

Programmation Web

Le CSS - Styles et unités

Pierre Blarre - Florian Rodriguez

Valeurs et unités (longueurs, nombres, couleurs)

Types de valeur numérique

- **<integer>** Une valeur de type <integer> est un nombre entier comme 1024 ou -55.
- **<number>** Une valeur de type <number> représente un nombre décimal (qui peut avoir ou ne pas avoir de partie décimale). Par exemple : 0.255, 128, ou -1.2.
- **<dimension>** Une valeur de type <dimension> est une valeur de type <number> à laquelle est attachée une unité. Par exemple : 45deg, 5s, ou 10px. <dimension> est une catégorie de type qui inclut les types <length>, <angle>, <time>, et <resolution>.
- **<percentage>** Une valeur de type <percentage> représente une fraction relative à une autre valeur. Par exemple : 50%. Les valeurs de pourcentages sont toujours relatives à une autre quantité. Ainsi, la longueur d'un élément pourra être relative à la longueur de son élément parent.

Longueurs - absolues

- **cm** Centimètres $1\text{cm} = 38\text{px} = 25/64\text{in}$
- **mm** Millimètres $1\text{mm} = 1/10\text{th of } 1\text{cm}$
- **Q** Quarts de millimètre $1\text{Q} = 1/40\text{th of } 1\text{cm}$
- **in** Pouces (inches) $1\text{in} = 2.54\text{cm} = 96\text{px}$
- **pc** Picas $1\text{pc} = 1/6\text{e de } 1\text{in}$
- **pt** Points $1\text{pt} = 1/72\text{e de } 1\text{in}$
- **px** Pixels $1\text{px} = 1/96\text{th de } 1\text{in}$

La plupart de ces unités sont utiles pour l'impression plutôt que pour l'affichage sur un écran. Ainsi, on n'utilise généralement pas cm (centimètres) sur un écran. La seule unité ici que vous rencontrerez fréquemment est px (pixels).

Longueurs - relative

- **em** Pour les propriétés typographiques comme font-size, relative à la taille de la police de l'élément parent. Pour les autres propriétés comme width, relative à la taille de la police de l'élément;
- **rem** Relative à la taille de la police de l'élément racine.
- **ex** La hauteur d'x de la police de l'élément.
- **ch** La chasse/avance du glyphe « 0 » pour la police de l'élément.
- **lh** La hauteur de ligne pour l'élément.
- **vw** 1% de la largeur du viewport (la zone d'affichage).
- **vh** 1% de la hauteur du viewport (la zone d'affichage).
- **vmin** 1% de la plus petite dimension du viewport (la zone d'affichage).
- **vmax** 1% de la plus grande dimension du viewport (la zone d'affichage).

ems et rems

Source

```
<!DOCTYPE html>
<html lang="en-US">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width">
    <title>Ems et Rems</title>
    <style>
      html {
        font-size: 16px;
      }

      .ems li {
        font-size: 1.3em;
      }

      .rems li {
        font-size: 1.3rem;
      }
    </style>
  </head>
  <body>
    <p>Ceci est un paragraphe</p>
    <ul class="ems">
      <li>One</li>
      <li>Two</li>
      <li>Three
```

Copy

Résultat

Ceci est un paragraphe

- One
- Two
- Three
 - Three A
 - Three B
 - Three B 2
- One
- Two
- Three
 - Three A
 - Three B
 - Three B 2

[Ouvrir l'exemple](#)

Pourcentages

- Dans la plupart des cas, un pourcentage est traité comme une longueur.
- Un pourcentage est toujours relatif à une autre valeur.
- Si vous définissez la propriété font-size d'un élément avec un pourcentage, ce pourcentage sera relatif à la valeur de font-size de l'élément parent.
- Si vous utilisez un pourcentage pour width, la valeur obtenue sera le pourcentage de la valeur de width pour l'élément parent.

Nombres

- Certains types de valeur acceptent des nombres, sans unité.
- Une telle propriété est, par exemple, opacity qui contrôle l'opacité d'un élément.
- Cette propriété acceptera une valeur numérique (sans unité) comprise entre 0 (transparence complète) et 1 (complètement opaque).

Couleurs

- RGB (red green blue) hexadécimales #000000 -> 256 possibilités
- RGB/RGBA `rgb(0, 0, 0)`
- HSL/HSLA (hue/teinte, saturation, luminosité) `hsl(0, 0%, 0)`

Style du texte (color, fonts, alignement, espacement)

Font / Police

```
p {  
  font-family: "Trebuchet MS", Verdana, sans-serif;  
  color:red;  
  font-size:16px;  
  font-style:italic;  
  font-weight:bold;  
  text-transform:uppercase;  
  text-decoration:underline;  
  text-align:center;  
  line-height: 1.5;  
  letter-spacing: 2px;  
  word-spacing: 4px;  
}
```

Copy

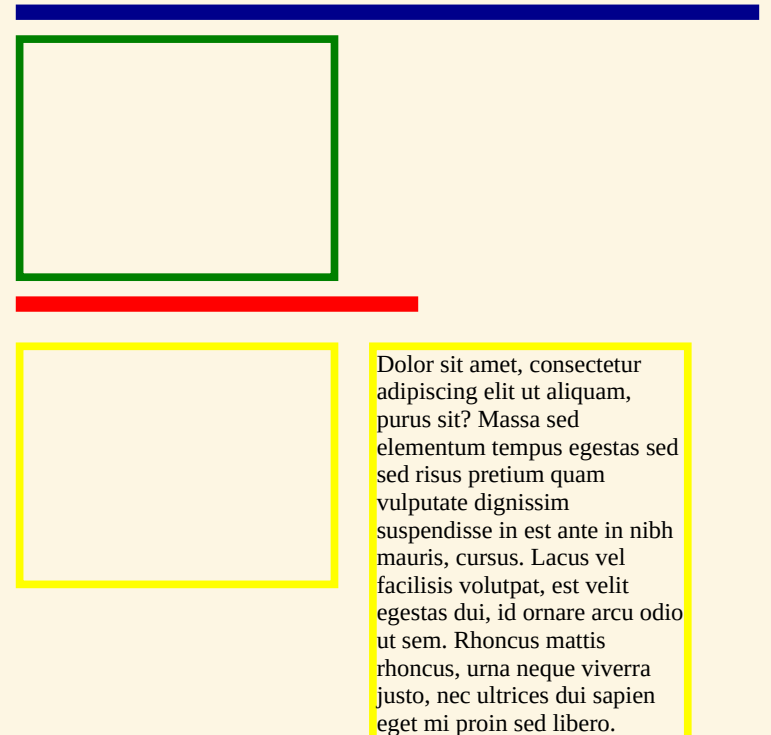
Taille des élément

Source

```
<!DOCTYPE html>
<html lang="en-US">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width">
    <title>Taille</title>
    <style>
      div {
        margin: 10px;
      }
      .box1 {
        border: 5px solid darkblue;
      }
      .box2 {
        border: 5px solid green;
        height: 150px;
        width: 200px;
      }
      .box3 {
        border: 5px solid red;
        width: 50%;
      }
      .box4 {
        border: 5px solid yellow;
        min-height: 150px;
        width: 200px;
      }
    </style>
  </head>
  <body>
    <div>
      <div class="box1"></div>
      <div class="box2"></div>
      <div class="box3"></div>
      <div class="box4">
        Dolor sit amet, consectetur
        adipiscing elit ut aliquam,
        purus sit? Massa sed
        elementum tempus egestas sed
        sed risus pretium quam
        vulputate dignissim
        suspendisse in est ante in nibh
        mauris, cursus. Lacus vel
        facilisis volutpat, est velit
        egestas dui, id ornare arcu odio
        ut sem. Rhoncus mattis
        rhoncus, urna neque viverra
        justo, nec ultrices dui sapien
        eget mi proin sed libero.
      </div>
    </div>
  </body>
</html>
```

Copy

Résultat



Les variables CSS

Les propriétés personnalisées CSS (custom properties en anglais, aussi parfois appelés variables CSS) sont des entités définies par les développeurs ou les utilisateurs d'une page Web, contenant des valeurs spécifiques utilisables à travers le document. Elles sont initialisées avec des propriétés personnalisées (par exemple `--main-color: black;`) et accessibles en utilisant la notation spécifique `var()` (par exemple `:color: var(--main-color);`).

- Déclaration

```
element {  
  --main-bg-color: brown;  
}
```

[Copy](#)

- Utilisation

```
element {  
  background-color: var(--main-bg-color);  
  color: var(--my-var, red); // valeur par défaut  
}
```

[Copy](#)

Media queries

- Les requêtes média (media queries) permettent de modifier l'apparence d'un site ou d'une application en fonction du type d'appareil (impression ou écran par exemple) et de ses caractéristiques (la résolution d'écran ou la largeur de la zone d'affichage (viewport) par exemple).
- Les requêtes média sont utilisées afin :
 - D'appliquer certains styles de façon conditionnelle avec le CSS grâce aux règles `@media` et `@import`.
 - De cibler certains médias pour les éléments `<style>`, `<link>`, `<source>` et d'autres éléments HTML grâce à l'attribut `media`.

Print

- utiliser la balise `@media print{}` dans une des feuilles de style appelée sur le site

```
@media print {  
  body {  
    font-size: 14px;  
  }  
}
```

Copy

- créer une feuille de style à part avec comme propriété `media print`.

```
<link media="print" href="print.css" />
```

Copy

Responsive

- En fonction par exemple de la taille de l'écran

```
/* Appliquer des styles à condition que la largeur de la zone d'affichage (viewport) soit inférieure à 1250px :  
@media (max-width: 1250px) { ... }  
  
/* Appliquer des styles aux appareils orientés en mode paysage et dont la largeur mesure au moins 30ems */  
@media (min-width: 30em) and (orientation: landscape) { ... }
```

Copy

- [Voir les différentes possibilités](#)