

CSS

.

CSS

C'est quoi ?



CSS

C'est quoi ?

Le CSS, c'est quoi ?

- CSS : **C**ascading **S**tyle**S**heets (Feuille de Style en Cascade)
- C'est une recommandation du W3C
- Apparu en 1996
- Regroupe dans un même document des caractéristiques de mise en forme associées à des groupes d'éléments.

Avantage du CSS

- Compense les limites du HTML concernant la mise en page et la présentation
- Séparation du contenu et de la mise en forme = meilleur lisibilité du HTML.
- Possible de créer plusieurs feuilles de style qui héritent en cascade.
- Les CSS sont mis en mémoire cache = chargement des pages plus rapide.

CSS 2 → 1997

- La prise en compte de la diversité de médias (écran, imprimante, braille, synthétiseur vocal, etc.).
- L'ajout de propriétés auditives pour les déficients visuel.
- La manipulation de curseurs.
- La gestion des débordements et de la visibilité des éléments.

CSS 3 → 1999

- Modularité des feuilles de style avec les “Media Queries”
 - Le CSS s’adapte au terminal
 - Début du Responsive design

CSS 4

- En cours d'écriture depuis 2010 !!
- Etat du dernier projet, publié le 2 mai 2013 à l'adresse :

<https://www.w3.org/TR/selectors4/>

CSS

Compatibilité

Compatibilité

- Tous les navigateurs supportent le CSS.
- Tous les navigateurs n'interprètent pas le CSS de la même façon.
- Les propriétés CSS en cours de validation ou d'implantation sont préfixés en fonction du moteur de rendu.

Compatibilité : Les moteurs de rendu

- **Webkit** : Chrome, Safari, Opera **préfix** : -webkit-
- **Gecko** : Firefox **préfix** : -moz-
- **Trident** : M.S. Internet Explorer **préfix** : -ms-
- **EdgeHTML** : Edge

CSS

Les commentaires

Les commentaires

- Les commentaires permettent de décrire le code CSS
- Ils ne sont pas interprétés par les moteurs de rendu

```
/* Ceci est un commentaire */
```

CSS

Intégration

Feuille de style interne

- La feuille de style est directement écrite dans le document HTML

```
<html>
  <head>
    <style type="text/css">
      h1 { color: red; }
    </style>
  </head>
  <body>
    ...
  </body>
</html>
```

Attribut de style - Style inline

- Ajout de l'attribut **style** sur l'élément HTML

```
<html>
  <body>
    <p style="color: green">Ce texte est vert.</p>
  </body>
</html>
```

- A ne pas faire dans une page web.
- A faire pour un email au format HTML.

Feuille de style externe

- Déclaration dans la section **<head />**
- Utilisation de la balise HTML **<link />**
- Les fichiers portent l'extension **.css**

```
<html>
  <head>
    <link type="text/css" rel="stylesheet" href="theme.css">
  </head>
  <body>
    ...
  </body>
</html>
```

Feuille de style externe - Importation

- Déclaration dans la section **<head />**
- Utilisation de la balise HTML **<link />**
- Les fichiers portent l'extension **.css**

```
<html>
  <head>
    <style type="text/css">
      @import url(http://site.com/theme.css);
    </style>
  </head>
  <body>
    ...
  </body>
</html>
```

Ordre de priorité

1. Feuille de style par défaut du navigateur
2. Feuille de style externe
3. Feuille de style interne
4. Feuille de style inline
 - Un style **inline** surcharge tous les autres !

CSS

Les sélecteurs

La syntaxe

- Sélecteur unique :

```
selecteur {  
    propriété : valeur ;  
}
```

- Groupement de sélecteurs :

```
selecteur_1, selecteur_2 {  
    propriété : valeur ;  
}
```

Le sélecteur universel

- Sélecteur : *

Exemple

```
/* Tous les éléments ont une bordure de 1 pixel de couleur rouge */
```

```
* { border: 1px solid red; }
```

Les sélecteurs de type.

- `element { propriété : valeur; }`

Exemple

HTML

```
<p>Hello <span>world</span>.</p>
```

CSS

```
span { color: red; }
```

Rendu

Hello world.

Les sélecteurs de classe

- `.classe { propriété : valeur; }`

Exemple

HTML

```
<p>Hello <span class="green">world</span>.</p>
```

CSS

```
.green { color: green; }
```

Rendu

Hello world.

Les sélecteurs d'ID

- `#id { propriété : valeur; }`

Exemple

HTML

```
<p>Hello <span id="blue">world</span>.</p>
```

CSS

```
#blue { color: blue; }
```

Rendu

Hello **world**.

Groupement de sélecteurs

HTML

```
<p><a href="...">Hello</a> <span>world</span>.</p>
```

CSS

```
a, span { background-color: orange; }
```

```
a { color: green; }
```

```
span { color: red; }
```

Rendu



Visual rendering of the HTML code: The word "Hello" is green and the word "world" is red, both are on an orange background.

Sélecteurs descendants

HTML

```
<p>Hello <span>world</span>.</p>
```

```
<p><a href=“...”>Hello <span>world</span>.</a></p>
```

CSS

```
p span { color: red; }
```

Rendu

Hello world.

Hello world.

Sélecteurs enfants

HTML

```
<p>Hello <span>world</span>.</p>
```

```
<p><a href="...">Hello <span>world</span>.</a></p>
```

CSS

```
p > span { color: red; }
```

Rendu

Hello world.

Hello world.

Sélecteurs adjacents

HTML

```
<ul>
  <li><a href="#">Accueil</a></li>
  <li><a href="#">A propos</a></li>
  <li><a href="#">Contact</a></li>
</ul>
```

CSS

```
li + li { border-left: 1px solid #aaa; }
```

Rendu

[Accueil](#) | [A propos](#) | [Contact](#)

CSS

Ordre de priorité des sélecteurs

Priorité des sélecteurs

- Le sélecteur d'ID prendra toujours le dessus.

HTML

```
<div id="test" class="test">test</div>
```

Rendu

Test

CSS

```
div {  
    color: green;  
}  
#test {  
    color: yellow;  
}  
.test {  
    color: red;  
}
```

CSS

Les pseudo classes

Les pseudos classes

- Mot clé ajouter au sélecteur qui définit l'état dans lequel l'élément doit être pour être ciblé
 - Définit un style en fonction de son contenu dans l'arbre du document.
 - Définit un style en fonction de l'historique de navigation (ex. `:visited`)
 - Définit un style en fonction du statut du contenu (ex. `:checked`)
 - Définit un style en fonction de la position de la souris (ex. `:hover`)

Les pseudos classes

- Syntaxe :

```
selecteur:pseudo-classe {  
    propriété : valeur ;  
}
```

Les pseudos classes

- **:root** cible la racine du document.
- **:link** cible les liens non visités du document.
- **:visited** cible les liens visités du document.
- **:hover** cible éléments survolés par la souris.
- **:first-child** cible le premier élément enfants.

CSS

Les pseudo éléments

Les pseudos classes

- Mot clé ajouter au sélecteur pour décrire un état spécial.
 - Permet de mettre en forme certaine partie de l'élément.

Les pseudos éléments

- Syntaxe :

```
selecteur::pseudo-element {  
    propriété : valeur ;  
}
```

CSS

Les unités

Les unités relative

- **em** Taille de police. 2em = 2 fois la taille de la police courant (*écran / imprimante*)
- **%** Pourcentage (*écran / imprimante*)
- **rem** Taille de la police relative à l'élément racine

Les unités absolue

- **cm** Centimètre (*imprimante*)
- **mm** Millimètre (*imprimante*)
- **in** Pouce (*imprimante*)
- **px** Pixel (*écran*)
- **pt** Point (*imprimante*)
- **pc** Pica (*imprimante*)

Relations entre les unités

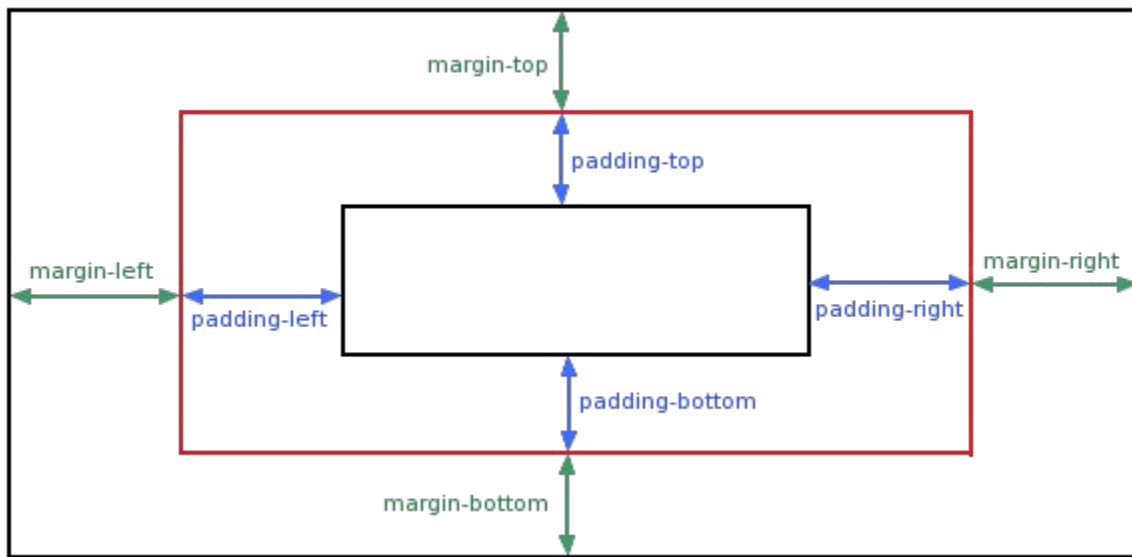
$$1\text{in} = 2.54\text{cm} = 25.4\text{mm} = 72\text{pt} = 6\text{pc}$$

CSS

Les bordures

Les bordures

la marge (margin) ; la bordure (border) ; le remplissage (padding)



Les bordures

border : *valeur* ;

border : *valeur(top-bottom) valeur(right-left)* ;

border : *valeur(top) valeur(right) valeur(bottom) valeur(left)* ;

border-top : *valeur* ;

border-right : *valeur* ;

border-bottom : *valeur* ;

border-left : *valeur* ;

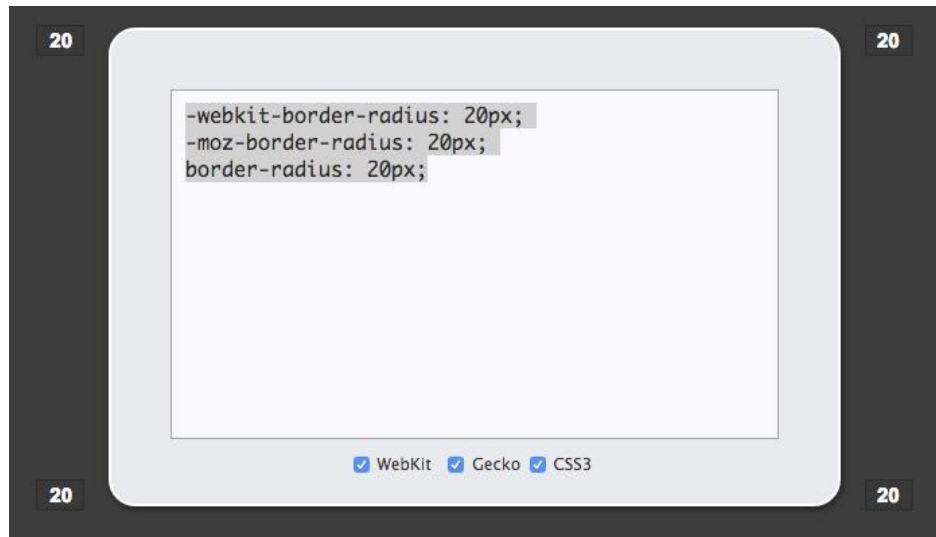
border-width : *valeur* ;

border-color : *valeur* ;

border-style : *valeur* ;

Les bordures : radius

- Permet d'arrondir les angles
- <http://border-radius.com/>



CSS

Les couleurs

Les couleurs

- Les couleurs Hexadécimal : `#FF0000`;
- Les couleurs RGB : `rgb(255, 0 0)`;
- Les couleurs RGBA : `rgba(255, 0, 0, .5)`;

CSS

Les outils en ligne

Quelques ressources en ligne

- **Google** : saisir “color picker” dans le moteur de recherche.
 - **Border Radius Generator** : <http://border-radius.com/>
 - **CSS Gradients** : <http://www.colorzilla.com/gradient-editor/>
 - **CSS3 Generator** : <http://css3generator.com/>
 - **CSS Arrow** : <http://www.cssarrowplease.com/>
-
- <https://css-tricks.com/almanac/>
 - <http://css4-selectors.com/>