



Université Jendouba



Faculté des Sciences Juridiques
Économiques et de Gestion

Responsive Web Design

Manel Ben Sassi

Université de Jendouba, FSJSEG

bensassi.manel@gmail.com

Sommaire

1 Objectifs

2 Les règles du design

- Design ou Web design
- Ergonomie : Définition et Règles

3 Rappel : la feuille de style CSS

- Les aspects avancés du CSS
 - Les notations courtes et longues
 - Les sélecteurs CSS avancés
 - Les pseudo classe CSS
 - Les pseudo Elements
- CSS 3 ... les nouveautés
 - Les effets visuels
 - Les animations en CSS

4 Responsive : Définition

- Quelles solutions autres que Responsive Web Design
- Les avantages du web responsive design
- Les fondamentaux du web responsive
 - La balise Viewport
 - Les Media Queries

Quelles sont les objectifs de ce cours ?



Les objectifs ...

Acquérir la culture de l'adaptativité dans le web

- Comprendre le concept de responsive web design.
- Analyser et discuter l'importance de la programmation web adaptative
- Comprendre et appliquer les fondamentaux du responsive web à savoir la métabalise "viewport" et les "media queries"



Comment rendre un site plus attractif ?



Y a t-il des règles ou des normes pour le développement web ?



Ergonomie et design ... les 2 fondamentaux du développement web

Mais pourquoi ?



Il faut capter les visiteurs avec un design attrayant et ensuite les retenir en facilitant leurs navigations grâce à une **ergonomie bien pensée** !



Ergonomie et design ... les 2 fondamentaux du développement web

Vous dites Design ?



Par définition...

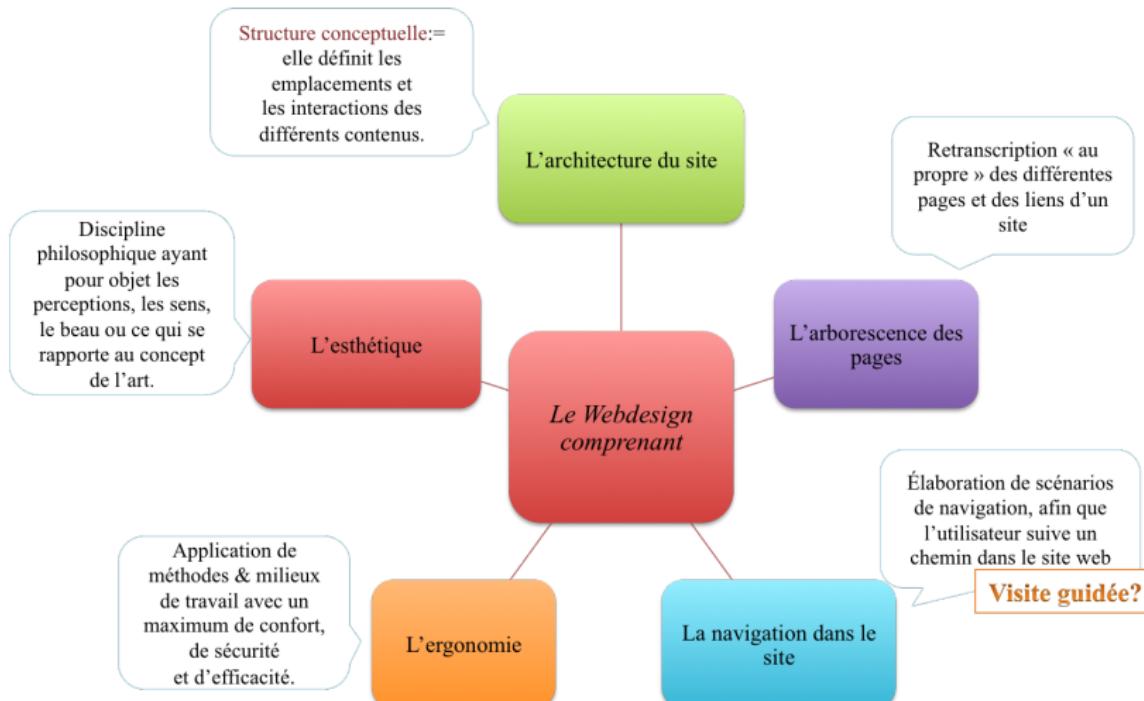
L'art de présenter de manière simple les choses les plus complexes.



Le web design requiert des compétences dans la programmation, le développement et l'intégration !

Design du Web ? Quels composants ?

Le Webdesign, c'est la conception d'une interface web



Ergonomie et design ... les 2 fondamentaux du développement web

Vous dites ERGONOMIE ?



Par définition...

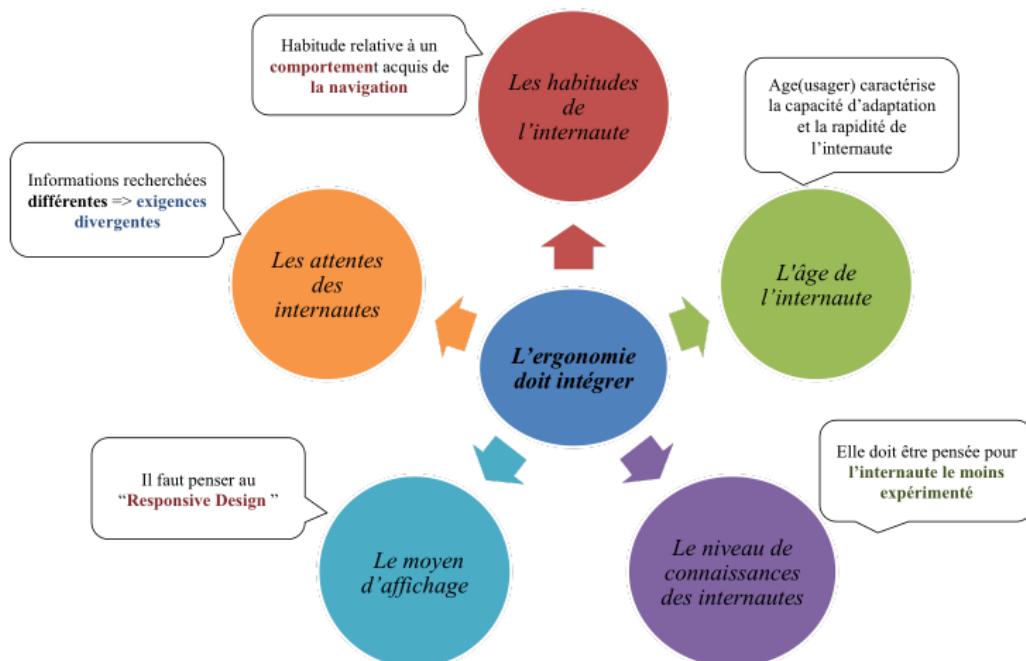
L'ergonomie d'un site Internet ⇒ Sa capacité à répondre efficacement aux attentes des internautes avec une "*navigation confortable*".

⇒ Le principal défi à relever ⇒ La diversité des profils des internautes.



Qu'est ce que l'ergonomie appliquée aux sites web ?

Les contraintes de l'ergonomie



Ergonomie et design ... les 2 fondamentaux du développement web

Possibilité de faire des tests utilisateurs et des audits d'accessibilité

Les principes de base à respecter avant ...

- # Afficher le nom et le logo en gros et bien en vue.
- # Éviter les éléments graphiques en filigrane (images en arrière-plan et texte au premier plan).
- # Éviter l'abondance de menus déroulants

Les commandements du web design

Règle n 1 : Mon site .. mon miroir

Créer du contenu à son propre image Quelle que soit votre activité, votre site doit refléter vos valeurs et véhiculer l'image de marque ⇒ 1. Utilisez vos couleurs + 2. Choisissez vos polices de caractères + 3. Créez un logo aligné vos aspirations esthétiques et **commerciales** mais aussi en fonction de votre public cible (Le langage visuel et textuel de HELLO KITTY ≠ de celui de SPIDERMAN)



Ciblez de bons mots clefs Comment ?



Les commandements du web design

Ciblez de bons mots clefs Comment ?



- ① Parlez la même langue que votre cible
- ② Ciblez des recherches populaires
- ③ Vérifiez la concurrence



Mettre en avant des témoignages clients

Les commandements du web design

Attention ! Retenez par convention ...

Contraste et lisibilité du texte Par exemple : pas du texte jaune sur une fond gris ! Taille de police confortable

Texte souligné Ne soulignez pas le texte ⇒ **Un lien** ! Pour faire ressortir le texte, il vaut mieux changer de couleur et/ou la taille

Segmenter le texte ⇒ Utiliser les titres et les sous-titres.

Peu d'animation ⇒ Éviter les animations pour les éléments essentiels (logo, titre, signature) ⇒ assimilé à de la publicité.



Les commandements du web design

Règle 2 ... Attirer, Engager et Convaincre

Partager du contenu sur les réseaux sociaux Objectif : Créer une relation de confiance pour mieux vendre son image ⇒ Inciter les clients à vous suivre sur les réseaux sociaux



Objectif : le faire revenir sur le site

Récupérer l'email de votre prospect

- ① Se créer une base de données prospects
- ② Offrir quelques choses en échange
- ③ Envoyer de l'information utile (PAS DE SPAMS)

Les commandements du web design

Règle ... Un Bon formulaire de contact

Contactez La Maison du Management

Nom*
Votre nom complet

Email*
Votre email 

Téléphone
Options

Message*
Votre message



Figure – Un bon formulaire de contact

Les commandements du web design

Règle 3 ... Un design professionnel et moderne

"Ne sous-estimez pas le pouvoir de la première impression"



Figure – The power of design

Les commandements du web design

Règle 3 ... Un design professionnel et moderne

"Booster le référencement naturel, mieux valoriser les produits"



Figure – Before and after

Les commandements du web design

Règle 3 ... Un design professionnel et moderne

"Une navigation agréable + facilité pour trouver l'information ► Meilleur satisfaction "

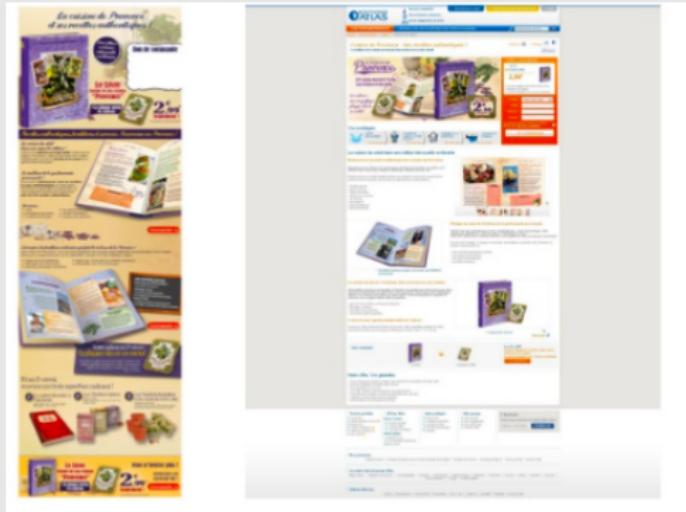


Figure – Before and after

Les commandements du web design

Règle 3 ... Un design professionnel et moderne

La fiche produit de bien high tech

Travail étudiant e-commerce sur une FP Amazon : Avant

This screenshot shows an older version of an Amazon product page. At the top, there's a navigation bar with links like "amazon.fr", "Débutez avec Firefox", "À la une", "CFP Mat", "FrenchToneDE", and "Aide". Below the navigation is a search bar and a "Rechercher Photo, Image, Son et GPS" button. The main content area features a large image of the laptop, followed by its name: "HP Pavilion dv7-6b99ef". Below the name are sections for "Prix" (1.442,00), "Voir les offres de ces vendeurs", and "Livraison à partir de EUR 1.442,00". A prominent blue button labeled "Ajouter au panier" is located at the bottom right of this section. To the right of the main content, there's a sidebar with "Découvrez le Guide des meilleurs cadeaux" and a "Vidéo conseil". At the very bottom, there's a footer with links like "Retrouvez-nous sur Facebook, Twitter, LinkedIn, YouTube, Google+, et plus encore" and a note about "Les offres sont en cours dans le monde entier mais elles peuvent être limitées par pays".

This screenshot shows a modernized version of the same product page. The layout is cleaner and more organized. The main product image is larger and centered. The product title "HP Pavilion dv7-6b99ef [H1]" is displayed prominently above the price. The price "1 article neuf : 675,65 €" is shown in a large red font. Below the price, there's a "En stock" button. To the right, technical specifications are listed: "3,2 GHz", "Intel Core i7-7700HQ", "4 Go", "Disque dur 1 To", and "Vidéoprojecteur intégré". A "Démonstration" button is also present. The overall design is more professional and user-friendly.

Figure – Before and After

Les commandements du web design

Attention ! Retenez par convention ...

Largeur des colonnes Evitez les colonnes trop étroites ou trop larges (entre 10 et 15 cm) ⇒ L'œil se fatigue.

Evitez les gadgets Les compteurs, animations flash, etc ⇒ **A éviter** ça fatigue les yeux et détourne l'attention du contenu !

Légèreté et la rapidité Un affichage rapide et léger

Simplicité des pages et des menus. La norme consiste à placer :

- ① Les menus et chapitres principaux en haut de la page,
- ② Les sous menus ou sous chapitres à gauche de la page,
- ③ Les références et sources en bas de la page,
- ④ Les renvois, les publicités, à droite de la page.



Les commandements du web design

Règle 4 ...

Contrôler la navigation La navigation sur le site est probablement le facteur qui influence le plus fortement l'expérience des usagers. Un site sur lequel il est agréable de naviguer verra ses visiteurs prolonger leur séjour sur ses pages.

Améliorer l'expérience de l'utilisateur

"Accompagner pour simplifier la tâche"

- ▶ L'ergonomie doit faciliter la navigation
- ▶ Les textes doivent donner envie
- ▶ Trop de texte tue le texte
- ▶ Évitez l'animation ...



Retenez ...

La communication est essentielle pour une bonne expérience utilisateur.
Le plus beau et efficace des environnements virtuels doit **parler** à
l'utilisateur afin qu'il puisse en profiter



A vous de jouer ...

Pourquoi dois-je avoir une feuille de style ?



Le CSS, Pour quelle raison ?

Question

*"HTML me permet aussi de rajouter des couleurs de mettre en gras ...
Alors pourquoi se compliquer la vie?"*

Le CSS, Pour quelle raison ?

Question

*"HTML me permet aussi de rajouter des couleurs de mettre en gras ...
Alors pourquoi se compliquer la vie?"*

Objectifs

- **Séparer** le fond (le contenu) de la forme (la présentation ou la mise en forme)...
- Avoir un code facilement compréhensible ⇒ maintenable, évolutif et réutilisable

On se pose la question ...

Je ne vois pas pourquoi ? je comprendrais toujours mon code... ?

Le CSS, Pour quelle raison ?

L'inconvénient d'avoir un code fusionné



Le CSS, Pour quelle raison ?

Séparation du fond et de la forme



contenu et structure logique



mise en forme

- HTML simple et facile à maintenir
- séparation de métiers (graphiste / dev. web)
- HTML compréhensible par
 - malvoyants
 - moteurs de recherche
 - autres logiciels
- présentation facile à changer


`<p id="p1">Pingouins</p>`
`<p>Les pingouins sont`
`des oiseaux...</p>`
`#p1{ font-size:200%; }`

titre???


`<h1>Pingouins</h1>`
`<p>Les pingouins sont`
`des oiseaux...</p>`

Le CSS, Pour quelle raison ?

Séparation du fond et de la forme

fond

(contenu, sens, structure)

HTML

```
<h1>Bonjour</h1>
```



« Ce texte ("Bonjour")
est un titre important »



mal voyant
moteur de recherche
autres « robots »

forme

(présentation, apparence)

CSS

```
h1  
{  
    font-size : 200%;  
}
```



« Les titres importants sont
affichés en grand »

Où placer une feuille de style ?



Où placer le code CSS ?

Deux endroits :

- Dans une balise **style** qui est elle-même placée dans la balise **head** de notre page HTML. C'est pratique car on a ainsi le code HTML et le code CSS dans la même page. C'est ce que nous ferons pour nos premiers essais.
- Dans un autre fichier dédié au style avec une extension **.css**. On utilise dans ce cas une balise link placé elle aussi dans la balise **head** de notre page HTML pour indiquer l'emplacement du fichier CSS.

Où placer le code CSS ?

Exemple de page HTML avec le code CSS intégré via la balise style :

```
<!DOCTYPE html>
<html>
    <head>
        <title>Librairie ABCD : rencontres</title>
        <link rel="icon" type="image/png" href="icone.png">
        <style>
            Code CSS...
        </style>
    </head>
    <body>
        <article>
            <h1>Rencontre avec Jean-Baptiste Botul</h1>
            <p>Rendez-vous le 14 octobre à partir de 15h.</p>
        </article>
    </body>
</html>
```

Où placer le code CSS ?

Exemple de page HTML avec le code CSS intégré via un fichier externe :

```
<head>
...
<link rel="stylesheet" media="screen" type="text/css"
title="Titre du css" href="nom_fichier.css" />
...
</head>
```

Principe du CSS ...

```
balise {  
propriete : valeur;  
propriete : valeur;  
}  
}
```

- Balise : Nom de la balise dont on modifie l'apparence (ex : < em >, < h1 >,etc)
- proprieté : catégorie d'effet de style (color, font-size)
- valeur : valeur associée à la propriété (red, blue).

```
em /*Les mots mis en valeur sont*/  
{  
    color : green; /*écrits en vert*/  
    font-size : 30px; /*avec une taille de 30 pixels*/  
}
```

Principe du CSS ...

Appliquer des règles à plusieurs balises

```
balise1, balise2, ..., BaliseN {  
propriete : valeur ;  
propriete : valeur ;  
}
```

- Balise : Nom de la (ou les) balise(s) dont on modifie l'apparence (ex : < em >, < h1 >, etc)
- propriété : catégorie d'effet de style (color, font-size)
- valeur : valeur associée à la propriété (red, blue).

```
<!DOCTYPE html>  
<html>  
    <head>  
        <title>Librairie ABCD : rencontres</title>  
        <link rel="icon" type="image/png" href="icone.png">  
        <style>  
            Code CSS...  
        </style>
```

Principe du CSS ...

Imbrication de balises

```
balise1 balise2 balise3 {  
propriete : valeur;  
}
```

- Quand on est dans la balise "*balise3*" contenue dans la balise "*balise2*", elle-même contenue ...

```
h1 em  
{  
    font-weight : bold;  
    color : gray;  
}
```

Attributs class et id

Comment faire pour que seulement quelques mots mis en valeur (mais pas tous !) soient en vert et de taille 30 pixels ?



Utilisation des attributs class et id.

- Permettent de définir un style personnalisé
- Peuvent être mis sur n'importe quelle balise
- Permettent de ne modifier que certains mots, titres, ...

Utilisation de l'attribut class

1. Modification du fichier html

On ajoute l'attribut "class" dans les balises que l'on souhaite modifier.

```
<h1 class="nom1"></h1> <p class="nom2"></p>
```

2. Modification du fichier CSS

On définit les valeurs des propriétés qui vont être appliquées aux balises ayant l'attribut class.

```
.nomdelaclasse {  
propriete : valeur;  
}
```

Contrairement aux balises, il faut ajouter un point devant le nom de la classe

Utilisation de l'attribut id

- L'attribut id s'utilise (quasiment) comme l'attribut class
- Définition dans le css avec # (et non un point)

Définition d'un identifiant dans le css

On définit les valeurs des propriétés qui vont être appliquées aux balises ayant l'attribut class.

```
#nomID {  
propriete : valeur;  
}
```

Attention : id ne peut être utilisé qu'une seule fois !

```
<p> j'utilise plusieurs fois la balise  
<em id = "enGras"> em </em> pour mettre plein de  
<em id="enRouge> mots </em> valeur mais je n'utilise  
pas les bonnes <em id="enGras"> balises!!! </em> </p>
```

Différence entre class et id

- **L'identifiant (id)**
 - utilisé pour référencer un élément
 - utilisé pour la disposition des éléments de la page (entête, menu, pied de page, ...)
- **L'attribut class**
 - utilisé plusieurs fois (pas de référence à un objet précis)
 - utilisé dans les autres cas.

Balises universelles

Comment modifier l'apparence d'un mot qui n'est contenu dans aucune balise

Utilisation des balises universelles

- Deux types de balises universelles :
 - Balise de type **inline** : ...
 - Balise de type **block** : <div> ... </div>
- Balises qui "ne servent à rien"
- Permettent de rajouter des **classe** modifiant la partie comprise entre ces balises.

Exemple

```
<span class="grasRouge"> Bonjour </span>
```

Les aspects avancés du CSS



Notations Long Hand Vs Short Hand

Pour définir une bordure, nous avons deux façons de faire :

Notation Long Hand Exemple : pour définir une bordure, nous avons utiliser les trois propriétés CSS **border-width**, **border-style** et **border-color**

Notation Short Hand Pour le même exemple, nous pouvons utiliser la propriété **border** avec les valeurs des trois propriétés précédentes

```
1 /* Notre titre h1 et notre paragraphe recevrons exactement la même mise en forme CSS */
2 #id1{
3     font-style: oblique;
4     font-weight: normal;
5     font-size: 16px;
6     line-height: 24px;
7     font-family: "Times New Roman", Times, serif;
8 }
9 #id2{
10    font: oblique normal 16px/24px "Times New Roman", Times, serif;
11 }
12 div {
13     background-color: lightblue;
14 }
```



D'autres exemples

- Cette règle s'applique aussi sur les propriétés des modèles de boîtes (**padding**, **border**, **margin**) et du Background (**background**)

Sélecteurs CSS Avancés

Etoile (*) sélecteur universel permet de sélectionner tous les éléments HTML d'une page d'un coup

```
HTML ▾
<!DOCTYPE html>
<html>
  <head>
    <title>Sélecteurs avancés </title>
    <meta charset= "utf-8">
  </head>

  <body>
    <h1>Les Sélecteurs avancés du CSS</h1>
    <p>Un paragraphe</p>
    <p>Un <strong>paragraphe important</strong></p>
  </body>
</html>
```

The screenshot shows a CSS editor interface with three main sections: HTML, CSS, and Output.

- HTML:** Displays the provided HTML code.
- CSS:** Displays the CSS rule `* { border: 1px solid black; padding: 5px; margin: 5px; }`. The asterisk (*) selector is highlighted in pink.
- Output:** Shows the rendered HTML with the applied CSS styles. The title "Les Sélecteurs avancés du CSS" has a black border, padding, and margin. The two paragraphs below it also have these styles applied.

Sélecteurs CSS Avancés

Imbrication (A B) Sélectionner un élément B contenu dans un élément A (même dans une classe).

The screenshot shows a browser's developer tools interface with three main sections: HTML, CSS, and Output.

HTML:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Sélecteurs avancés</title>
    <meta charset= "utf-8">
  </head>

  <body>
    <h1>Sélecteurs CSS avancés</h1>

    <div class="test">
      <p>Un premier paragraphe dans un div<br/><p>Un <strong>paragraphe important</strong></p>
    </div>

    <div>
      <p>Un paragraphe dans un deuxième div</p>
    </div>
    <p>Un paragraphe hors div</p>
  </body>
</html>
```

CSS:

```
.test p{
  font-size: 20px;
  color: Magenta;
}
```

Output:

Run with JS Auto-run JS

Sélecteurs CSS avancés

Un premier paragraphe dans un div

Un paragraphe important

Un paragraphe dans un deuxième div

Un paragraphe hors div

Sélecteurs CSS Avancés

A + B Sélectionner tous les éléments B de même niveau et suivant immédiatement les éléments A

HTML ▾

```
<!DOCTYPE html>
<html>
  <head>
    <title>Sélecteurs avancés</title>
    <meta charset= "utf-8">
  </head>

  <body>
    <h1>Sélecteurs CSS avancés</h1>

    <div class="test">
      <p>Un premier paragraphe dans un div</p>
      <p>Un <strong>paragraphe important</strong></p>
    </div>

    <div>
      <p>Un paragraphe dans un deuxième div</p>
    </div>
    <p>Un paragraphe hors div</p>
  </body>
</html>
```

CSS ▾

```
div + p{
  background-color: orange;
}
```

Output Run with JS Auto-run JS

Sélecteurs CSS avancés

Un premier paragraphe dans un div

Un **paragraphe important**

Un paragraphe dans un deuxième div

Un paragraphe hors div

Sélecteurs CSS Avancés

A > B Sélectionner tous les éléments B qui sont des éléments enfants directs de A.

```
HTML ▾
<!DOCTYPE html>
<html>
  <head>
    <title>Sélecteurs avancés</title>
    <meta charset= "utf-8">
  </head>

  <body>
    <div class="page-conteneur">
      <h1>Sélecteurs CSS avancés</h1>

      <div class="test">
        <p>Un premier paragraphe dans un div
        <p>Un <strong>paragraphe important</strong>
      </div>

      <div>
        <p>Un paragraphe dans un deuxième div
      </div>
      <p>Un paragraphe hors div</p>
    </div>
    <div>
      je suis une 2eme division
    </div>
  </body>
</html>
```

```
CSS ▾
body > div{
  border: 1px solid black;
}
```

Run with JS Auto-run JS

Sélecteurs CSS avancés

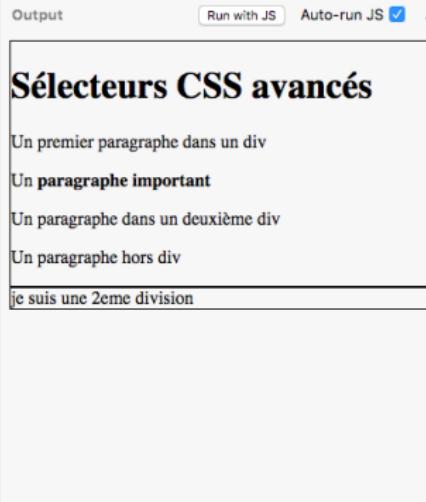
Un premier paragraphe dans un div

Un **paragraphe important**

Un paragraphe dans un deuxième div

Un paragraphe hors div

je suis une 2eme division



Les pseudo classes CSS

- :link** permet de styliser un lien non visité ;
- :visited** permet de styliser un lien une fois celui-ci visité (parfois un style ignoré par les navigateurs pour des raisons de sécurité)
- :hover** permet de changer l'aspect d'un élément lorsque les visiteurs poseront leur curseur dessus ;
- :active** permet de modifier l'aspect d'un lien lors du clic.

Les pseudo Classes CSS

:first-child sélectionner le premier élément enfant HTML d'un certain type par rapport à un élément parent.

:last-child sélectionner le dernier élément enfant HTML d'un certain type par rapport à un élément parent.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Pseudo classes</title>
    <meta charset= "utf-8">
  </head>
  <body>
    <h1>Pseudo classes CSS</h1>
    <div class="test">
      <p>Un premier paragraphe</p>
      <p>Un deuxième paragraphe</p>
      <p>Un troisième paragraphe</p>
    </div>
    <p>Un paragraphe hors div</p>
  </body>
</html>
```

```
p:first-child{
  color: orange;
}
p:last-child{
  color: green;
}
```

Pseudo classes CSS

Un premier paragraphe

Un deuxième paragraphe

Un troisième paragraphe

Un paragraphe hors div

Les Pseudo Classes CSS

:nth-child permet de cibler certains enfants d'un élément HTML. Nous mettons entre les parenthèses () soit un nombre, soit un mot clef. (Exemple paire ou impaire)

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title>Pseudo classes CSS</title>
    <meta charset= "utf-8">
  </head>

  <body>
    <h1>Titre de niveau 1</h1>
    <h2>Titre de niveau 2</h2>
    <p>Un premier paragraphe</p>
    <p>Un deuxième paragraphe</p>

    <div>
      <h2>Un autre titre de niveau 2</h2>
      <p>Un troisième paragraphe</p>
      <p>Un quatrième paragraphe</p>
    </div>
  </body>
</html>
```

CSS

```
p:nth-child(even){
  color: purple;
}

h2:nth-child(2){
  color: green;
}
```

Output

Run with JS

Titre de niveau 1

Titre de niveau 2

Un premier paragraphe

Un deuxième paragraphe

Un autre titre de niveau 2

Un troisième paragraphe

Un quatrième paragraphe

Les Pseudo Elements CSS

Les pseudo éléments

- Permettent de modifier l'apparence d'une partie seulement d'un ou de plusieurs éléments HTML, ou encore d'ajouter du contenu au début ou à la fin d'un certain élément HTML.
- Sont reconnaissables en CSS à leur écriture. Ils commencent par « : : » (un double deux-points).

: **:first-letter** permet de Ne sélectionner Que la première lettre d'un texte contenu dans un élément ;

: **:first-line** permet de Ne sélectionner Que la première ligne d'un texte contenu dans un élément ;

: **:selection** permet de sélectionner la partie d'un élément sélectionnée (par un double clic) par l'utilisateur ;

: **:before** permet d'insérer du contenu au début d'un élément HTML ;

: **:after** permet d'insérer du contenu à la fin d'un élément HTML.

Les Pseudo Elements CSS

Exemple

The screenshot shows a code editor with three panes. The left pane displays the HTML code:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Pseudo éléments</title>
    <meta charset= "utf-8">
  </head>

  <body>
    <h1>Pseudo éléments CSS</h1>

    <div>
      <p>Un premier paragraphe</p>
      <p>Ceci est un deuxième paragraphe</p>
      <p>Et voilà un troisième paragraphe</p>
    </div>
    <p>Un paragraphe hors div</p>
  </body>
</html>
```

The middle pane displays the CSS code:

```
p::first-letter{
  font-size: 30px;
  color: #A92;
}
```

The right pane shows the output of the code, displaying the text "Pseudo éléments CSS" with the first letter of each word in large, bold, orange font.

Output

Pseudo éléments CSS

U n premier paragraphe

C eci est un deuxième paragraphe

E t voilà un troisième paragraphe

U n paragraphe hors div

Figure – La lettrine

Nb : Pour plus d'exemples, voir l'annexe

Les nouveautés apportés au CSS3 ?



Les nouveautés- Quoi de neuf ?

Les nouveautés sont divisées en trois grandes parties :

- Les effets visuels
- Les sélecteurs
- Les nouveaux outils pratiques

Il ne s'agit pas de l'intégralité des nouveautés, mais ça vous permettra d'avoir un très bon aperçu global.

Les effets visuels ...



Effets visuels

Border radius : Arrondit les bords d' un bloc



Transition : Passe d'un état à un autre d'un élément avec par une transition animée



Effets visuels

Box-shadow : Applique une ombre sur un bloc :



Text-shadow : Applique une ombre sur un texte :

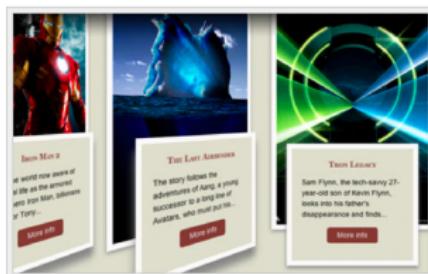
Css3 Text Shadow

Effets visuels

Transform : Déplace, déforme, ou effectue la rotation d'un élément



3D- Transform : Effectue des transformations en 3D



Que veut dire Responsive ?



Responsive ...

Le terme Responsive

- ▶ Fournir "Réponses ou une adaptativité rapide et positive" aux différents usagers (profils)
- ▶ Le "**Responsive Web Design**" fait référence à : "*création d'un ou plusieurs catalogue(s) de patrons de modèles de multi-dispositifs*".



Le web Responsive Design ...

- ▶ "Responsive Web design" terme introduit par "*Ethan Marcotte*" en 2010
- ▶ Le *Responsive Web Design* est **une approche de conception Web** qui vise à l'élaboration de sites offrant une expérience de lecture et de navigation optimales pour l'utilisateur quelle que soit sa gamme d'appareil (téléphones mobiles, tablettes, liseuses, moniteurs d'ordinateur de bureau).



Améliorer l'expérience de l'utilisateur

- ▶ Une expérience utilisateur "**Responsive**" réussie implique un minimum de redimensionnement (zoom), de recadrage, et de défilements multidirectionnels de pages.

Pourquoi devrons nous développer *Responsive* ? d'autres Alternatives ?



Site dédié, application ou responsive ?

3 manières de circuler et diffuser l'information :

1. Site dédié
2. Application native
3. Version responsive de site web



1^{re} solution : Site dédié

Analysons ces trois solutions ! !

- Un projet de **site dédié** consiste à concevoir deux ou plusieurs entités différentes selon le dispositif visé : un site principal, un site pour smartphones, un site pour tablettes, etc.

Avantages

- ① La possibilité d'affiner précisément la structure du site et ses contenus en regard du périphérique utilisé
- ② La possibilité de cibler et de s'adapter à des fonctionnalités variées
- ③ Peut être une alternative rapide, en attendant une refonte complète (et responsive) de son site web

1^{re} solution : Site dédié

Inconvénients

- ① Le contenu dupliqué ("duplicate content")
- ② La maintenance de plusieurs versions de site et de plusieurs adresses web (moins facilement indexables par un moteur de recherche)



2^eme solution : Une application native

- Une application native est un produit développé spécifiquement dans divers "**langages**" (iOS, Android, Windows Phone) et qui se télécharge et se référence au sein d'un "**Store**" (AppStore, Google Play, Windows store).

Avantages

- ① La prise en charge facilitée de fonctionnalités natives (touch, accéléromètre, notifications, GPS, etc.)
- ② Une installation possible sur son périphérique
- ③ Une totale "**acclimatation**" au périphérique (ergonomie, performances, densité de pixels)
- ④ La présence de sa marque sur l'AppStore (pour ne citer que lui) et de pouvoir disposer d'un "**raccourci**" sur le smartphone de l'utilisateur

2^eme solution : Une application native

Inconvénients

- ① Un développement spécifique dans plusieurs langages (propres à iOS, Android, Windows Phone, etc.)
- ② Le coût du développement, des licences, et de la maintenance pour chaque système d'exploitation
- ③ Un contenu non indexable par un moteur de recherche web classique
- ④ La mise à jour de l'application nécessite une action de l'utilisateur



3^{eme} solution : Une version responsive

- Des centaines de tailles et formats d'écrans différents se connectent à chaque instant, la méthode du Responsive Web design apparaît comme la "solution de facilité" en vertu de son objectif principal



S'adapter à tout type d'appareil de manière transparente pour l'utilisateur



Les avantages du Responsive web designing ?



3^eme solution : Une version responsive

Avantages

- ① Des coûts et des délais généralement inférieurs aux techniques citées précédemment
- ② Une maintenance de projet facilitée (une seule feuille de style, un seul fichier HTML, etc.)
- ③ Une mise à jour transparente et un déploiement multiplateformes
- ④ Le Responsive peut être envisagé après la conception initiale du site (même si ce n'est pas l'idéal)

Mais également

- Depuis quelques temps est que Google met en avant les sites "adaptés au mobile" au sein de ses résultats de recherche

3^eme solution : Une version responsive

Avantages



3^eme solution : Une version responsive

Inconvénients

- De bonnes connaissances techniques, et une veille technologique constante, sont indispensables
- Il est nécessaire de prévoir des tests nombreux et variés tout au long du projet ("device labs", simulateurs)
- Il est difficile de contourner les limites ergonomiques et de performances des navigateurs web
- Faire du responsive, c'est coûteux en termes de temps (plus de temps pour le développement)

Web Responsive

L'essentiel pour créer votre 1^{er} site adaptatif



One site for every screen.
Small + Medium + Large

- CSS3 et les Media queries
- Les méta tags et les viewport
- Grid System
- Frameworks

Comment procéder pour développer responsive ?



Les fondamentaux du web responsive design

Trois points essentiels doivent être mis en place :

- ① Une grille d'affichage **flexible** ⇒ Soit un gabarit (affichage globale de la page) qui ne dépende pas d'une résolution minimale, maximale ;
- ② **Des médias flexibles** ⇒ mes images, vidéos, etc, ne débordent pas du cadre de notre grille d'affichage ;
- ③ **un ensemble de règles CSS basé sur Media Queries** : le principe est de mettre des conditions sur l'affichage afin **d'afficher/masquer, voire changer le rendu de notre application Web.**

Les fondamentaux du web responsive design

Viewport

- Désigne schématiquement la surface de la fenêtre du navigateur.
- La notion de **viewport** sur un appareil mobile est différente de celle sur un écran de bureau ⇒ le navigateur mobile ne dispose pas de "*fenêtre*" réelle, ni de barres de défilement !!



Faut-il connaître toutes les différentes surfaces ?

Les fondamentaux du web responsive design

Les différentes surfaces d'un mobile

La surface physique C'est la résolution de l'écran telle quelle est définie par le constructeur.

La surface utilisable En pixels CSS : également appelée `device-width` (`device-height`) ou `screen.width` (`screen.height`).



Le hic est que cette surface ne correspond pas toujours à la surface physique, notamment pour les mobiles dits "Retina" ou haute définition

Les fondamentaux du web responsive design

Il faut savoir que ...

- Par défaut, la **taille du viewport** d'un terminal mobile ne correspond ni à la taille de son écran réelle ni celle en "pixels CSS".
- Elle est généralement bien supérieure à la surface physique, afin de pouvoir y "*caler*" n'importe quelle page web en lui affectant un niveau de **(dé)zoom**.

quelques valeurs par défauts

- *Android 1, 2 et 3 : 800px*
- *Android 4 : 980px*
- *Opera mini : 850px*
- *Opera mobile : 980px*
- *Safari mobile : 980px*
- *Internet Explorer mobile : 1024px*

Les fondamentaux du web responsive design

Compte tenu de ces différentes surfaces, les pages web s'affichent par défaut de manière à ce que toute la surface entre dans celle de l'écran.



Ce niveau de zoom initial correspond à une simple division mathématique de `device-width/viewport`

Les fondamentaux du web responsive design

La balise <meta> viewport

- ▷ La balise <meta> de HTML permet de contourner ce zoom rendant le contenu illisible !
- ▷ Les attributs de cette balise sont **name** et **content** qui permet de fixer la largeur de **viewport** à la valeur souhaitée, voire de l'adapter automatiquement à la valeur de **device-width** du terminal.

`<meta name="viewport" content="width=device-width, initial-scale=1.0">`

Les fondamentaux du web responsive design

La balise <meta> autorise d'autres valeurs utilisables dans vos projets :

width largeur de fenêtre viewport (par exemple
width="device-width")

height hauteur de fenêtre viewport (par exemple
height="device-height")

initial-scale niveau de zoom initial (par exemple initial-scale="1.0")

minimum-scale niveau de zoom minimal (par exemple
minimum-scale="0.5")

maximum-scale niveau de zoom maximal (par exemple
maximum-scale="3.0"). Attention, la valeur "1.0" interdit le
zoom et peut rendre vos pages inaccessibles

user-scalable possibilité à l'utilisateur de zoomer (par exemple
user-scalable="yes"). Attention, la valeur "no" interdit le
zoom et peut rendre vos pages inaccessibles

Les fondamentaux du web responsive design

Instructions d'usage des feuilles de style

Les valeurs en pourcentage

- Définissez une taille égale à 100% pour votre élément body, puis ensuite **une largeur** de 50% pour un div enfant direct du body, la taille du div sera toujours égale à 50% de celle du body.
- Pareil pour les images et les vidéos

```
Body{  
    width: 100%;  
}  
Div{  
    width: 50%;  
    border: 1px solid black;  
}
```

Les Media Queries ? on entend parler ?



Media Queries

- ① **Media Queries** = spécification CSS3 qui définit les techniques pour l'application de feuilles de styles en fonction des périphériques de consultation utilisés pour du HTML
- ② Favoriser les bonnes pratiques : **séparer le contenu de la forme**

Intérêt

Pouvoir satisfaire des contraintes de dimensions, de résolutions et d'autres critères variés pour améliorer l'apparence graphique et la lisibilité

Approche historique des media queries

- Avec CSS2 et HTML4, il était déjà possible de spécifier un média de destination pour l'application d'une ou plusieurs feuilles de style.
- **Comment ?** A travers la balise `<link>` qui est dupliquée pour autant de feuilles de style que nécessaire, et comporte un attribut `media` précisant le contexte dans lequel les styles doivent être appliqués

Exemple ?

Media queries

Exemple

```
1 <!doctype html>
2 <head>
3 <meta charset="utf-8">
4 <title>Media Queries !</title>
5 <link rel="stylesheet" media="screen" href="screen.css" type="text/css" />
6 <link rel="stylesheet" media="print" href="print.css" type="text/css" />
7 </head>
8 <body> ...
9 </body>
10 </html>
```

Media queries

Les **Media Queries** permettent donc de cibler :

- Le type de média
- La taille de l'écran
- La taille de la fenêtre
- La résolution
- Le nombre de couleurs
- L'orientation
- ...



L'attribut magique "media"

L'attribut media peut prendre (depuis CSS2) les valeurs suivantes :

Screen : Écrans

Handheld : Périphériques mobiles ou de petite taille

Print : Impression

Aural : (CSS 2.0) / speech (CSS 2.1) Synthèses vocales

Braille Plages braille

Embossed : Imprimantes braille

L'attribut magique "media"

L'attribut **media** peut prendre (depuis CSS2) les valeurs suivantes :

Projection : Projecteurs (ou présentations avec slides)

tty Terminal (police à pas fixe)

tv Téléviseur

all Tous les précédents

L'attribut magique "media"

Ces directives peuvent parfaitement être intégrées au sein même d'une feuille de style grâce à une règle @media suivie directement du type. La syntaxe sera alors légèrement différente :

```
@media print {  
    #menu, #footer, aside {  
        display:none;  
    }  
    body {  
        font-size:120%;  
        color:black;  
    }  
}
```

Syntaxe des Media Queries CSS3

- Requêtes des médias : c'est une combinaison multiples de plusieurs critères.
- La valeur de la requête : c'est une expression dont la valeur est toujours vraie ou fausse.
- Il suffit d'associer les différentes déclarations possibles avec un opérateur logique pour définir l'ensemble des conditions à réunir pour l'application des styles compris dans le bloc adjacent.
- Les opérateurs logiques peuvent être :
 - and "et",
 - only "uniquement"
 - not "non"
 - Pour obtenir le **OU logique** , il suffit d'énumérer les médias séparés par ", " .

Syntaxe des Media Queries CSS3

Règles générales

- On combine ensemble un type de média (screen, all...) et une expression grâce à and, bien qu'une expression seule puisse être utilisée. L'expression est toujours écrite entre parenthèses.

“

```
<link rel="stylesheet" media="screen and (max-width: 640px)"  
      href="smallscreen.css" type="text/css" />
```

”

“

```
@media screen and (max-width: 640px) {  
    .bloc {  
        display: block;  
        clear: both; }  
}
```

”

Syntaxe des Media Queries CSS3

Exemple 2

```
“  
@media screen and (min-width: 200px) and (max-width: 640px) {  
  
.bloc {  
  
display:block;  
  
clear:both; }  
  
}  
”
```

Les fonctionnalités (les critères) associées media (1)

La plupart des critères (ou fonctionnalités) peuvent être préfixés par **min-** et **max-** lorsqu'elles acceptent des valeurs numériques pour définir des valeurs minimales ou maximales à respecter.

- **Color support** : de la couleur (bits/pixel)
- **color-index** : périphérique utilisant une table de couleurs indexées
- **device-aspect-ratio** ratio du périphérique de sortie (par exemple 16/9)
- **aspect-ratio** ratio de la zone d'affichage
- **device-height** dimension en hauteur du périphérique
- **device-width** dimension en largeur du périphérique

Les fonctionnalités (les critères) associées media (2)

La plupart des critères (ou fonctionnalités) peuvent être préfixés par **min**- et **max**- lorsqu'elles acceptent des valeurs numériques pour définir des valeurs minimales ou maximales à respecter.

- **Grid** périphérique bitmap ou grille (ex : lcd)
- **Height** dimension en hauteur de la zone d'affichage
- **monochrome** périphérique monochrome ou niveaux de gris (bits/pixel)
- **Orientation** orientation du périphérique (portait ou landscape)
- **Resolution** résolution du périphérique (en dpi, dppx, ou dpcm)
- **Scan** type de balayage des téléviseurs (progressive ou interlace)

Notez ces remarques

- ▶ Les dimensions pourront être évaluées avec des unités (px, em). Les ratios avec des fractions (entier/entier). Une résolution sera définie en dpi (points par pouce) ou en dpcm (points par centimètres).
- ▶ Certaines de ces propriétés peuvent être testées d'une façon raccourcie sans valeur, par exemple (color) qui sera équivalent à (min-color : 1) ou considérée comme vraie pour une valeur différente de 0. La fonctionnalité monochrome n'est pas uniquement booléenne avec la syntaxe raccourcie (monochrome), on peut aussi considérer un nombre de niveaux de gris, par exemple (min-monochrome : 2) pour 2 bits par pixel.

Démonstration



Au service des mobiles

Media Queries

Pour produire des améliorations spécifiques à l'affichage sur les mobiles, qui sont directement concernés par des critères sur les dimensions de l'écran (en terme de résolution et d'espace disponible) et sur l'utilisation tactile.

Nous trouvons des règles pour :

- ▶ agrandir la taille du texte
- ▶ agrandir la taille des contrôles et zones cliquables (pour une utilisation au doigt)
- ▶ faire passer le contenu sur une seule colonne
- ▶ masquer ou afficher des éléments spécifiques
- ▶ ajuster les dimensions et marges

Au service des mobiles

Exemple

La fonctionnalité orientation a été introduite pour des périphériques pouvant être orientés. On peut alors déclarer une feuille de style spécifique pour ajuster l'affichage

Exemple

“

```
<link rel="stylesheet" media="(orientation:portrait)" href="portrait.css">  
  
<link rel="stylesheet" media="(orientation:landscape)" href="paysage.css">
```

”