

Cours CSS

Les directives (partie 1)

Les directives, ce sont les règles que l'on applique aux éléments ciblés par les sélecteurs css. Par exemple : la couleur (couleur de texte, couleur de fond), l'opacité, la taille, le positionnement sur la page

Documentation de référence : <https://developer.mozilla.org/fr/docs/Web/CSS>

Propriétés courantes

Les propriétés de texte

Exemples

```
color: #be0f34;  
font-size: 16px;  
line-height: 1,2rem;  
font-weight: bold;  
text-align: center;  
text-decoration: none;  
text-transform: uppercase ;  
opacity: 0.7;  
cursor: pointer;
```

Listes

Exemples

```
list-style-type: none;  
padding-left: 0;
```

Styles des blocs

Exemples

```
padding: 1em;  
margin: 1em 2em;  
border: 3px solid #be0f34;
```

```
border-radius: 5px;  
opacity: 0.7;  
box-shadow: 0 0 20px 0 rgba(0, 0, 0, 0.2), 0 5px 5px 0 rgba(0, 0, 0, 0.24);  
outline: 0;
```

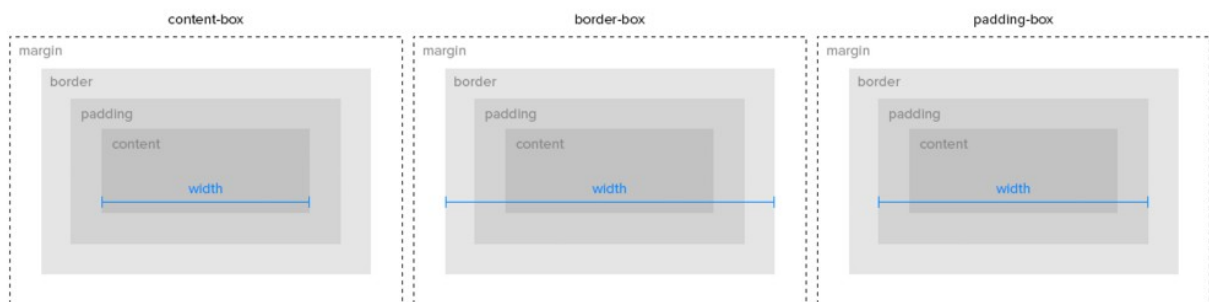
Dimensionnement

Les éléments qui sont de type « block » occupent par défaut l'ensemble de la largeur disponible.

Exemples de directives

```
width: 100%;  
max-width: 1000px;  
height: 350px;  
calc(100% - 80px);  
box-sizing: border-box;
```

Box-sizing : Les différents modes de calcul de la largeur



Positionnement

Centrer horizontalement

```
width: 500px;  
margin: 0 auto;
```

Décaler

```
margin-top: -60px;  
left : 30px  
transform: translate(0,100px) ;
```

Positionnement absolu

```
position: absolute;
```

```
position: relative; // Appliqué au parent !  
bottom: 0;  
left: 2%;  
z-index: 5
```

Exemple de positionnement absolu : Légende positionnée sur l'image.



Les unités de mesure

- pixel
- em
- rem
- %
- vh, vw

Formats de couleur

Exemples

```
purple  
#F09  
#ff0099  
rgb(255,0,153)  
rgba(51, 170, 51, 0.7)
```

La propriété display

```
display: block;  
display: inline;  
display: inline-block;
```

```
display: flex;
```

La directive « content »

Pour créer du contenu html en css

Exemple :

```
<p class="obligatoire">Nom :</p>

p.obligatoire:after {
  content: "*";
  color: red;
}
```

Pour aller plus loin

Can I use

Vérifier la compatibilité d'une directive pour l'ensemble des navigateurs

S'entraîner

<https://www.commentcoder.com/jeux-apprendre-css/>

Le reset

Le reset CSS est une technique qui consiste à réinitialiser à 0 la valeur de certains éléments HTML afin d'éviter une partie des différences d'affichage sur les divers navigateurs.

<https://www.alsacreations.com/astuce/lire/36-la-technique-du-reset-css.html>

Les variables

Les variables CSS sont des outils extrêmement utiles pour centraliser et réutiliser des valeurs de style. Elles permettent de définir une valeur une seule fois, puis de la réutiliser dans la feuille de style. Cette fonctionnalité apporte de nombreux avantages en termes de maintenabilité et de flexibilité.

Variable « globales »

Exemple :

```
/* Déclaration des variables */

:root {
  --blanc: #fff;
  --gris-fonce: #555;
}

/* Utilisation des variables */

body {
  background-color: var(--blanc);
  color: var(--gris-fonce);
}
```

Variables « locales »

Les variables CSS peuvent être redéfinies localement pour des sections spécifiques .

Par exemple, si vous souhaitez changer la couleur principale dans une certaine section de votre site, vous pouvez le faire en redéfinissant la variable uniquement pour cette partie.

```
.section-1 {
  --main-color: #d32f2c;
}

.section-2 {
  --main-color: #3fcef2;
}
```

Ensuite, dans vos styles, vous utilisez toujours `var(--main-color)` comme d'habitude, mais la valeur s'adaptera en fonction de la section où elle est utilisée :

```
.section-1 h1 {
  color: var(--main-color); /* Couleur rouge */
}

.section-2 h1 {
  color: var(--main-color); /* Couleur cyan */
}
```

```
}
```

<https://talks.freelancerepublik.com/variables-css/>

<https://www.alsacreations.com/tuto/lire/1929-Tout-savoir-sur-les-variables-CSS-custom-properties.html>

La convention BEM

BEM, acronyme de Block, Element, Modifier, est une convention de nommage qui vise à organiser les classes CSS de manière hiérarchique. Elle repose sur trois notions essentielles :

- Block (B) : un composant autonome, comme un bouton ou une carte ;
- Element (E) : une partie du bloc qui ne peut exister de façon indépendante (exemple : un titre dans une carte) ;
- Modifier (M) : une variation du bloc ou de l'élément (exemple : un bouton actif, un titre en gras).

Exemple

```
.button {  
  background-color: blue;  
  color: white;  
}  
.button--primary {  
  background-color: red;  
}  
.button__icon {  
  margin-right: 8px;  
}
```

Dans cet exemple, `.button` est un bloc, `.button__icon` est un élément et `.button--primary` est un modificateur.

Doc :

<https://alticreation.com/blog/bem-pour-le-css/>

Pour aller plus loin

S'entraîner

https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Styling_basics/Test_your_skills