

# CSS : interface *réactive*

**Tristan Crolard**

Laboratoire CEDRIC  
Equipe « Systèmes Sûrs »

**`tristan.crolard@cnam.fr`**

**`cedric.cnam.fr/sys/crolard`**

# Interface web *réactive* (*responsive*)

L'interface d'un site web peut-être :

- ▶ **statique** : dimensions figées quelle que soit la surface de la zone d'affichage.
- ▶ **fluide** : largeurs de colonnes exprimées en unités variables (pourcentages, *em*, *vw*, etc.) et s'adapte à la taille de la zone d'affichage
- ▶ **adaptive** : largeurs de colonnes fixes, mais différentes selon la largeur de la zone d'affichage (utilise les *Media Queries* de CSS 3), généralement pour un nombre fixe de gabarits (par exemple, largeur : *small*  $\leq 768\text{px}$ , *medium*  $\leq 1536\text{px}$ , et *large*) qui définissent les points de rupture (*breakpoints*).
- ▶ **réactive** (*responsive*) : en même temps **fluide** et **adaptive**.

## Unités de longueur absolue

La liste qui suit contient uniquement des unités de longueur **absolue**. Ces quantités ne sont pas relatives à quoi que ce soit d'autre et leur taille sera considérée comme constante.

### In this article

Qu'est-ce qu'une valeur CSS ?

**Nombres, longueurs et pourcentages**

Couleur

Images

Position

Chaînes de caractères et identifiants

Fonctions

Testez vos compétences !

Résumé

Dans ce module

### Related Topics

Unité	Nom	Équivalent à
cm	Centimètres	1cm = 38px = 25/64in
mm	Millimètres	1mm = 1/10th of 1cm
Q	Quarts de millimètre	1Q = 1/40th of 1cm
in	Pouces (inches)	1in = 2.54cm = 96px
pc	Picas	1pc = 1/6e de 1in
pt	Points	1pt = 1/72e de 1in
px	Pixels	1px = 1/96th de 1in

## In this article

Qu'est-ce qu'une valeur CSS ?

**Nombres, longueurs et pourcentages**

Couleur

Images

Position

Chaînes de caractères et identifiants

Fonctions

Testez vos compétences !

Résumé

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).

## Unités de longueur relative

Les unités de longueur relative permettent d'exprimer des quantités relatives à quelque chose d'autre comme la taille de la police de l'élément parent ou la taille de la zone d'affichage (viewport). L'avantage d'utiliser des unités relatives est qu'avec un peu d'organisation, on peut faire que la taille du texte ou d'autres éléments se mette à l'échelle, relativement à quelque chose d'autre sur la page. La plupart des unités les plus utiles pour le développement web sont présentes dans le tableau qui suit.

## In this article

Qu'est-ce qu'une valeur CSS ?

**Nombres, longueurs et pourcentages**

Couleur

Images

Position

Chaînes de caractères et identifiants

Fonctions

Testez vos compétences !

Résumé

Dans ce module

## Related Topics

Unité	Relative à
em	Pour les propriétés typographiques comme <a href="#">font-size</a> , relative à la taille de la police de l'élément parent. Pour les autres propriétés comme <a href="#">width</a> , relative à la taille de la police de l'élément;
ex	La hauteur d'x de la police de l'élément.
ch	La <a href="#">chasse/avance</a> du glyphe « 0 » pour la police de l'élément.
rem	La taille de la police pour l'élément racine.
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).

# Interface d'un site web

## Principes de base d'ergonomie :

- ▶ *scroller* de manière horizontale n'est pas naturel (la largeur de l'écran est donc importante, sa hauteur moins)
- ▶ *zoomer* n'est pas pratique, et donc à éviter (cf. point précédent)
- ▶ les polices de caractères peuvent être plus petite sur smartphone/tablette de 20% (mais pas moins)
- ▶ l'interface doit être utilisable avec un écran tactile (moins précis)

# Manipulation d'un smartphone<sup>1</sup>



49%



36%



15%

*Smartphone use is defined by three basic handholds, and we often shift among them.*

---

**1.** <https://alistapart.com/article/how-we-hold-our-gadgets>

# Conception réactive<sup>2</sup>



---

2. [https://developer.mozilla.org/fr/docs/Learn/CSS/CSS\\_layout/Responsive\\_Design](https://developer.mozilla.org/fr/docs/Learn/CSS/CSS_layout/Responsive_Design)



# Media queries<sup>3</sup>



The screenshot shows a web browser window with the address bar displaying "developer.mozilla.org". The page title is "Responsive design - Apprendre le développement web | MDN". The main heading is "Media Queries". The text explains that responsive design became a Recommendation Candidate in 2009 and that media queries allow for selective CSS application based on user needs. An example is provided: a media query for screen with a minimum width of 800px, which applies a margin of 1em 2em to the ".container" class. The text concludes by stating that multiple media queries can be used in a style sheet to adapt the page to different screen sizes, with the points where changes occur called "breakpoints".

## Media Queries

Le responsive design n'a pu voir le jour uniquement grâce aux média queries. La spécification de niveau 3 des média queries est devenue une Recommandation Candidate en 2009, ce qui signifie qu'elle a été jugée prête à être mise en œuvre dans les navigateurs. Les Media Queries nous permettent d'effectuer une série de tests (par exemple, si l'écran de l'utilisateur dépasse une certaine largeur, ou une certaine résolution) et d'appliquer le CSS de manière sélective pour donner à la page le style approprié aux besoins de l'utilisateur.

Par exemple, la média query suivante teste si la page Web actuelle est affichée comme média d'écran (donc pas un document à imprimer) et si la fenêtre de visualisation a au moins 800 pixels de large. Le CSS du sélecteur `.container` ne sera appliqué que si ces deux éléments sont vrais.

```
@media screen and (min-width: 800px) {  
  .container {  
    margin: 1em 2em;  
  }  
}
```

Vous pouvez ajouter plusieurs medias queries dans une feuille de style, afin de modifier votre mise en page ou certaines de ses parties pour les adapter aux différentes tailles d'écran. Les endroits où une média query est introduite et où la mise en page est modifiée sont appelés points d'arrêt.

3. [https://developer.mozilla.org/fr/docs/Learn/CSS/CSS\\_layout/Responsive\\_Design#media\\_queries](https://developer.mozilla.org/fr/docs/Learn/CSS/CSS_layout/Responsive_Design#media_queries)

# Flexbox<sup>4</sup>

Table of contents

- Pourquoi Flexbox ?
- Voici un exemple simple
- Détermination des éléments à disposer en boîtes flexibles
- Aparté sur le modèle flex
- Colonnes ou lignes ?
- Enveloppement
- Forme abrégée « flex-flow »
- Taille modulable des éléments flex
- Flex : forme abrégée vs forme longue
- Alignement horizontal et vertical
- Ordonner les éléments flex
- Boîtes flex imbriquées
- Compatibilité des navigateurs
- Résumé

Related Topics

[Bienvenue aux débutants !](#)

## Flexbox

[Précédent](#) [Aperçu : CSS layout](#) [Suivant](#)

Flexbox est une méthode de mise en page selon un axe principal, permettant de disposer des éléments en ligne ou en colonne. Les éléments se dilatent ou se rétractent pour occuper l'espace disponible. Cet article en explique tous les fondamentaux.

<b>Prérequis :</b>	Les fondamentaux du HTML (étudiez <a href="#">Introduction au HTML</a> ) et avoir une idée de la manière dont la CSS fonctionne (étudiez <a href="#">Introduction aux CSS</a> ).
<b>Objectif :</b>	Apprendre à utiliser le système Flexbox pour créer des mises en page web.

## Pourquoi Flexbox ?

Pendant longtemps, les seuls outils de mise en page CSS fiables et compatibles avec les navigateurs, étaient les propriétés concernant les [flotteurs \(boîtes flottantes\)](#) et le [positionnement](#). Ces outils sont bien et fonctionnent, mais restent à certains égards plutôt limitatifs et frustrants.

Les simples exigences de mise en page suivantes sont difficiles si ce n'est impossibles à réaliser de

4. [https://developer.mozilla.org/fr/docs/Learn/CSS/CSS\\_layout/Flexbox](https://developer.mozilla.org/fr/docs/Learn/CSS/CSS_layout/Flexbox)

# Grilles<sup>5</sup>

developer.mozilla.org

Grilles - Apprendre le développement web | MDN

## Table of contents

- Qu'est qu'une disposition en trame ?
- Création d'une trame CSS
- Placement sur les lignes
- Placer avec « grid-template-areas »
- « CSS Grid » : une structure de quadrillage
- Résumé
- Voir également
- Dans ce module

## Related Topics

**Bienvenue aux débutants !**

- ▶ Commencer avec le Web

**HTML**

- ▶ Introduction au HTML
- ▶ Multimedia and embedding

# Grilles

[Précédent](#) [Aperçu : CSS layout](#) [Suivant](#)

« CSS Grid Layout » (Trames avec les CSS) est un système de mise en page bidimensionnel. Il vous permet de disposer les contenus en lignes et en colonnes ; il possède de nombreuses fonctionnalités simplifiant la construction de mises en page complexes. Cet article vous indique tout ce que vous devez savoir pour commencer une mise en page avec une trame.

<b>Prérequis :</b>	Les fondamentaux du HTML (étudiez <a href="#">Introduction au HTML</a> ) et une idée de la manière dont la CSS fonctionne (étudiez <a href="#">Introduction aux CSS</a> et <a href="#">Styles de boîtes</a> ).
<b>Objectif :</b>	Comprendre les concepts fondamentaux sous-jacents de la disposition en trame et comment la mettre en œuvre en utilisant « CSS Grid ».

## Qu'est qu'une disposition en trame ?

Une trame est simplement un ensemble de lignes horizontales et verticales créant un quadrillage dans lequel nous pouvons agencer les éléments à afficher. Elles nous aident à créer des compositions dans lesquelles les éléments ne sautent pas ou ne changent pas de largeur au fur et à mesure que nous nous