

CSS

Cours pour les développeurs d'applications Web et mobile

Ceci est la référence de cours pour CSS. Les différents aspects à traiter au sein de ce module y sont détaillés.

Prérequis

[HTML](#)

Outils recommandés pour le développement en CSS

- Visual Studio Code
 - *Extensions* : HTML CSS Support

Mise en place

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    ...
    <link rel="stylesheet" href="main.css" type="text/css">
  </head>
</html>
```

```
/* Fichier main.css - p est un sélecteur */
p {
  color: "red";
}
```

Utilisation dans le HTML

```
<p style="color: blue">Un paragraphe</p>
<!-- ou -->
<head>
  <style>
    p {
      color: green;
    }
  </style>
</head>
```

Selecteurs

```
* {
  font-family: monospace;
}

body {
  font-size: 20px;
}

h1, h2 {
  color: blue;
}

.gray {
  color: gray;
}

#two {
  font-style: italic;
}

p span {
  text-transform: uppercase;
  background-color: gold;
}
```

Couleurs

```
p {
  color: blue;
  /* or */
```

```
color: rgb(0, 0, 255);
/* or */
color: #0000FF;

color: rgba(0, 255, 0, 0.5);
}
```

A visiter : [colors.co](https://www.colors.co)

Unités

```
p {
    font-size: 20px;
    border: 2px dashed black;
    padding: 0.5em;
}

.parent {
    width: 100%;
    font-size: 2rem;
}

.child {
    width: 50%;
    font-size: 2em;
}

body {
    min-height: 100vh;
}

main {
    width: 50vw;
    height: 50vh;
}

.text-box {
    width: 40ch;
}
```

```
* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
}
```

Box Model

```
.box {  
    margin: 2em;  
    margin-top: 1.5em; /* pareil avec right, left, bottom */  
    margin: 2em 1.5em; /* haut-bas gauche-droite */  
    margin: 2em 1.5em 1em 0.5em; /* haut droite bas gauche */  
    margin: 3em auto; /* centrer horizontalement */  
  
    border: 5px solid green; /* fonctionne aussi pour chaque partie */  
    /* solid / dashed / dotted / double */  
    border-radius: 50px;  
  
    outline: 5px solid purple;  
    outline-offset: 5px; /* peut être négatif */  
}
```

Typographie

```
body {  
    font-size: 2rem;  
}  
  
p, input, button {  
    font: inherit;  
}  
  
p {  
    text-decoration: underline;  
    /* underline / overline / line-through / none */  
    text-transform: capitalize;  
    /* capitalize / lowercase / uppercase */  
    text-align: left;  
    /* left / right / justify */  
    text-indent: 2em;  
    line-height: 1.5;  
    letter-spacing: 0.1em;  
    word-spacing: 0.5em;  
  
    font-weight: bold;  
    /* normal / bold / bolder / light / lighter */  
    font-style: italic;  
    /* italic / oblique / normal */  
}
```

```
    font-family: monospace;
    /* serif / sans-serif / monospace / cursive / fantasy */
}

h1 {
    text-shadow: 2px 2px 5px #000;
}
```

Liens

```
a {
    text-decoration: none;
    cursor: pointer;
    /* pointer / not-allowed */
    color: blue;
}

a:visited {
    color: green;
}

a:hover, a:focus {
    color: dodgerblue;
    opacity: 0.8;
    background: gold;
}

a:active {
    color: red;
}
```

Listes

```
ol {
    list-style-type: lower-alpha;
    /* lower-alpha / decimal / georgian / decimal-leading-zero / disc /
    circle / square / upper-roman */
}

ul {
    text-align: center;
    list-style-position: inside; /* outside (default) */
    list-style-image: url('path/to/image.png')
```

```

        list-style: square url('path/to/image.png') inside;
    }

    li:nth-child(2) { /* 2 / even / odd */
        color: red;
    }

    ::marker {
        font-size: 1.5em;
        color: red;
        font-family: fantasy;
        content: "Content -> "
    }

```

Affichage

```
<p>This is a <span>inline</span> element</p>
```

```

p {
    display: block; /* par défaut */
}

span {
    margin: 1rem; /* Doesn't apply */
    padding: 1rem; /* Apply */

    display: inline-block;
    margin: 1rem; /* Now apply */
}

```

```

<nav>
    <ul>
        <li><a href="#">Accueil</a></li>
        <li><a href="#">A propos</a></li>
        <li><a href="#">Contactez-nous</a></li>
    </ul>
</nav>

```

```

ul {
    list-style-type: none;
    padding-left: 0;
    text-align: right;
}

```

```
}

li {
    display: inline-block;
    margin-inline: 0.5rem;
}

li {
    display: none;
}
```

Flottants

```
<section>
    <div class="block left">Float</div>
    <p>lorem</p>
</section>
<div class="clear"></div>
<p>lorem</p>
```

```
.left {
    float: left;
    margin-right: 1rem;
}

.clear {
    clear: both;
}

section {
    display: flow-root;
}
```

Colonnes

```
<section class="columns">
    <p>...</p>
    <p>...</p>
    <h2>...</h2>
    <p class="quote">Quote <span class="nowrap"> -- Author</span></p>
    <p>...</p>
    <p>...</p>
```

```
</section>
<p>...</p>
```

```
.columns {
  column-count: 4;
  column-width: 250px;
  /* or */
  columns: 4 250px;

  column-rule: 3px solid black;
  column-gap: 2rem;
}

.columns p {
  margin-top: 0;
}

.columns h2 {
  margin-top: 0;
  background-color: lightgrey;
  color: #fff;
  padding: 1rem;
  /* Problem when reducing the size of the screen */
  break-inside: avoid;
  break-before: column; /* if necessary */
}

.columns .quote {
  text-align: center;
  column-span: all;
  margin-top: 2rem;
}

.nowrap {
  white-space: nowrap;
}
```

Positions

```
.static {
  position: static; /* this is the default value of position */
}

#absolute-parent {
  position: relative; /* absolute relative to the closest relative
```



```

parent */
}

.absolute {
    position: absolute;
    top: 0;
    left: 0;
}

.relative {
    position: relative; /* relative to the parent component */
}

.fixed {
    position: fixed;
    top: 100px;
}

.sticky {
    position: sticky;
    top: 0; /* Stay at top 0 until it reached the end of its parent */
}

html {
    scroll-behavior: smooth;
}

```

Flexbox

```

.container {
    display: flex;
    flex-direction: row;
    /* row is default value / column / row-reverse / column-reverse */
    justify-content: center; /* with respect to the main axis */
    /* flex-start / center / flex-end / space-around / space-between /
space-evenly */
    gap: 1rem;
    align-items: center; /* with respect to the secondary axis */
    /* flex-start / center / flex-end / space-around / space-between /
space-evenly */

    flex-direction: row;
    flex-wrap: wrap;
    /* or */
    flex-flow: row wrap;
}

```

```

.box {
  display: flex;
  justify-content: center;
  align-items: center;

  flex-basis: 100px; /* minimum value */
  flex-grow: 1;

  flex-shrink: 1;

  flex: 1 1 250px; /* grow shrink basis */
}

.box:nth-child(2) {
  flex-grow: 2;
  order: 4; /* order: -1; */
}

```

A regarder : flexbox-froggy

Grid Layout

```

.container {
  display: grid;
  grid-auto-flow: row; /* row is the default value / column */

  grid-template-columns: 200px 100px 200px;
  grid-template-columns: 2fr 1fr 2fr;
  grid-template-columns: repeat(4, 1fr);

  grid-auto-rows: 200px;
  grid-auto-rows: minmax(150px, auto);

  /* Rq: the opposite exists, with grid-template-rows and grid-auto-
columns */

  column-gap: 1rem;
  gap: 1rem 2rem;
}

.box:first-child {
  grid-column-start: 1;
  grid-column-end: 4;
  grid-row-start: 1;
  grid-row-end: 3;
  /* or */
  grid-column: 1 / 4;
}

```

```

    grid-row: 1 / 3;

    display: grid;
    align-content: center; /* start / center / end */
    justify-content: center; /* start / center / end */
    /* or */
    place-content: end center; /* or center if both are centered */
}

```

```

body {
    display: grid;
    grid-template-columns: repeat(9, 1fr);
    grid-auto-rows: 75px auto 75px;
    grid-template-areas:
        "hd hd hd hd hd hd hd hd hd"
        "mn mn mn mn mn mn mn sb sb"
        "ft ft ft ft ft ft ft ft ft";
}

.header {
    grid-area: hd;
}

.footer {
    grid-area: ft;
}

.main {
    grid-area: mn;
}

```

A regarder : grid garden

Images

```



```

```

img {
    width: 25%;
    height: auto;
    display: block;

    border: 5px double gray;
    border-radius: 50%;
}

```

```

}

hero {
  background-color: lightyellow;
  background-image: url('../images/background.png')
  background-repeat: no-repeat;
  /* repeat / repeat-x / no-repeat */
  background-size: cover;
  /* cover / contain */
  background-color: hsla(0, 0%, 100%, 0.35);

  background-position: center center; /* bottom / top / right / left
/ center */
  background-image: linear-gradient(to left, steelblue, #fff);
  /* possible to cumulate linear-gradient and url for background-
image */

  background-image: radial-gradient(steelblue, #fff);
}

```

Media Queries

```

/* Syntax
@media media type and (condition: breakpoint) and ... {
    CSS rules
}
*/

@media screen and (min-width: 481px) {}
@media screen and (min-aspect-ratio: landscape) {} /* 16/9 7/4 */

```

Pseudo elements

```

nav a:hover, nav a:focus {} = nav is:(a:hover, a:focus)

.card:target {} /* last target element from link */
.card img[alt] {} /* image with alt attribut */
.card img:not([alt]) {}

.card:nth-child(2) {} /* even / odd */

.card figcaption::before {
  content: "Hello ";
}

```

```
}

.card p::before {
    content: open-quote;
}

.card p::after {
    content: close-quote;
}

.paragraph::first-letter {
    font-size: 2rem;
    color: red;
    font-weight: bold;
}

.paragraph::first-line {
    color: red;
    font-style: italic;
}

.paragraph::selection {
    background-color: red;
    color: white;
}
```

Variables

```
:root {
    --font-family: "Source Code Pro";
    --background-color: lightblue;
}

body {
    background-color: var(--background-color);
    font-family: var(--font-family);
}

@media (prefers-color-scheme: dark) { /* light */
    :root {
        --background-color: darkblue;
    }
}
```

Functions

```

p {
    font-size: min(2.5rem, 3vh);
    font-size: max(2.5rem, 3vh);
    font-size: clamp(1.75rem, 3vh, 2.5rem);
}

main {
    width: calc(30% - 5px);
}

a:hover {
    filter: brightness(150%);
    filter: hue-rotate(180deg);
}

```

Animation

```

div:hover {
    background-color: midnightblue;
    transform: rotate(180deg);
    transition-property: background-color, transform;
    transition-duration: 2s, 3s;
    transition-delay: 0.5s;
    transition-timing-function: ease; /* ease / ease-in / ease-out /
ease-in-out / linear */
    /* or */
    transition: all 2s ease-in-out 0.5s;
}

.animate {
    animation-name: slide;
    animation-duration: 5s;
    animation-timing-function: ease-in-out;
    animation-delay: 1s;
    animation-iteration-count: 5;
    animation-direction: reverse; /* normal / alternate / alternate-
reverse */
    animation-fill-mode: forwards;
}

@keyframes slide {
    0% {
        transform: translateX(0);
    }
    33% {
        transform: translateX(600px) rotate(180deg);
    }
}

```

```
}
66% {
    transform: translateX(-600px) rotate(-180deg);
}
100% {
    transform: translateX(0);
    background-color: purple;
}
}
```

Data selector

```
<div data-sometext=["Some Text"]></div>
```

```
[data-sometext] {
    position: relative;
}

[data-sometext]::before {
    content: attr(data-sometext);
    position: absolute;
    ...
}
```

Transform

```
.box {
    transform: 50px;
    /* or */
    transform: 50px 100px;

    scale: .5;

    rotate: x|y|z 45deg;
}

/* With animation */
```

```
.box.hover {  
    animation: rotate 500ms infinite alternate;  
}  
  
@keyframes rotate {  
    0% {  
        rotate: 0;  
    }  
  
    100% {  
        rotate: 45deg;  
    }  
}
```

Pseudo éléments

```
/* Before et After */  
.paragraph::before {  
    content: "Avant le paragraphe"  
}  
  
.paragraph:after {  
    content: "Après le paragraphe"  
}  
  
/* Sur du texte */  
.paragraph::first-letter {  
    font-size: 2rem;  
    color: red;  
    font-weight: bold;  
}  
  
.paragraph::first-line {  
    color: red;  
    font-style: italic;  
}  
  
.paragraph::selection {  
    background-color: red;  
    color: white;  
}
```