS permet également de mettre en forme des éléments selon un état contextuel (concept de pseudo-classe)

- Exemple : les différents états d'un lien hypertexte

- `a:link` : lien non visité

- `a:visited` : lien visité

- `a:active` : lien au moment du clic sur celui-ci (avant relâchement)

- `a:hover` : lien au passage de la souris

# Sélecteurs d'identifiant

## □ Partie HTML

- `<ul id="nav"><li>Partie 1</li><li>Partie 2</li></ul>`

## □ Partie CSS

- `#nav { padding: 30; }`

## □ Une page HTML est souvent structurée avec les identifiants suivants :

- `#header` : en-tête

- `#footer` : pied de page

- `#aside` : partie latérale

- `#main` : partie latérale

- `#nav` : menu de navigation

# Sélecteurs d'attribut

- ❑ Le sélecteur d'attribut permet de cibler un attribut HTML ou la valeur d'un attribut
  - ❑ [attribut] { propriété: valeur; }
- ❑ Il est possible de préciser l'élément concerné
  - ❑ E[attribut] { propriété: valeur; }
- ❑ D'autres correspondances sont possibles
  - ❑ [attr|="mot"] : attribut dont la valeur commence par « mot »
  - ❑ [attr\*="mot"] : attribut dont la valeur contient « mot »
- ❑ Exemples
  - ❑ input[type="text"] [id="nom"]
    - Zones de texte court ayant « nom » comme identifiant
  - ❑ img[alt]
    - Éléments images ayant un attribut alt

# Ancêtres, parents et frères

- Les différents éléments d'un fichier HTML (titres, paragraphes, liens, images...) s'organisent sous la forme d'un arbre généalogique, également nommé « arbre du document » (ou DOM pour Document Object Model)
  - Cet arbre généalogique est composé de fratries et de degrés de parenté qui sont exploités par les sélecteurs CSS pour cibler les différents éléments du document
- Soit X un élément situé au sein de l'arborescence
  - X est susceptible d'être entouré d'éléments ancêtres, descendants, parents, enfants et frères
- Le terme d'ancêtre désigne tout élément situé dans la même branche que l'élément X mais à un niveau supérieur dans la hiérarchie
  - Le parent est un élément unique qui fait référence à l'élément placé directement au-dessus de X
- Le concept de descendant fait référence à tout élément qui est situé dans la même branche que X mais à un niveau plus bas dans la hiérarchie
  - Les enfants sont les éléments directement placés sous X
- Tous les éléments situés au même niveau que X et partageant le même parent sont appelés frères
- Les sélecteurs avancés ciblant les éléments selon leur hiérarchie sont utilisables depuis Internet Explorer 7



# Sélecteurs d'enfant

- Le sélecteur d'enfant s'applique aux enfants d'un élément

- $E > F$  { propriété: valeur; }

- Déclaration pour l'élément F s'il est enfant de E

- Exemple :

<p>Un paragraphe contenant

    <a>un lien

        <b>important</b>

    </a>

</p>

- $p > b$  { color: green; } /\* b non enfant de p, pas de couleur \*/

- $p b$  { color: red; } /\* b descendant de p, b devient rouge \*/

- $a > b$  { color: blue; } /\* b enfant de a, b devient bleu \*/

- $a b$  { color: yellow; } /\* b descendant de a, b devient jaune \*/

# Sélecteurs de frère adjacent

- Le sélecteur de frère adjacent affecte un élément à condition qu'il soit immédiatement précédé d'un autre élément spécifié
  - `E + F { propriété: valeur; }`
    - Déclaration pour l'élément F s'il est frère de E et s'il lui succède dans l'ordre de déclaration HTML
- Exemple :
  - `<h1>Titre de la page</h1>`
  - `<p>Paragraphe 1</p>`
  - `<p>Paragraphe 2</p>`
  - `h1 + p { font-style: italic; }`
    - Seul le premier paragraphe sera en italique

# Règles @

- La règle `@import` inclut une feuille de styles CSS au sein d'une autre feuille de styles
  - Cette règle doit absolument être placée en tête du fichier sous peine d'être invalide
    - Exemple : `@import url(css/feuille2.css);`
- La règle `@media` indique qu'un bloc de règles ne concernera que des périphériques de sortie déterminés
  - Les principaux médias reconnus sont all (tout média), aural et speech (synthèses vocales), handheld (mobiles), print (imprimantes), projection (projecteurs), screen (écrans d'ordinateurs)
    - Exemple :  
`@media screen { body {background: url(photo.jpg); } }`  
`@media print { body {background: #FFFFFF; } }`
  - A savoir : Safari Mobile se comporte en screen



# La mise en ligne de pages Web

# Les outils de vérification d'un code source

- ❑ Firebug (Firefox, Chrome, Safari et Opera)
  - Ce module permet d'inspecter en détail les codes HTML, de modifier en temps réel les propriétés CSS et la position des différents éléments de la page
- ❑ Web Developer (Firefox et Chrome)
  - Ce module permet de masquer des éléments, de désactiver les styles CSS et de modifier la résolution d'affichage de la page
- ❑ Fireshot (Firefox) / Capture de page Web (Chrome)
  - Ce module permet de faire des captures d'écran (pleine page, partie visible de l'écran ou zone personnalisée)
- ❑ ColorZilla (Firefox et Chrome)
  - Ce module permet d'afficher et de conserver dans le presse-papier la couleur de la zone survolée par la souris
- ❑ IETester (Internet Explorer)
  - Cette application permet de simuler l'affichage d'une page Web sur toutes les versions du navigateur depuis IE 5.5

# Le transfert de fichiers

- ❑ Il est conseillé de regrouper les fichiers de même nature au sein de répertoires spécifiques (images, documents, style, videos...)
- ❑ Le transfert de fichiers se fait grâce à un logiciel FTP (File Transfer Protocol)
  - ❑ Logiciel opensource très utilisé : FileZilla
  - ❑ 2 vues utilisateur
    - Fichiers et répertoires de l'ordinateur du développeur à gauche
    - Fichiers et répertoires du serveur distant à droite
- ❑ L'accès à un serveur Web distant via le protocole FTP nécessite les informations suivantes :
  - ❑ Adresse du serveur FTP
    - Exemple : ftp.free.fr
  - ❑ Identifiant / mot de passe du compte FTP
    - Accès à tout ou partie de l'arborescence du serveur