

CSS3 : Flexbox

- **Introduction et historique**
- **Principes**
- **Flex-container et flex-item**
- **Propriétés du flex-container**
- Propriétés du flex-item
- Applications pratiques
- Autres ressources

Introduction

- Objectifs de **CSS3 Flexible Box Layout** :

fournir un moyen efficace de disposer, aligner et distribuer l'espace entre les items d'un container, même lorsque leurs dimensions sont inconnues et/ou dynamiques

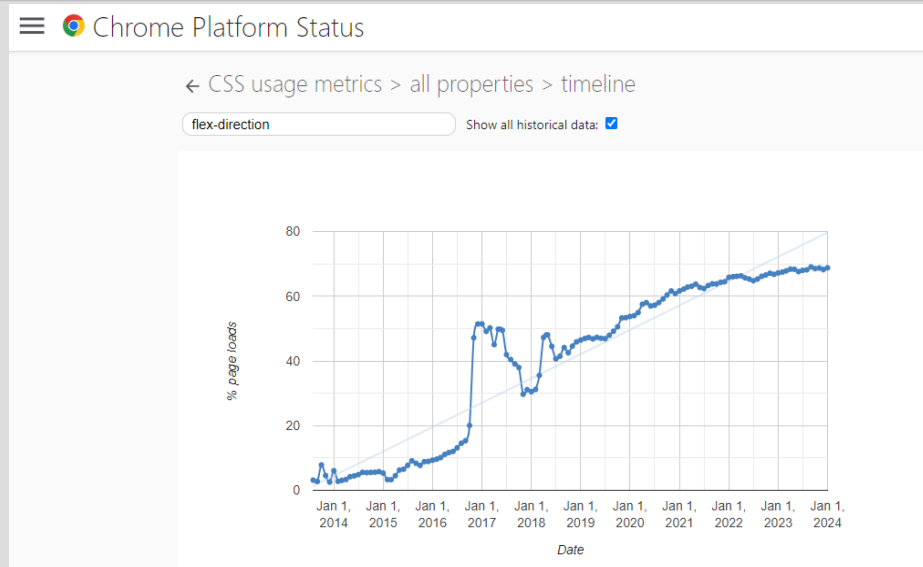
donner au **container** la possibilité de changer les largeur et hauteur des éléments contenus (**items**), afin de remplir au mieux l'espace, et de **s'adapter** à **tous les terminaux** et toutes les **tailles d'écran**

- **Attention, il y a eu plusieurs versions !**

Historique

- 2003 : premiers essais sur Apple Safari
- 2006 : spécifications revues chez Mozilla Firefox, Google Chrome et Microsoft
- 2009 : premier brouillon W3C (display:box)
- 2011 : deuxième version (display:flexbox) avec implémentation pour IE10 (display:-ms-flexbox)
- **2015** : version officielle (**display:flex**)
 - https://developer.mozilla.org/fr/docs/Web/CSS/CSS_flexible_box_layout
 - https://www.w3schools.com/css/css3_flexbox.asp

Utilisation



CSS Flexible Box Layout Module - CR

Baseline Widely available across major browsers

Method of positioning elements in horizontal or vertical stacks. Support includes all properties prefixed with `flex`, as well as `display: flex`, `display: inline-flex`, `align-content`, `align-items`, `align-self`, `justify-content` and `order`.

Current aligned Usage relative Date relative Filtered All

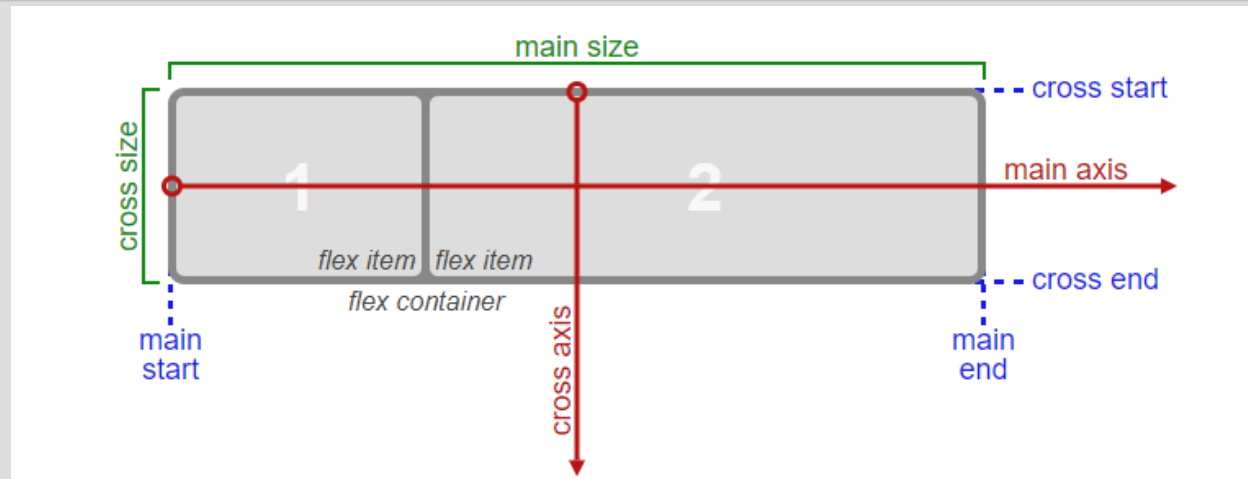
Chrome	Edge	Safari	Firefox	Opera	IE	Chrome for Android	Safari on iOS	Samsung Internet	Opera Mini	Opera Mobile	UC Browser for Android	Android Browser	Firefox for Android	QQ Browser	Baidu Browser	KaiOS Browser
				10-11.5												
1 4-20		1 3.1-6	1 2-21	12.1			1 3.2-6.1									
21-28		6.1-8	3 22-27	15-16	6-9		7-8.4			12		1 2.1-4.3				
29-119	12-119	9-17.1	28-120	17-105	2 10		9-17.1	4-22		12.1		4.4-4.4.4				
120	120	17.2	121	106	4 11	120	17.2	23	all	73	15.5	120	119	13.1	13.18	2.5
121-123	121	17.3-TP	122-124				17.3									

Usage % of all users ?

Global 98.01% + 0.53% = 98.54%

unprefixed: 97.99% + 0.42% = 98.41%

Principes



- Un élément parent (**flex-container**) va contenir plusieurs éléments enfants (**flex-item**).
- Les items sont disposés soit sur l'axe principal (**main axis**) entre **main-start** et **main-end** ou sur l'axe secondaire (**cross axis**) entre **cross-start** et **cross-end** qui est perpendiculaire

Propriétés Flexbox

- **Flex-container**

- display
- flex-wrap
- flex-direction
- flex-flow
- justify-content
- align-items
- align-content

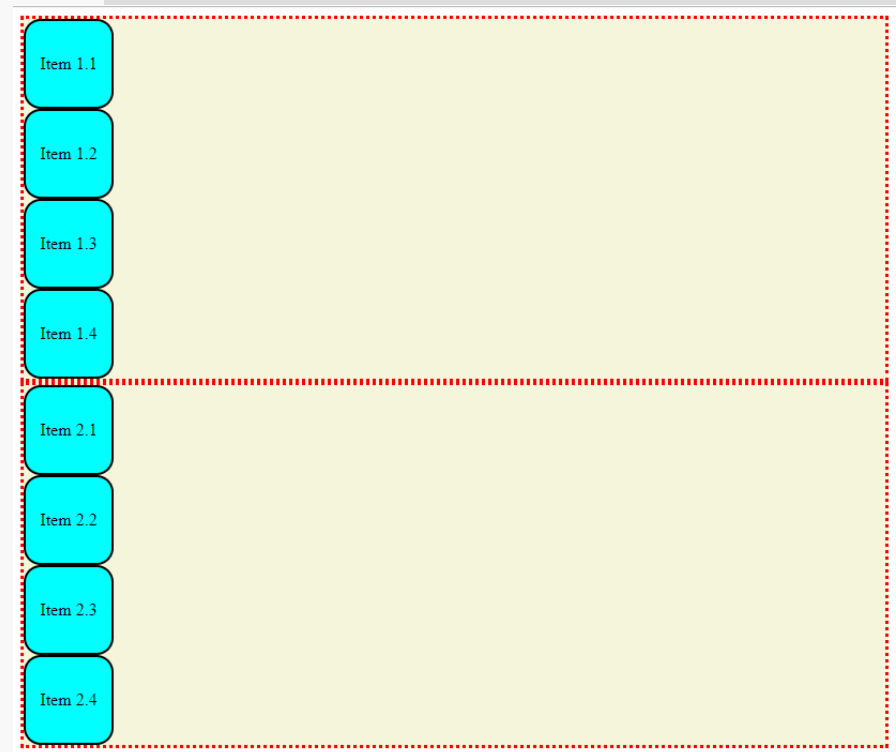
- **Flex-item**

- flex-grow
- flex-shrink
- flex-basis
- flex
- order
- align-self

Code de base

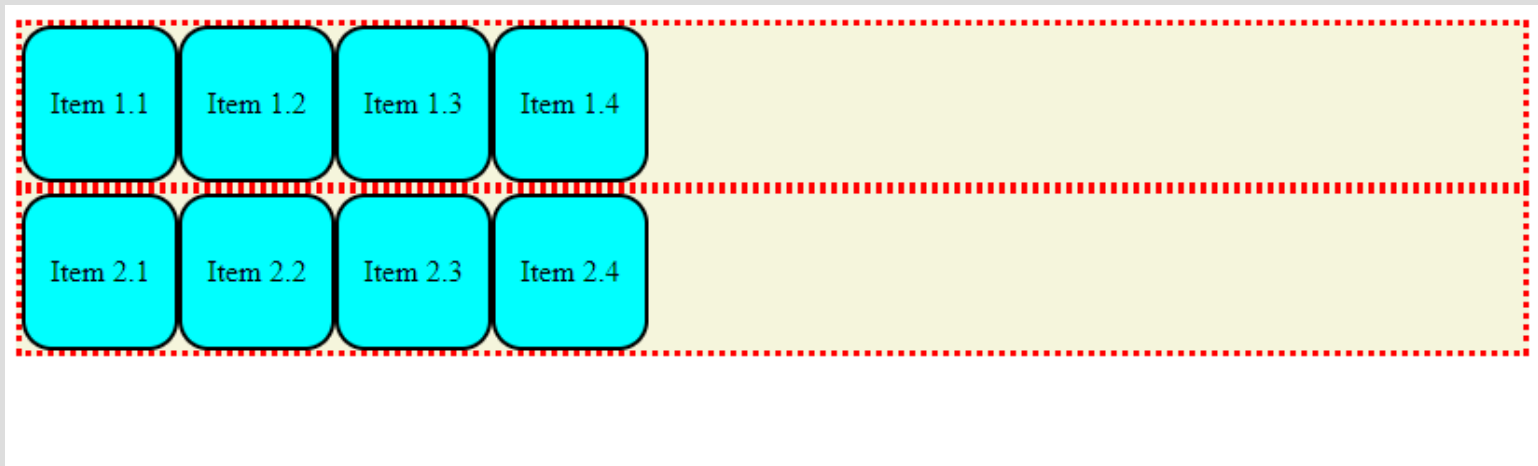
```
<html lang="fr">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Flexbox</title>
    <style>
      section {
        background-color: beige;
        border: 3px dotted red;
      }

      article {
        width: 5em;
        min-height: 5em;
        line-height: 5em;
        border: 2px solid black;
        border-radius: 1em;
        text-align: center;
        background-color: aqua;
      }
    </style>
  </head>
  <body>
    <section class="parent">
      <article class="enfant">Item 1.1</article>
      <article class="enfant">Item 1.2</article>
      <article class="enfant">Item 1.3</article>
      <article class="enfant">Item 1.4</article>
    </section>
    <section class="parent">
      <article class="enfant">Item 2.1</article>
      <article class="enfant">Item 2.2</article>
      <article class="enfant">Item 2.3</article>
      <article class="enfant">Item 2.4</article>
    </section>
  </body>
</html>
```

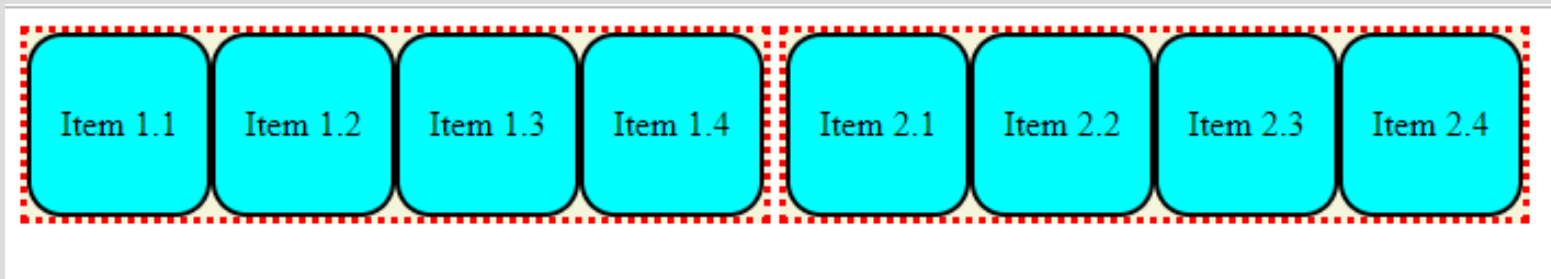


display

- `display: flex;` container positionné en block

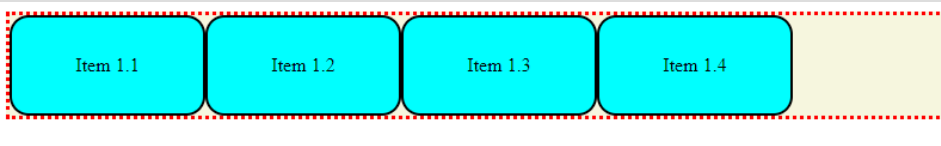


- `display: inline-flex;` container positionné en inline-block

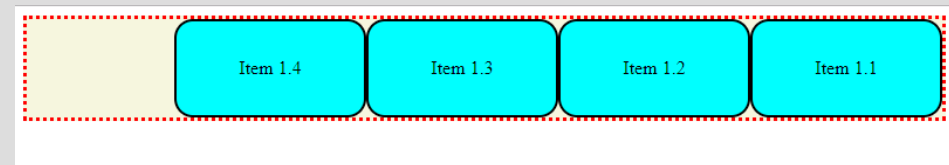


flex-direction

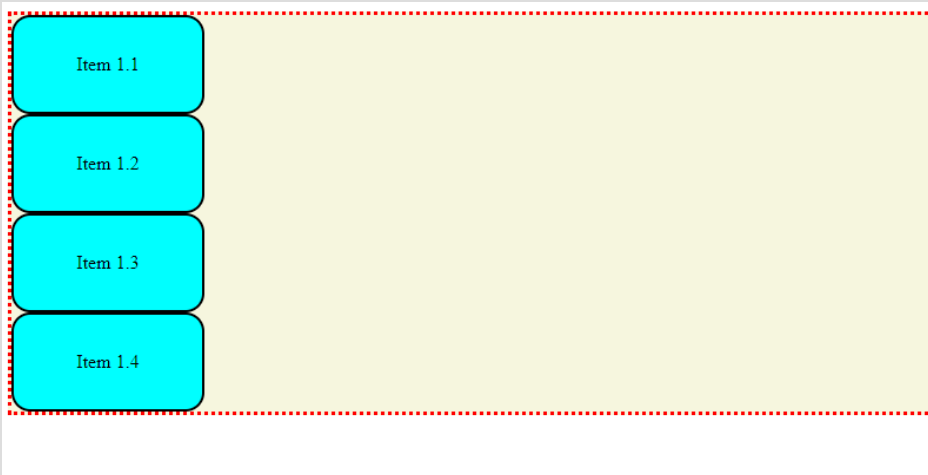
- **row** (par défaut)



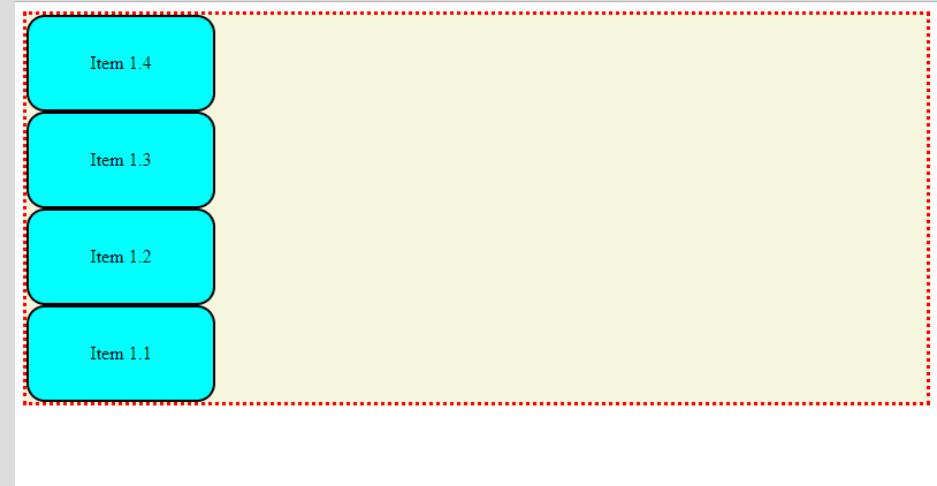
- **row-reverse**



- **column**

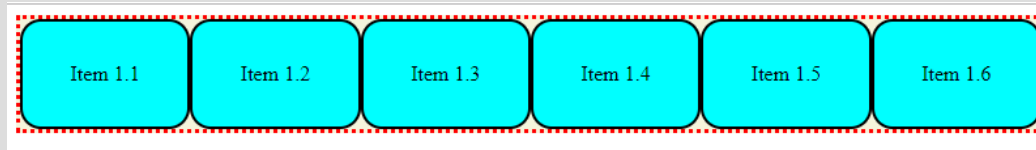


- **column-reverse**

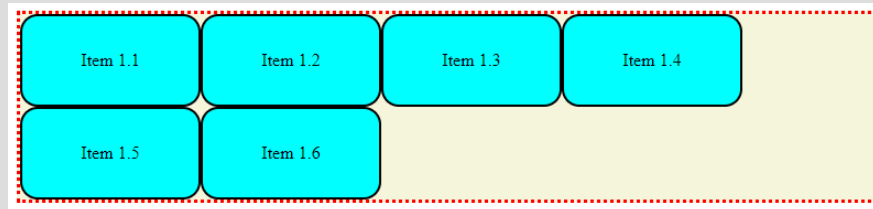


flex-wrap

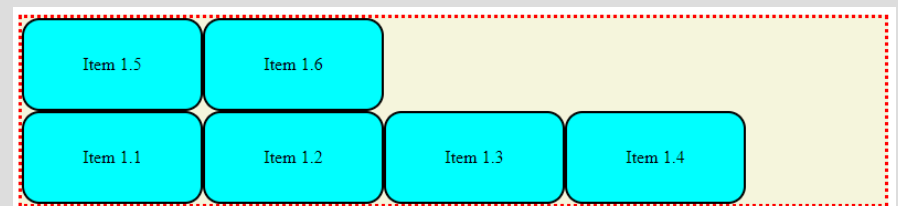
- **nowrap** (par défaut)



- **wrap**



- **wrap-reverse**



flex-flow

- **flex-flow** est la propriété qui regroupe, séparées par un espace et dans l'ordre, les propriétés **flex-direction** et **flex-wrap**
- La valeur **par défaut** est donc :
flex-flow : row nowrap;