

# CSS3 : les bases

- Rappel : structure d'un document CSS
- Sélecteurs
- Couleurs
- Unités
- Propriétés pour l'arrière-plan
- Propriétés pour la typographie et les polices
- Propriétés pour le texte
- Pseudo-classes, pseudo-éléments, @-règles

# Préalable

- La syntaxe d'une règle CSS est simple :  
**sélecteur** { *propriété:valeur* ; }
- La difficulté avec CSS est le nombre de sélecteur, de propriété et de valeur disponibles
- **Conseils :**
  - Prendre le temps de comprendre toutes les subtilités du langage CSS
  - Ne pas faire un site comme un magazine (au pixel près) car trop grande diversité de navigateurs, taille d'écran,...

# Structure d'un document CSS

- Un document CSS doit toujours s'adapter à une structure HTML
- Idéalement, on place le code **CSS** dans un **fichier séparé** et on le **lie au fichier HTML**
- Exemple de code CSS :

```
h1 { color:red; background-color:yellow;}
```

```
#titreprincipal {text-align:center;}
```

```
.news p {font-family:Arial; font-size:16px;}
```

# CSS = Standard du W3C

- Le langage **CSS** est spécifié par des **standards** définis par le **W3C**
- Parfois, les constructeurs de navigateur veulent **implémenter des propriétés** plus rapidement, on place le **préfixe** du constructeur **devant le nom**
  - moz- : Mozilla Firefox
  - webkit- : Google Chrome et Apple Safari
  - o- : Opera
  - ms- : Microsoft Internet Explorer et Edge

# Sélecteurs (CSS1/CSS 2.1)

- **Le nom d'un élément**

```
h1 {color:blue}
```

- **Un élément descendant d'un autre élément**

```
ul li a {text-decoration:none}
```

- **Un élément avec l'attribut id="unevaleur"**

```
#unevaleur {background-color : lightgreen}
```

- **Un élément avec l'attribut class="news"**

```
.news {width : 250px}
```

# Sélecteurs (CSS1/CSS 2.1)

- **Combinaison de plusieurs sélecteurs** avec les mêmes règles de style

h1, h2, h3 {font-family : Helvetica}

- **N'importe quel élément**

\* {margin:0; padding:0}

- Un élément **descendant direct** d'un autre

header > h1 {font-size : 150%}

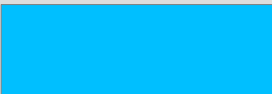
- Un élément **successeur direct** d'un autre

h2 + p {color : aquamarine}

# Sélecteurs (CSS1/CSS 2.1)

- Un élément avec un **attribut spécifique**  
`p[title] {font-style: italic}`
- Un élément avec un **attribut="valeur"**  
`input[name="type"] {width:1em}`
- Un élément avec un **attribut contenant ...**  
`div[class~="col"] {border:none}`
- Un élément avec un **attribut commençant par...**  
`p[lang|="fr"] {color : aquamarine}`

# Couleurs

- Pour **définir la couleur**, on a plusieurs moyens :
  - mot-clé (red, blue, black,...)
  - notation hexadécimale (#23FC48)
  - valeur RGB : rgb(0,128,255) ou rgb(5%,15%,75%)
  - Valeur HSL (Hue Saturation Lightness)
  - Valeur CMYK (Cyan Magenta Yellow Black)
- Par exemple :  deepskyblue = #00BFFF  
= rgb(0, 191, 255) = hsl(195, 100%, 50%) =  
cmyk(100%, 25%, 0%, 0%)



# Binaire, décimal, hexadécimal

Décimal	Binaire				Hexadécimal
	8	4	2	1	
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	2
3	0	0	1	1	3
4	0	1	0	0	4
5	0	1	0	1	5
6	0	1	1	0	6
7	0	1	1	1	7
8	1	0	0	0	8
9	1	0	0	1	9
10	1	0	1	0	A
11	1	0	1	1	B
12	1	1	0	0	C
13	1	1	0	1	D
14	1	1	1	0	E
15	1	1	1	1	F

- En **binaire**, 2 valeurs possibles : 0 ou 1
- En **décimal**, 10 valeurs possibles : de 0 à 9
- En **hexadécimal**, 16 valeurs possibles : 0 à 9 et A, B, C, D, E, F

<http://villemin.gerard.free.fr/Wwwgvm/Numerati/ConHexa.htm>

# Couleurs (liens et outils)

- Quelques liens et outils à propos des couleurs :

[https://developer.mozilla.org/fr/docs/Web/CSS/Type\\_color](https://developer.mozilla.org/fr/docs/Web/CSS/Type_color)

<https://www.w3.org/wiki/CSS/Properties/color/keywords>

– <https://www.tutorialrepublic.com/css-reference/css-color-names.p>

– [https://www.w3schools.com/colors/colors\\_converter.asp](https://www.w3schools.com/colors/colors_converter.asp)

– [https://developer.mozilla.org/fr/docs/Web/CSS/Couleurs\\_CSS/Sé](https://developer.mozilla.org/fr/docs/Web/CSS/Couleurs_CSS/Sé)

– <https://color.adobe.com/fr/create/color-wheel/>

– [https://www.w3schools.com/colors/colors\\_picker.asp](https://www.w3schools.com/colors/colors_picker.asp)

– <https://99designs.fr/blog/conseils-design/la-psychologie-des-couleur>

– [https://fr.wikipedia.org/wiki/Couleur\\_du\\_Web](https://fr.wikipedia.org/wiki/Couleur_du_Web)

# Couleurs spéciales

- Il y a aussi une couleur nommée **transparent**
- Pour définir la **transparence d'un élément**, on a la propriété **opacity** ayant une valeur entre 0.0 (transparent) et 1.0 (opaque)
- On peut définir l'**alpha d'une couleur** mais la **transparence ne s'applique qu'à la couleur** pas à tout l'objet.  
On utilise **rgba(...)** au lieu de **rgb(...)**
  - Exemple : `background-color : rgba(228,77,5,0.6)`  
pour une opacité de 60% sur la couleur

# Unités de longueur

- les **unités relatives** ne représentent pas une longueur prédéfinie mais une longueur **par rapport à une référence**

**% em ex ch rem fr dpi vw vh vm**

- Les **unités absolues** représentent une **longueur prédéfinie**, donc dépendante de la résolution du média sur lequel on affiche

**mm cm in px pt pc**

*1 in (inch:pouce) = 25,4 mm = 2,54 cm*

*1 in = 72 pt (point)      1 pc (pica) = 12 pt*

[https://developer.mozilla.org/fr/docs/Apprendre/CSS/Introduction\\_à\\_](https://developer.mozilla.org/fr/docs/Apprendre/CSS/Introduction_à_)

# Autres unités

- Pour les transformations en CSS3, on a ajouté des **unités d'angle**

**deg grad rad turn**

*1 tour (turn) = 360 deg = 400 grad =  $2\pi$  rad*

- Pour les animations en CSS3, on a ajouté des **unités de temps**

**s ms**     *1000 ms = 1 s (seconde)*

<https://developer.mozilla.org/fr/docs/Apprendre/CSS/Intro>

# Arrière-plan (CSS1/CSS 2.1)

- Toutes les images dans le design d'une page Web ne doivent pas être placées dans des `<img>`. Il y a aussi celles en **arrière-plan** pour la décoration qu'on peut régler avec les propriétés :
  - background-color
  - background-image
  - background-repeat
  - background-attachment
  - background-position
  - background

# Arrière-plan (CSS3)

- En CSS3, on peut avoir **plusieurs arrière-plans** pour un seul élément. Il faut séparer les données par une virgule dans la règle de style
- Il y a aussi des **nouvelles propriétés** :
  - background-clip
  - background-origin
  - background-size
  - background-blend-mode

# Typographie et polices

- La typographie est importante dans le design pour contrôler l'affichage du texte d'un site web
- Les différentes familles de police sont :
  - **serif** : police à empattement (Times New Roman)
  - **sans-serif** : police sans empattement (Arial)
  - **monospace** : police à chasse fixe (Courier)
  - **cursive** : police manuscrite (Comic Sans MS)
  - **fantasy** : police décorative (Impact)

<https://www.w3.org/Style/Examples/007/fonts.fr.html>



# Typographie (CSS1/CSS 2.1)

- Pour modifier l'affichage du texte en fonction des **caractéristiques des polices**, on dispose de plusieurs propriétés :
  - font-style
  - font-variant
  - font-weight
  - font-size\* / line-height
  - font-family\*
  - font

\* valeurs obligatoires pour **font**

# Typographie (CSS3)

- En CSS3, il y a des **nouvelles propriétés** mais non supportées sur tous les navigateurs : (à vérifier sur : <https://www.caniuse.com> )
  - font-size-adjust
  - font-stretch
  - font-kerning
  - font-variant-\* et font-feature-settings
- Il est possible d'**ajouter des polices** qui peuvent être téléchargées avec la règle **@font-face**

# Polices embarquées

- Avec la règle **@font-face**, on peut ajouter des polices téléchargées sur Internet
- Les formats sont : TTF, OTF, EOT, WOFF, SVG
- Quelques sites où trouver des polices :
  - <https://fonts.google.com/>
  - <https://www.fontsquirrel.com/>
  - <https://www.1001freefonts.com/>
  - <https://www.dafont.com/fr/>

# Texte ( CSS1 / CSS 2.1 )

- Il y a plusieurs propriétés modifiant le **texte** :
  - color
  - text-align et vertical-align
  - text-decoration
  - text-indent
  - letter-spacing et word-spacing
  - text-transform
  - line-height
  - direction
  - white-space

# Texte (CSS3)

- En CSS3, il y a des **nouvelles propriétés** mais non supportées sur tous les navigateurs : (à vérifier sur : <https://www.caniuse.com> )
  - text-overflow
  - text-wrap
  - word-wrap
  - text-shadow
  - text-orientation
  - hyphens

# Pseudo-classes (CSS1/CSS 2.1)

- Les **pseudo-classes** s'associent aux sélecteurs ordinaires pour devenir des sélecteurs plus spécifiques. Elles commencent par un **simple deux-points**
- Pour les interactions souris et clavier :  
:link    :visited    :hover    :active    :focus
- Et encore :  
:first-child    :lang

# Pseudo-éléments (CSS1/CSS 2.1)

- En CSS, les **pseudo-éléments** permettent de sélectionner une partie d'un élément. Pour les distinguer des pseudo-classes, ils commencent par un **double deux-points**

::first-line

::first-letter

::before

::after

# Sélecteurs (CSS3)

- Élément dont la ***valeur de l'attribut*** commence par ...

```
p[title^="test"] {font-style: italic}
```

- Élément dont la ***valeur de l'attribut*** se termine par ...

```
input[name$="type"] {width:1em}
```

- Élément dont la ***valeur de l'attribut*** contient ...

```
div[class*="col"] {border:none}
```

- Élément qui **suit directement** un autre élément

```
div~p {color:blueviolet}
```



# Pseudo-classes (CSS3)

- Pour désigner le **n<sup>ième</sup> enfant** de son parent  
:nth-child(n)
- Pour désigner le **n<sup>ième</sup> enfant de son parent en partant de la fin**  
:nth-last-child(n)
- Pour désigner le **premier** ou le **dernier** enfant  
:first-child            :last-child

# Pseudo-classes (CSS3)

- Élément qui a **n frères du même type avant lui**  
:nth-of-type(n)
- Élément qui a **n-1 frères du même type après lui**  
:nth-last-of-type(n)
- Élément qui est le **premier** ou le **dernier** enfant de son type parmi les enfants de son parent  
:first-of-type                      :last-of-type
- Élément qui est **enfant unique** ou **seul enfant d'un type donné** de son parent ou **aucun enfant**  
:only-child                      :only-of-type                      :empty

# Pseudo-classes (CSS3)

- Pour trouver la **racine** du document (<html>)  
:root
- Pour désigner l'élément **cible d'une ancre** (#) et modifier l'aspect d'une partie de la page  
:target
- Pour un **élément qui n'est pas représenté** par l'expression  
:not(expression)

# @-règles

- Ce sont des **règles qui étendent la syntaxe**, elles doivent être **écrites en premier** avant les commentaires. Quelques exemples :

**@charset 'UTF-8'**; pour définir l'encodage des CSS

**@import url('style.css')**; charger une CSS externe

**@font-face** pour définir une police externe

**@page** pour définir la mise en page pour imprimer

**@media** pour cibler un type de média

- CSS3 en a ajouté : **@viewport**, **@namespace**, **@supports**, **@counter-style**,...