



**IMT Atlantique**  
Bretagne-Pays de la Loire  
École Mines-Télécom

# HTML5 & CSS3

Issam REBAI – Équipe MOTEL – Département informatique – Campus de Brest  
✉ [Issam.rebai@imt-atlantique.fr](mailto:Issam.rebai@imt-atlantique.fr)  
Version 0.1 [R.2018.10.11]



**IMT Atlantique**  
Bretagne-Pays de la Loire  
École Mines-Télécom

HTML5  
CSS3  
Responsive Web Design


## AU SOMMAIRE

INT475 - Développement Web  
Front End pour les débutants

2



**IMT Atlantique**  
Bretagne-Pays de la Loire  
École Mines-Télécom




Objectifs, philosophie  
Nouveautés  
Les formulaires  
Le multimédia

# HTML5 – SOMMAIRE

INT475 - Développement Web  
Front End pour les débutants

[ 3 ]






# HTML5 ► Point historique

- Fin 2006 : W3C décide de réfléchir à l'avenir du HTML
  - Échec du XHTML 2
  - Les travaux du WHATWG (*Web Hypertext Application Technology Working Group*)
    - Ian Hickson (Opera)
    - Mozilla, Opera, Apple et Google
- 22 janvier 2008 : Spécification en Working Draft
- 28 octobre 2014 : Recommandation finale HTML5
- 03 octobre 2017 : Recommandation HTML5.1
- 14 décembre 2017 : Recommandation HTML5.2
  - <https://www.w3.org/TR/html52/>
- Pour les *designers* Web : liste des éléments HTML
  - <http://w3c.github.io/html-reference/>

INT475 - Développement Web  
Front End pour les débutants

[ 4 ]



# HTML5 ▶ Objectifs

- Rester compatible avec l'existant (contrairement à XHTML 2)
- Décrire précisément ce que doit faire et afficher les navigateurs (HTML 4.01 comporte énormément d'imprécisions)
- Gérer le traitement des erreurs
  - Que doit afficher le navigateur si le document est invalide ?
- Bien gérer l'interopérabilité et la compatibilité des navigateurs
  - Gestion identique du DOM (l'arbre du document)

Pour plus d'infos voir la source: **HTML Design Principles**

<https://www.w3.org/TR/html-design-principles/>

HTML5

INT475 - Développement Web  
Front End pour les débutants

5

IMT Atlantique  
Campus de Brest

# HTML5 ▶ Philosophie

+

- Plus souple
- Peu contraignant

⊖

- Perte de la ressemblance des scripts
- Difficulté à interpréter le code d'autrui

HTML5

INT475 - Développement Web  
Front End pour les débutants

6

IMT Atlantique  
Campus de Brest

# HTML5

## ► Simplification

- Éléments qui ne sont plus nécessaires : <html>, <head>, <body>, <thead>, <tfoot>, <tbody>
- Simplification
  - DOCTYPE : utile que pour les outils de validation
  - Encodage des caractères
  - Déclaration d'un *script*
    - <script src="file.js"/>
  - Déclaration d'une feuille de style
    - <link href="myStyle.css" rel="stylesheet"/>

```
1.<!DOCTYPE html>
2.<meta charset=utf-8/>
3.<title>My HTML page</title>
4.<h1>Title of my page</h1>
5.<p>My text</p>
```

```
1.<!DOCTYPE html>
2.<html lang="en">
3.<head>
4.<meta charset="utf-8">
5.<title>My HTML page</title>
6.</head>
7.<body>
8.<h1>Title of my page</h1>
9.<p>My text</p>
10.</body>
11.</html>
```

HTML5

INT475 - Développement Web  
Front End pour les débutants

7

IMT Atlantique  
Brest - Nantes - La Rochelle

# HTML5

## ► Permissivité

- Les attributs : guillemets non obligatoires sauf pour les attributs à plusieurs valeurs
  - <meta **charset=utf-8**/>
  - <div class="style1 style2">
- Les attributs booléens : valeur optionnelle
  - <input type="checkbox" name="vehicle" value="Car" **checked**> I have a car<br>
- Éléments avec la seule balise d'ouverture : indication de fermeture optionnelle
  - <br/>
  - 
- Éléments à balise de fermeture optionnelle ré-autorisés : plus besoin de fermer certaines balises comme : <p>, <dl>, <td>, ...
  - <p> un paragraphe sans la balise fermante.
- Casse des éléments et attributs : tolérance complète
  - <mEtA Name="viewport" **CONTENT**="width=device-width">

HTML5


INT475 - Développement Web  
Front End pour les débutants

8

IMT Atlantique  
Brest - Nantes - La Rochelle


# HTML5 ▶ Sémantisation

- L'attribut **role** permet d'apporter des informations sémantiques complémentaires sur le contenu des éléments structurels d'une page Web
  - Exemples d'usage :
    - Meilleure indexation par les moteurs de recherche
    - Gestion du contenu des pages pour les déficients visuels
  - Les valeurs de l'attribut **role** :
    - **main** : contenu principal du document
    - **secondary** : partie secondaire du document
    - **navigation** : barre de navigation du document
    - **banner** : bannière contenant normalement le logo, le slogan de l'entreprise
    - **contentinfo** : des informations à propos du contenu (auteur, copyrights, etc.)
    - **definition** : la définition d'un élément
    - **note** : note entre parenthèses ou de bas de page
    - **seealso** : informations en relation avec le contenu principal de la page
    - **search** : formulaire de recherche d'une page
  - Structuration de la page HTML avec de nouvelles balises sémantiques :



INT475 - Développement Web  
Front End pour les débutants

9



# HTML5 ▶ Sémantisation

`<header>` : zone pour l'entête d'une zone affichage (page, section, article, etc.)


`<footer>` : zone pour pied de page d'une zone affichage (page, section, article, etc.)

`<nav>` : zone de navigation avec des liens hypertextes

- Possibilité d'avoir plusieurs zones de navigation dans une même page (primaire, secondaire)


`<section>` : regroupe des éléments ayant la même thématique

- Contenu + entête + pied



INT475 - Développement Web  
Front End pour les débutants

10



# HTML5 ▶ Sémantisation

<article> : contenu autonome réutilisable ailleurs sans que sa compréhension soit compromise

- Entête + pied + titre + texte

<aside> : contenu lié au contenu auquel il est associé (barre latérale, complément sur un article)

<main> : contenu principal de la page

- Une seule balise par page
- Ne peut pas être à l'intérieur d'<article>, <aside>, <footer>, <header> ou <nav>

<div> : zone de contenu encore valable dans HTML5

HTML5

INT475 - Développement Web  
Front End pour les débutants

11

IMT Atlantique  
Brest - Nantes - La Rochelle

# HTML5 ▶ Sémantisation ▶ Exemple

<header>

<nav>

<section>

<article><aside>

<article><aside>

<footer>

INT475 - Développement Web  
Front End pour les débutants

12

IMT Atlantique  
Brest - Nantes - La Rochelle

Issam REBAI

6

HTML5

HTML5 ▶ Dépréciation ▶  
Éléments

<frame>

<frameset>

<noframes>

<applet>

<acronym>

<bgsound>

<dir>

<hgroup>

<isindex>

<basefont>

<big>

<blink>


<center>

<font>

<marquee>

INT475 - Développement Web  
Front End pour les débutants

13

  
IMT Atlantique  
Développement Web  
Front End pour les débutants

HTML5

HTML5 ▶ Enrichissement ▶  
Éléments

<time>

<mark>

<meter>

<detail> et  
<summary>

<wbr>

<bdi>

<progress>


<ruby>, <rt>  
et <rp>

<figure> et  
<figcaption>

<template>

INT475 - Développement Web  
Front End pour les débutants

14

  
IMT Atlantique  
Développement Web  
Front End pour les débutants

HTML5

HTML5 ▶ Dépréciation ▶ Texte

<acronym>

<basefont>

<font>

<big>

<center>

<strike>

<tt>

<u>

INT475 - Développement Web

Front End pour les débutants

15

HTML5

HTML5 ▶ Redéfinition ▶ Texte

<b>

<strong>

<i>

<em>

<small>

<cite>

<dl>

<ol>

<hr>

<a>

<s>

<u>

INT475 - Développement Web

Front End pour les débutants

16



# HTML5 ▶ Principe d’affichage

- L’ancien principe d’affichage HTML 4 (`block`, `inline`) est devenu obsolète
- Le concepteur du site Web doit indiquer les règles d’affichage des éléments
  - Si ce n’est pas le cas alors la feuille de style par défaut du navigateur sera appliquée
  - La propriété **display** en CSS permet d’indiquer comment afficher les éléments du document HTML

HTML5

INT475 - Développement Web  
Front End pour les débutants

17

IMT Atlantique  
Brest - Nantes - La Rochelle

Par binôme :

1. Étudier un nouvel élément de HTML5 (rôle et attributs)
2. Réaliser un exemple personnel
3. Partager sur l’espace commun l’explication (rôle et attributs) et l’exemple

## HTML5 ▶ ACTIVITÉ PRATIQUE COLLABORATIVE

HTML5

INT475 - Développement Web  
Front End pour les débutants

18

IMT Atlantique  
Brest - Nantes - La Rochelle

## HTML5 ▶ Formulaire



- En HTML 4
  - La validation des données saisies dans les formulaires se fait coté serveur et optionnellement, pour de meilleures performances, coté client avec du JavaScript
  - Les types de contenu offrent une interactivité réduite : zone de texte, liste déroulante, bouton radio, case à cocher, téléchargement de fichier
  - Les envois de données se font via les méthodes GET ou POST
  - L'élément `<form>` doit contenir tous les champs du formulaire



## HTML5 ▶ Formulaire ▶ Exemple



```
<form action="urlDestination" method="post">
  <div>
    <label for="nom">Nom :</label>
    <input type="text" id="nom" name="user_name">
  </div>
  <div>
    <label for="courriel">Courriel :</label>
    <input type="text" id="courriel" name="user_email">
  </div>
  <div>
    <label for="message">Message :</label>
    <textarea id="message" name="user_message">
  </textarea>
  </div>
  <div class="button">
    <button type="submit">Envoyer le message</button>
  </div>
</form>
```



HTML5 ▶ Formulaire ▶ Exemple

HTML5

INT475 - Développement Web  
Front End pour les débutants

21

IMT Atlantique  
Brest - Nantes - La Rochelle

Nom :

Courriel :

Message :

Envoyer le message

HTML5 ▶ Formulaire

HTML5

INT475 - Développement Web  
Front End pour les débutants

22

IMT Atlantique  
Brest - Nantes - La Rochelle

• En HTML5

• La validation des données se fait nativement par le navigateur ➔ plus besoin d’écrire du code JavaScript mais **toujours continuer à vérifier les données coté serveur**


• Les envois de données se font via les méthodes GET, POST, PUT et DELETE (ces deux dernières sont retirées de HTML5)

• Les champs du formulaire peuvent se trouver en dehors de l’élément `<form>`

• Création de nouveaux types de champs plus interactifs

- Chaque navigateur est libre de choisir comment seront affichés ces éléments
- Rendus différents selon le type de terminal (PC, « *smartphone* », tablette, etc.).
- Si un navigateur ne connaît pas le type d’input, il appliquera par défaut le type texte


HTML5 ▶ Formulaire ▶ Nouveaux types




email	tel	url
date	time	number
range	search	color

INT475 - Développement Web  
Front End pour les débutants

( 23 )



HTML5 ▶ Formulaire




- Exemple de champs en dehors du `<form>`

```
<form id="myForm">  
  <input type="texte"/>  
  <input type="submit"/>  
</form>  
<div>Other content</div>  
<textarea form="myform"> Change the content of  
this element</textarea>
```
- La validation coté client par le navigateur se fait en se basant sur :
  - Le type de l'input
  - Des attributs communs à tous les types d'input
    - `required` : booléen indiquant si le champ est obligatoire ou non
    - `pattern` : regex définissant la saisie autorisée dans le champ
  - Des attributs spécifiques à certains inputs
    - `max`, `min` : pour les nombres
    - `minlength`, `maxlength` : pour les textes

INT475 - Développement Web  
Front End pour les débutants

( 24 )



# HTML5 ▶ Formulaire

- En cas d'erreur de saisie
  - L'envoi des données est bloqué par le navigateur
  - Le navigateur est totalement libre de l'affichage utilisé pour signaler les erreurs de saisie
- Améliorations ergonomiques
  - `placeholder` : attribut texte d'aide à la saisie en affichant un message dans la zone de saisie qui sera effacé dès que l'utilisateur clique sur le champ
  - `autofocus` : attribut booléen pour affecter le focus à un champ
  - `autocomplete` : attribut d'input et de form acceptant la valeur `on` ou `off`  
Il indique au navigateur s'il affiche les dernières données saisies dans le champ ou le formulaire. Par défaut, c'est `on`

HTML5

INT475 - Développement Web Front End pour les débutants

25

IMT Atlantique

# HTML5 ▶ ACTIVITÉ PRATIQUE

Par binôme :

1. Réaliser un formulaire Web en HTML5 utilisant les différents types d'input
2. Exploiter les attributs spécifiques à chaque type
3. Tester le fonctionnement du formulaire

HTML5

INT475 - Développement Web Front End pour les débutants

26

IMT Atlantique

## HTML5 ► Multimédia



- Avant HTML5
  - Navigateur peut afficher : texte, images fixes (gif, png, jpg), images animées (gif), JavaScript
  - Autre contenu (animations interactives, audio, vidéo, etc.) : usage d'un moteur d'interprétation → *plugins* (FlashPlayer d'Adobe)
  - Inconvénients
    - Code d'insertion pour le *plugin* dans du HTML
    - Lourd, failles de sécurité, *plugin* absent, plugin bloqué, version incompatible, lenteur
- Avec HTML 5
  - Création d'éléments pour insérer de l'audio et de la vidéo
  - Problèmes : format audio et vidéo, droit d'auteur (DRM), interaction avec les utilisateurs, compatibilité avec tous les navigateurs → Reste à faire



## HTML5 ► Multimédia ► Audio



- Pour insérer un fichier audio (son, musique, podcast, etc.), il faut utiliser l'élément audio
  - Syntaxe basique: `<audio src="myfile.mp3"></audio>`
- Pour paramétrer le lecteur audio, il faut ajouter des attributs
  - `controls` : booléen pour afficher la barre de contrôle native du lecteur
  - `autoplay` : booléen pour lancer le fichier au chargement de la page → à éviter
  - `loop` : booléen pour lecture en boucle → pas ergonomique
  - `preload` : 3 valeurs (**auto**, none, metadata)
  - `muted` : mettre le son à 0
- Exemple à ne pas faire
  - `<audio src="myfile.mp3" controls preload autoplay loop></audio>`
  - Remarque: `preload` est ignoré quand `autoplay` est utilisé
- Formats audio pour le HTML5
  - Le W3C ne précise pas de format audio à privilégier
  - La « guerre des formats »
  - Chaque navigateur ne supporte que certains formats



## HTML5 ▶ Multimédia ▶ Audio

Navigateur	MP3	Ogg	Wav	acc
Chrome	Oui	Oui	Oui	Oui
Safari	Oui	Non	Oui	Oui
Firefox	Oui (depuis la version 21)	Oui	Oui	Non
Internet Explorer	Oui	Non	Non	Non
Opera	Oui	Oui	Oui	Non

- MP3 supporté par la plupart des navigateurs récents
- Compatibilité avec la majorité des navigateurs anciens et récents → utiliser plusieurs formats audio

```
<audio controls>
  <source src="myFile.mp3" type=""/>
  <source src="myFile.ogg" audio/mpeg" type="audio/ogg;
codecs='vorbis'"/>
  <source src="myFile.acc" type="audio/acc"/>
  Merci de mettre à jour votre navigateur pour profiter de l'audio
</audio>
```
- Il est possible de personnaliser l'interface du lecteur via l'API audio et du JavaScript → voir <https://www.w3.org/html/wiki/Elements/audio>

HTML5

INT475 - Développement Web Front End pour les débutants

29

IMT Atlantique

## HTML5 ▶ Multimédia ▶ Vidéo

- Pour insérer un fichier vidéo, il faut utiliser l'élément `video`
  - Syntaxe basique: `<video src="myVideo.mp4"></ video >`
- Pour paramétrer le lecteur vidéo, il faut ajouter des attributs
  - `controls` : booléen pour afficher la barre de contrôle native du player
  - `autoplay` : booléen pour lancement du fichier au chargement de la page → à éviter
  - `loop` : booléen pour lecture en boucle → pas ergonomique
  - `preload` : 3 valeurs (**auto**, none, metadata)
  - `muted` : mettre le son à 0
  - `height` et `width` : taille de la vidéo
  - `poster` : url de l'image à afficher en attente de la lecture de la vidéo
- Exemple

```
<video src="myFile.mp4" controls width="800"
height="600" poster="MyPoster.jpg"></video>
```
- Formats vidéo pour le HTML5 (idem que pour l'audio)
  - Le W3C ne précise pas de format audio à privilégier
  - La « guerre des formats »
  - Chaque navigateur ne supporte que certains formats

HTML5

INT475 - Développement Web Front End pour les débutants

30

IMT Atlantique

HTML5

HTML5 ▶ Multimédia ▶ Vidéo

Navigateur	H.264 (.mp4)	Ogg Theora (.ogv)	WebM (.webm)
Chrome	Oui	Oui	Oui
Safari	Oui	Non	Non
Firefox	Oui (depuis la version 21)	Oui	Oui
Internet Explorer	Oui	Non	Non
Opera	Oui	Oui	Oui

- MP4 supporté par la plupart des navigateurs récents
- Compatibilité avec la majorité des navigateurs anciens et récents → utiliser plusieurs formats vidéo

```
<video controls>
  <source src="myFile.ogg" type="video/ogg ;
codecs='theora,vorbis'"/>
  <source src="myFile.mp4"/>
  <source src="myFile.webm" />
  Merci de mettre à jour votre navigateur pour profiter de la vidéo
</video>
```
- Il est possible de personnaliser l'interface du *player* via l'API vidéo et du JavaScript → voir <https://www.w3.org/html/wiki/Elements/video>

INT475 - Développement Web  
Front End pour les débutants

31

IMT Atlantique  
Brest - Nantes - La Rochelle

HTML5

HTML5 ▶ Multimédia ▶ Vidéo

- Un format vidéo c'est 3 couches
  - Enveloppe : conteneur pour encapsuler le fichier audio et le fichier vidéo
    - Exemples : .avi, .mov, .mp4, .mkv
  - Codec Audio : compresser le son
    - Exemples : MP3, OGG, AAC, etc.
  - Codec Vidéo : compresser les images
    - Exemples : H.264 puissant mais a des problèmes de droit et de diffusion ; Ogg Theora utilisé dans le monde Linux ; WebM proposé par Google
- Outil pour convertir les vidéos
  - Miro Video Converter : <http://www.mirovideoconverter.com/>

INT475 - Développement Web  
Front End pour les débutants

32

IMT Atlantique  
Brest - Nantes - La Rochelle



# HTML5 ▶ Bonnes pratiques

- Syntaxe : respecter les règles d’usage et de syntaxe du XHTML
  - Balises en minuscule
  - Utiliser toujours des guillemets
  - Fermer par ‘/’ les balises auto fermantes
  - Fermer les balises dont la fermeture est optionnelle
  - Indenter le code
- Structure :
  - « Sémantiser » autant que possible le document
  - Éviter les éléments et attributs obsolètes

HTML5

INT475 - Développement Web  
Front End pour les débutants

33

IMT Atlantique  
Campus de Brest  
Brest - Nantes - Rennes

En individuel :

Réaliser un site Web en HTML5 référençant toutes les productions produites auparavant

# HTML5 ▶ ACTIVITÉ PRATIQUE COLLABORATIVE

HTML5

INT475 - Développement Web  
Front End pour les débutants

34

IMT Atlantique  
Campus de Brest  
Brest - Nantes - Rennes




Organisation  
Nouveautés  
CSS3 et les navigateurs  
Les nouveaux sélecteurs  
Largeur des boîtes  
Les transformations  
Outils en ligne

## CSS3 – SOMMAIRE

INT475 - Développement Web  
Front End pour les débutants

35






## CSS3 ► Organisation

- Par rapport à CSS 2.1
  - Ajout de nouvelles fonctionnalités
  - Redéfinition de certaines fonctionnalités
- CSS 3 est découpé en plusieurs modules catégorisant les propriétés CSS
  - ➔ éviter d'avoir une énorme norme à faire évoluer
- Suivre l'état d'avancement de la spécification CSS
  - <https://www.w3.org/Style/CSS/current-work>
- Code statut

Abréviation	Nom complet
FPWD	<a href="#">First Public Working Draft</a>
WD	<a href="#">Working Draft</a>
CR	<a href="#">Candidate Recommendation</a>
PR	<a href="#">Proposed Recommendation</a>
REC	<a href="#">Recommendation</a>
SPSD	<a href="#">Superseded Recommendation</a>

INT475 - Développement Web  
Front End pour les débutants

36



CSS

# CSS3 ▶ Nouveautés

Appliquer des images aux bordures

Ajouter plusieurs bordures

Créer des arrière-plans en dégradé, des images multiples

Supporter la transparence dans les couleurs et les éléments

Appliquer une ombre aux éléments (boîte, texte, etc.)

Créer des transformations, des transitions et des animations sur les propriétés et les éléments

Insérer des polices de caractères variées

Créer des sites Web adaptables aux différentes tailles d'écrans

...

INT475 - Développement Web  
Front End pour les débutants

37

CSS

# CSS3 et les navigateurs ▶ Les « préfixes vendeurs »

• Implémentation de nouvelles propriétés (éditeur de navigateur)

• Pour spécifier une mise en forme spécifique à un navigateur

• Si la spécification atteint le statut de **Recommandation** →

• Implémentation doit être supportée par les navigateurs

• Cesser d'utiliser les préfixes pour la propriété

• Exemple : créer des ongles arrondis

• `mySelector { -moz-border-radius : 10px }`

Préfixe	Moteur de rendu
-moz-	Gecko (Mozilla FireFox)
-webkit-	Webkit (Safari & Chrome)
-o-	Opera
-ms-	Microsoft Internet Explorer
-khtml-	KHTML dans plusieurs navigateurs sous Linux


INT475 - Développement Web  
Front End pour les débutants

38

## CSS3 ▶ Les nouveaux sélecteurs


- Les sélecteurs
  - Adjacent général : `x~y {...}` → appliqué aux éléments y qui suivent x même s'il y a d'autres éléments entre x et y
  - Attribut (enrichissement) :
    - `[a^="value"] {...}` → appliqué aux éléments dont l'attribut a commence exactement par value
    - `[a$="value"] {...}` → appliqué aux éléments dont l'attribut a fini exactement par value
    - `[a*="value"] {...}` → appliqué aux éléments dont l'attribut a contient au moins une fois value
  - Questions : Quelle est la signification des règles CSS suivantes?
    - `a[href$=".zip"]{ font-weight: bold;}`
      - Mettre en gras le texte des liens vers des fichiers ayant l'extension zip
    - `a[href^="mailto:"]{ background: url(personne.png) left center no-repeat;}`
      - Afficher en arrière plan l'image 'personne.png' à gauche, centrée et sans répétition pour les liens de type adresse mail

CSS



INT475 - Développement Web  
Front End pour les débutants


39



## CSS3 ▶ Les nouveaux sélecteurs


- Les pseudo-classes
  - Ancre : `:target {...}` → appliqué à l'ancrage que l'utilisateur a utilisée
    - Exemple : `a:target{ background-color: yellow;}`
  - Structure
    - Racine de la page : `:root {...}` → appliqué à HTML  
L'importance de `:root` est supérieure à celle de l'élément HTML
    - Dernier enfant : `:last-child {...}`
    - N<sup>ième</sup> enfant : `:nth-child(x) {...}` → appliqué à l'enfant dont l'ordre de fratrie est défini par x :
      - Un nombre positif → l'enfant à la nième position
      - `odd` → les enfants impairs
      - `even` → les enfants pairs
      - Calcul sous la forme `an+b` avec n un compteur qui démarre à 0 et s'incrémente par pas de 1. a et b sont des entiers relatifs

CSS



INT475 - Développement Web  
Front End pour les débutants

40



## CSS3 ▶ Les nouveaux sélecteurs

- Nième dernier enfant : `:nth-last-child(x){ ...}` → appliqué à l'enfant dont l'ordre de fratrie à partir de la fin est défini par x (idem que `nth-child`)
- Premier élément d'un type donné : `:first-of-type{ ...}`
- Dernier élément d'un type donné : `:last-of-type{ ...}`
  - Exemple : `li:first-of-type, li:last-of-type { background-color: gold; }` → appliqué au premier élément et dernier élément de la liste
- Nième premiers éléments d'un type donné : `:nth-of-type{ ...}`
- Nième derniers éléments d'un type donné : `:nth-last-of-type{ ...}`
- Élément seul sans frère : `:only-child{ ...}`
- Élément seul d'un type de donné sans frère : `:only-of-type{ ...}`
- Élément vide : `:empty{ ...}`
- Tous les éléments sauf celui spécifié : `:not(x){ ...}` → appliqué à tous les éléments sauf x
  - Exercice : Mettre en surbrillance tous les éléments de mise en forme (gras, italique, etc.) d'un paragraphe sauf les soulignés :  
`P>:not(u) { background-color: gold; }`

CSS

INT475 - Développement Web  
Front End pour les débutants

41

## CSS3 ▶ Largeur des boîtes

- La propriété `box-sizing` indique comment sont calculées la largeur et la hauteur des boîtes :
- Deux modes de calcul sont possibles:
  - `content-box` :
    - Valeur par défaut
    - Comportement identique à CSS2.1
    - Largeur d'affichage = `margin + border + padding + width`
  - `border-box` :
    - `border` et `padding` sont inclus dans le `width`
    - Largeur d'affichage = `margin + width`

CSS


INT475 - Développement Web  
Front End pour les débutants

42

## CSS3 ▶ Les transformations


- Un élément HTML peut subir des transformations de rotation, déplacement, déformation, de mise à l'échelle et de déplacement
- Pour appliquer une fonction de transformation sur l'élément auquel elle est rattachée il faut définir les propriétés
  - `transform-origin`: point de référence; c'est le point de départ pour le calcul des transformations
    - Par défaut: centre de l'élément
    - Accepte :
      - des %
      - des mots-clés: `left`, `center`, `right`, `top`, `center`, `bottom`
      - des valeurs exprimées en pixels
  - `transform`: permet de définir la (les) fonction(s) de transformation à appliquer:
    - `translate(v1px , v2px)` → déplacement sur l'axe x et y
    - `translateX(Vpx)` → déplacement sur l'axe x
    - `translateY(Vpx)` → déplacement sur l'axe y
    - `scale(V)` → mise à l'échelle
    - `rotate(V)` → rotation en degrés (deg) ou en radians (rad)
    - `skew(V1, V2)` → déformation en (deg) ou en radians (rad)
    - Etc.

CSS



INT475 - Développement Web  
Front End pour les débutants

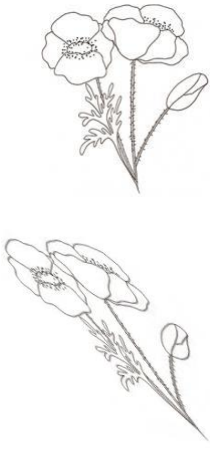
43




## CSS3 ▶ Les transformations ▶

### Exemple

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport"
content="width=device-width">
  <style>
    img{
      transform-origin: left;
      transform: rotate(10deg)
      translateX(30px) skewX(45deg);
    }
  </style>
</head>
<body>
  
</body>
</html>
```




CSS



INT475 - Développement Web  
Front End pour les débutants

44




## CSS3 ▶ Outils en ligne

### Page Publicité !


- Tester la reconnaissance de CSS par un navigateur  
<http://css3test.com/>
- Exemples des possibilités de CSS3 et de développement Web  
<http://lea.verou.me/projects/>
- Outils de génération de *script* CSS3
  - <https://css3gen.com/>
  - <https://css3generator.com/>
  - <http://westciv.com/>
  - <http://www.display-inline.fr/projects/css-gradient/>
  - <http://www.colorzilla.com/gradient-editor/>
  - <https://www.css3maker.com/>
  - <http://layerstyles.org/>

CSS



INT475 - Développement Web  
Front End pour les débutants


45



1. Réaliser une transition verticale de 250 px du haut vers le bas d'un paragraphe durant 5 secondes
2. Réaliser une animation d'une balle qui se déplace d'une position x à une position y à la moitié du temps d'animation et ensuite revient à sa position initiale
3. Intégrer ces pages au site précédent


## CSS3 ▶ ACTIVITÉS PRATIQUES

CSS



INT475 - Développement Web  
Front End pour les débutants

46






Un mot sur

RESPONSIVE WEB DESIGN

INT475 - Développement Web  
Front End pour les débutants

47





Problématique

Prolifération de terminaux ayant des écrans de


- différentes résolutions
- différentes tailles
- différents Formats

Terminaux cibles

- Smartphones
- Tablettes
- Ordinateurs fixes ou portables
- Les Télévisions
- Les projecteurs

INT475 - Développement Web  
Front End pour les débutants

48





# Objectifs

- Offrir une consultation confortable
- Consulter le même site à travers plusieurs équipements
- Disposer d'un même confort visuel quelque soit l'équipement
- Éviter de recourir aux défilements ou au zoom
- Éviter de dégrader l'expérience utilisateur
- Garantir la plasticité des interfaces

INT475 - Développement Web  
Front End pour les débutants

49

# Définition

- 1<sup>er</sup> apparition du terme : mai 2010 par Ethan Marcotte dans <https://alistapart.com/article/responsive-web-design>
- Acronyme: RWD
- Traduction de « *Responsive Web Design* » (En) :
  - **Site Web Adaptatif** (Fr): Site Web dont l'affichage s'adapte à différentes tailles d'écrans de terminaux grâce à la détection de leurs caractéristiques et à la sélection de la version appropriée du code Web parmi les différents codes par défaut  
*Office québécois de la langue française*
  - **Site Web Réactif** (Fr) : Site Web dont l'affichage s'adapte dynamiquement à différentes tailles d'écrans de terminaux grâce à des solutions techniques qui partent d'une version unique du code Web  
*Office québécois de la langue française*
- Dans l'usage les deux termes sont souvent confondus


INT475 - Développement Web  
Front End pour les débutants

50

## Solutions possibles


source: [www.alsacreations.com/article/lire/1615-cest-quoi-le-responsive-web-design.html](http://www.alsacreations.com/article/lire/1615-cest-quoi-le-responsive-web-design.html)

- Évolution des usages du Web par les *smartphones* et équipements connectés → priorité aux appareils connectés
- Trois façons de faire
  - Site dédié : concevoir et développer plusieurs entités différentes selon l'équipement visé : un site principal, un pour *smartphone*, un pour tablette, etc.  
Le serveur détecte le type d'équipement et renvoie vers une adresse dédiée (`monbile.mySite.com`)
  - Application native : développée spécifiquement pour un équipement (iOS, Android, Windows)
  - Version responsive : un seule site qui s'adapte à tout type d'équipement de manière transparente à l'utilisateur



INT475 - Développement Web  
Front End pour les débutants


51



## Solutions possibles


source: [www.alsacreations.com/article/lire/1615-cest-quoi-le-responsive-web-design.html](http://www.alsacreations.com/article/lire/1615-cest-quoi-le-responsive-web-design.html)

Solution	Avantages	Inconvénients
Site dédié	<ul style="list-style-type: none"><li>• affiner précisément la structure du site et ses contenus au terminal</li><li>• cibler et de s'adapter à des fonctionnalités variées (tactile)</li><li>• rapide</li></ul>	<ul style="list-style-type: none"><li>• contenu dupliqué</li><li>• maintenance de plusieurs versions de site et de plusieurs adresses web</li><li>• détection côté serveur du terminal souvent biaisée</li></ul>
Application native	<ul style="list-style-type: none"><li>• prise en charge facilitée de fonctionnalités natives (capteurs)</li><li>• installation possible</li><li>• disposer d'un "raccourci" sur le <i>smartphone</i></li></ul>	<ul style="list-style-type: none"><li>• développement spécifique dans plusieurs langages</li><li>• coût du développement, des licences, et de la maintenance</li><li>• contenu non indexable par un moteur de recherche</li><li>• mise à jour de l'application par action de l'utilisateur</li></ul>
Version responsive	<ul style="list-style-type: none"><li>• coûts et délais réduits</li><li>• maintenance de projet facilitée</li><li>• mise à jour transparente et déploiement multiplateformes</li></ul>	<ul style="list-style-type: none"><li>• bonnes connaissances techniques, et une veille technologique constante</li><li>• prévoir des tests nombreux et variés</li><li>• dépendant des limites ergonomiques et de performances des navigateurs web</li><li>• plus long (25% du temps supplémentaire)</li></ul>



INT475 - Développement Web  
Front End pour les débutants

52





# Comment faire du responsive ?


source: [www.salsacreation.com/article/lire/1615-cest-quoi-le-responsive-web-design.html](http://www.salsacreation.com/article/lire/1615-cest-quoi-le-responsive-web-design.html)

- Une grille fluide, où les largeurs des éléments de structure sont débarrassées des unités de pixels
- Des images, des médias et des contenus flexibles leur permettant de ne pas déborder de leur parent lorsque celui-ci se restreint
- Une adaptation de l'affichage au Viewport du terminal
- Des CSS3 Media Queries permettant d'appliquer différentes règles de styles CSS selon la taille, l'orientation ou le ratio du terminal
- Des adaptations conditionnelles (menu de navigation) côté client, basées sur JavaScript ou jQuery
- Une philosophie "Mobile First" et "Enrichissement progressif" facilitant l'accessibilité, la compatibilité et la performance des pages produites

INT475 - Développement Web  
Front End pour les débutants

53






Tester des feuilles de style gratuites sur le web et vérifier leur comportement responsive en utilisant le navigateur

- <https://www.free-css.com/>
- <https://designseer.com/>

## CSS3 ▶ ACTIVITÉ PRATIQUE

INT475 - Développement Web  
Front End pour les débutants

54



# Bibliographie

Ce cours est construit à partir des livres suivants

- HTML5 et CSS3 (Christophe AUBRY)
  - ISBN-13 : 978-2-409-00520-6
- CSS avancées : Vers HTML5 et CSS3 (Raphaël Goetter)
  - ISBN-13: 978-2212134056
- Et aussi plusieurs ressources sur le Web

