

HTML et CSS

Introduction

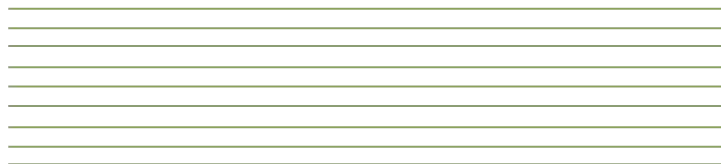
HTML - CSS : Langages informatiques

Indispensables pour des projets web

Dernières versions : HTML5 et CSS3



Définitions et usages



HTML :

- 1991
- HyperText Markup Language / Langage de Balisage Hypertexte
- Donner du sens et structurer les contenus

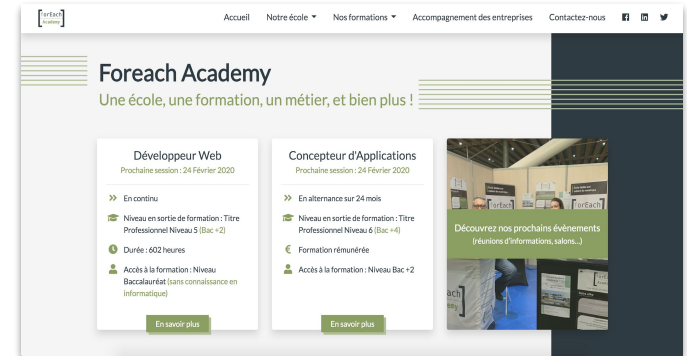
CSS :

- 1996
- Cascading StyleSheets / Feuilles de Styles en Cascade
- Mettre en forme les contenus

Fonctionnement d'une page web

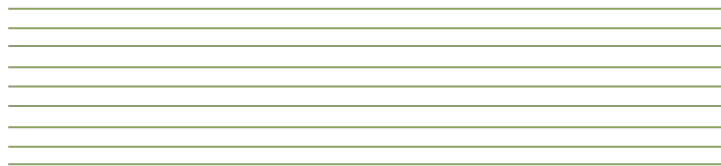
```
94 </head>
95 <body class="">
96
97 <nav id="menu" class="navbar navbar-expand-lg shadow fixed-top">
98   <a href="/" id="logo"></a>
99
100   <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbar" aria-controls="navbar" aria-expanded="false" aria-label="Toggle navigation" data-cs="2" data-kind="parent">
101     <span class="fas fa-bars"></span>
102   </button>
103   <div class="collapse navbar-collapse" id="navbar">
104     <ul class="navbar-nav ml-auto">
105       <li class="nav-item selected">
106         <a href="/" class="nav-link animate-link">
107           Accueil
108         </a>
109       </li>
110       <li class="nav-item dropdown">
111         <a href="#" class="nav-link dropdown-toggle" id="ecole" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false" data-level="1">
112           Notre école
113         </a>
114         <div class="dropdown-menu subnavbar" aria-labelledby="ecole">
115           <ul class="nav nav-level-1">
116             <li class="nav-item">
117               <a href="/ecole/presentation" class="nav-link animate-link">
118                 Présentation
119             </li>
120             <li class="nav-item">
121               <a href="/ecole/financement" class="nav-link animate-link">
122                 Financement
123             </li>
124             <li class="nav-item">
125               <a href="/ecole/formation" class="nav-link animate-link">
126                 Nos formations
127             </li>
128             <li class="nav-item">
129               <a href="/ecole/accompagnement" class="nav-link animate-link">
130                 Accompagnement des entreprises
131             </li>
132             <li class="nav-item">
133               <a href="/ecole/contact" class="nav-link animate-link">
134                 Contactez-nous
135             </li>
136           </ul>
137         </div>
138       </li>
139       <li class="nav-item dropdown">
140         <a href="#" class="nav-link dropdown-toggle" id="formations" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false" data-level="1">
141           Nos formations
142         </a>
143         <div class="dropdown-menu subnavbar" aria-labelledby="formations">
144           <ul class="nav nav-level-1">
145             <li class="nav-item">
146               <a href="/formation-developpeur-web-lille" class="nav-link animate-link">
147                 Développeur Web
148             </li>
149             <li class="nav-item">
150               <a href="/formation-concepteur-applications-lille" class="nav-link animate-link">
151                 Concepteur d'Applications
152             </li>
153           </ul>
154         </div>
155       </li>
156     </ul>
157   </div>
158 </nav>
```

Page HTML



HTML + CSS

Outils



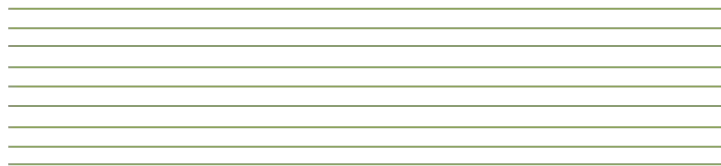
Éditeur de texte :



Navigateur :



HTML : Les fondamentaux



Éléments : Définit un objet dans notre page (titre, paragraphe, lien...)

Balises: Permet de hiérarchiser et de structurer le contenu

- Balise ouvrante `< .. >`
- Balise fermante `</ .. >`
- Balise orpheline `< .. / >`

Attributs : Information supplémentaire placée dans une balise

Exemples de balises

élément

`<p>Je suis un paragraphe</p>`

balise ouvrante balise fermante

élément

``

attribut

balise orpheline

Structure d'une page HTML5

page.html

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8">
    <title>Titre de la page</title>
  </head>
  <body>
    <!-- Le contenu de la page -->
  </body>
</html>
```

Bonnes pratiques:

- Imbrication
`<a><a/>`
- Indentation
`<a>`
 ``
`<a/>`
- Commentaire
`<!-- Commentaire -->`

Les différents éléments

Textes

- Titre : h1, h2, h3, h4, h5, h6
- Paragraphe: p
- Retour à la ligne : br
- Contenu (niveau d'importance): strong, em, mark

Dans le page HTML :

1. Intégrer un titre principal et un titre secondaire
2. Intégrer un paragraphe avec un retour à la ligne
3. Dans le paragraphe intégrer un un mot important, un mot semi-important et un mot en surlignage

Les différents éléments

Liste

- Liste non ordonnée : ul
- Liste ordonnée : ol
- Élément de la liste : li

```
<ul>  
  <li>Formation 1</li>  
  <li>Formation 2</li>  
</ul>
```

Dans le page HTML :

1. Intégrer une liste ordonné
2. Intégrer une liste non ordonné

Les différents éléments

Liens

- Externe
- Interne
- Mail / Téléphone
- Téléchargement

```
<a href="#" aria-label="nom du lien">  
    Nom lien  
</a>
```

Dans le page HTML :

1. Intégrer un un lien externe et un lien interne
2. Intégrer un numéro de téléphone et un e-mail
3. Intégrer un fichier à télécharger

Les différents éléments

Tableaux

- table
- tr, td et th
- Structure: thead, tbody, tfoot
- colspan, rowspan

```
<table>
  <tr>
    <td>Lille</td>
    <td>Arras</td>
  </tr>
</table>
```

Ville	Département	Matin	Après-midi
Lille	59	Pluie	Nuageux
Calais		Ensoleillé	
Arras	62	Pluie	Averses

Les différents éléments

Médias

- Image : img
- Audio
- Vidéo
- iframe

```

<figure>
  
  <figcaption>Ma légende</figcaption>
</figure>
```

Dans le page HTML :

1. Intégrer une image avec un lien externe et interne
2. Intégrer une image avec une légende
3. Intégrer une vidéo Youtube

Les éléments div et span

Conteneurs génériques

Aucunes valeurs sémantiques

Simplifier la mise en page

```
<div class="ma-premiere-div">  
    
  <p>Mon <span>super</span> paragraphe</p>  
</div>
```

Les formulaires

Form

- méthode GET ou POST
- action

```
<form method="post" action="form.php">  
  <input type="text" name="nom" >  
  <input type="password" name="motDePasse">  
  <input type="email" name="email">  
</form>
```

Dans le page HTML :

1. Intégrer un formulaire avec la méthode post et qui redirigera vers form.php
2. Intégrer un champ texte, un champ mot de passe et un champ email

Les formulaires

Label

- for
- id

```
<label for="pseudo">Entrez un nom : </label>  
<input type="text" id="pseudo">
```

Placeholder

Dans le page HTML :

1. Intégrer un formulaire avec la méthode post et qui redirigera vers form.php
2. Intégrer un champ texte, un champ mot de passe et un champ email

Les formulaires

Bouton envoyer

- type

```
<input type="submit" value="Envoyer">
```

Dans le formulaire :

1. Intégrer un bouton “envoyer”

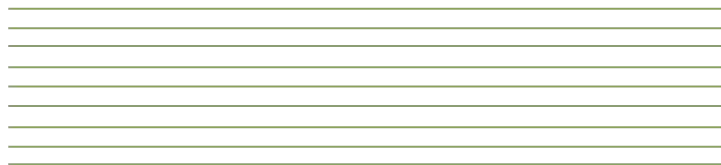
Les formulaires : type d'input

- text
- number
- email
- date
- password
- checkbox
- radio
- url
- file
- submit

Les formulaires : contraintes sur les données

- size
- minlength
- maxlength
- min
- max
- step
- autocomplete
- required
- pattern

Accessibilité



Rendre l'information et le contenu du site accessibles aux personnes handicapées quelque soit la nature du handicap et quelque soit la façon dont c'est consulté

Pour qui ?

- Déficients visuels
- Déficients auditifs
- Déficients moteurs
- Déficients cognitifs
- Personnes âgées

Dans quel but ?

Donner accès à l'information à tous les utilisateurs

Mais aussi pour

- Le développement
- La responsivité
- Le SEO

Accessibilité

Entrez un nom :



```
<label>Entrez un nom : </label>  
<input type="text">
```

Modifier le texte
- vierge



```
<label for="nom">Entrez un nom : </label>  
<input type="text" id="nom">
```

Entrez un nom -
modifier le texte

Accessibilité



```
<a href="#"><i class="fab fa-facebook-square"></i></a>  
<a href="#"><i class="fab fa-twitter-square"></i></a>  
<a href="#"><i class="fab fa-linkedin"></i></a>
```

lien,
lien,
lien



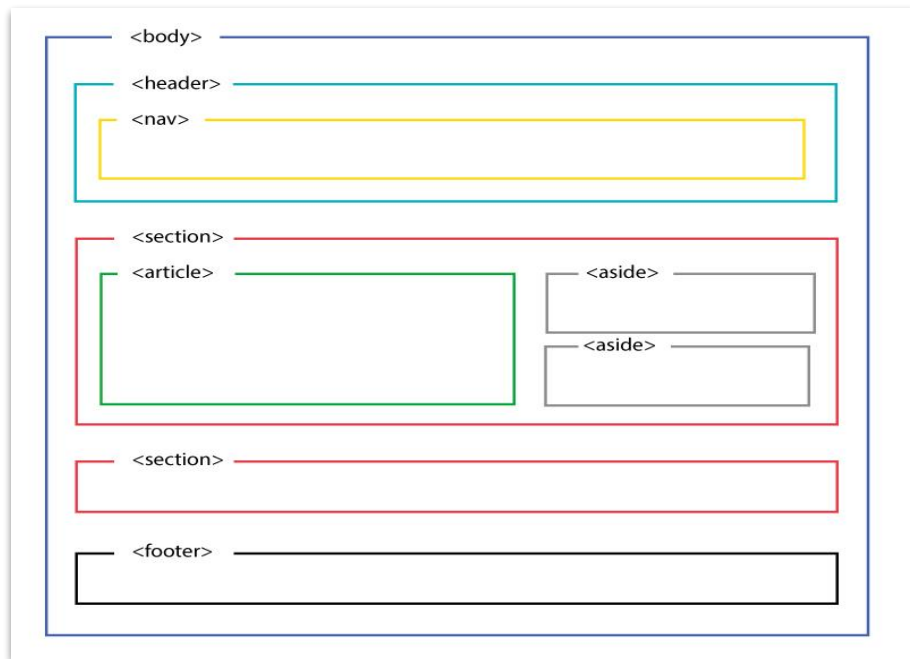
```
<a href="#" aria-label="page Facebook">  
  <i class="fab fa-facebook-square"></i>  
</a>
```

lien -
page Facebook

Accessibilité : la sémantique

Éléments structurants

- header
- nav
- main
- section
- article
- aside
- footer



Accessibilité

Définir la langue du site

Hiérarchiser les titres : h1 -> h2 -> h3
/!\ pas pour styliser !

UX Design :

- Contraste couleurs
- Espacements des liens / boutons
- Actions
- ...

<https://www.atalan.fr/agissons/fr>

<http://www.accessiweb.org>

<https://app.contrast-finder.org>

Normes W3C

W3C : World Wide Web Consortium
Organisme de standardisation

But : Le Web accessible pour tous quelque
soit leur matériel, logiciel, réseau, langue ou
handicap



<https://www.w3.org/>
<https://validator.w3.org/>

CSS : Les sélecteurs

Sélecteurs : définissent les éléments sur lesquelles s'applique un ensemble de règles CSS

- Sélecteurs de type : **nomElement**
- Sélecteurs de classe : **.nomClasse**
- Sélecteurs d'identifiant : **#nomIdentifiant**
- Sélecteur universel : *****
- Sélecteur d'attributs : **[attr]**

```
p {  
  color: red;  
}  
  
.texte {  
  color: red;  
}  
  
#contenu {  
  color: red;  
}  
  
* {  
  color: red;  
}  
  
a[href="www.site.fr"] {  
  color: red;  
}
```

Les propriétés

Propriétés : accompagnés d'une ou plusieurs valeurs pour définir le comportement ou l'aspect d'un éléments HTML

```
selecteur {  
    propriété: valeur(s) ;  
    propriété: valeur(s) ;  
    propriété: valeur(s) ;  
}
```

```
p {  
    color: red;  
    font-size: 20px;  
    padding: 10px;  
}
```

Bonnes pratiques:

- Indentation
- Commentaire

```
/* Commentaire */
```

!! Navigateur !!
caniuse.com

Déclarations

1. Élément *style*

Placé dans la page HTML dans l'élément *head*

```
<head>
...
<style>
  p {
    color: red;
  }
</style>
</head>
```

2. Attribut *style*

Placé dans la page HTML dans la balise ouvrante d'un l'élément

```
<p style="color: red;">
  Voilà un paragraphe rouge
</p>
```

3. Fichier séparé

Placé dans la page HTML le lien vers le fichier .css

```
<head>
...
<link href="./style.css" rel="stylesheet">
</head>
```

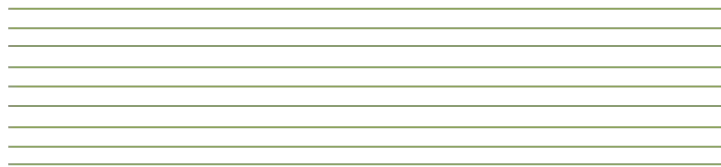
Class et id

- Class
- Id

Exercices

1. Créer une page CSS
2. Ajouter le lien dans votre page HTML
3. Définir une taille et la couleur en bleu des paragraphes
4. Définir la couleur d'une classe "*texte*" dans une div en vert
5. Définir la couleur en jaune d'un paragraphe avec un id "*paragraphe-jaune*"

Sélecteurs complexes



Sélecteurs complexes / combineurs :

- **A, B** : sélectionne les éléments A et B
- **A B** : sélectionne les éléments B dans les éléments A
- **A > B** : sélectionne les éléments B qui sont directement situés sous un élément A
- **A + B** : sélectionne l'élément B qui suit immédiatement un élément A
- **A ~ B** : sélectionne les éléments B qui suivent (immédiatement ou non) un élément A et qui ont le même élément parent

Exercices

1. Définir la taille et la couleur en bleu des paragraphes et des titres
2. Définir la couleur en jaune pour les titres dans une balise header
3. Définir une couleur bleu d'un paragraphe dans une div avec la classe "contenu"

Les propriétés à connaître

Propriétés de mise en forme du texte :

font-family	Police
font-size	Taille
font-weight	Epaisseur caractère
text-transform	Capitales
text-align	Alignement horizontal
line-height	Interligne
color	Couleur

Les propriétés à connaître

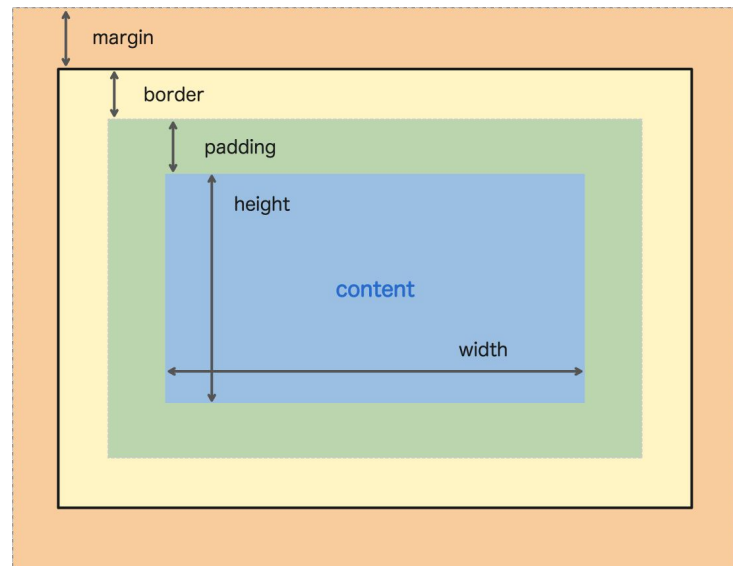
Propriétés de fond :

background-color	Image de fond
background-image	Couleur de fond
background-size	Taille du fond
background-repeat	Répétition du fond
background-position	Position du fond
background	Propriété de fond (combine)

Les propriétés à connaître

Propriétés des éléments

height	Hauteur
width	Largeur
min-height / min-width	Hauteur / Largeur minimum
max-height / max-width	Hauteur / Largeur maximum
margin	Marge
padding	Marge intérieur
border	Bordure



Les propriétés à connaître

Propriétés de positionnement (position)

static (<i>par défaut</i>)	Les éléments se positionnent les uns à la suite des autres, selon leur ordre dans le document
absolute	La position de l'élément est définie par les propriétés top, right, bottom et left en fonction de son parent
fixed	La position de l'élément est définie par les propriétés top, right, bottom et left et ne défile pas avec la page
relative	La position de l'élément est définie par les propriétés top, right, bottom et left par rapport à la position qu'il aurait eu en mode statique

Les propriétés à connaître

Propriétés d'affichage (display)

none	L'élément ne s'affiche pas
block	L'élément commence sur une nouvelle ligne et s'étire autant qu'il peut sur la gauche et la droite
inline	L'élément s'inscrit dans le flux , donc il ne 'saute' pas à la ligne comme le ferait un élément block
inline-block	L'élément est inline tout en préservant sa capacité d'élément block
flex, inline-flex, table, list-item, grid ...	

Les propriétés à connaître

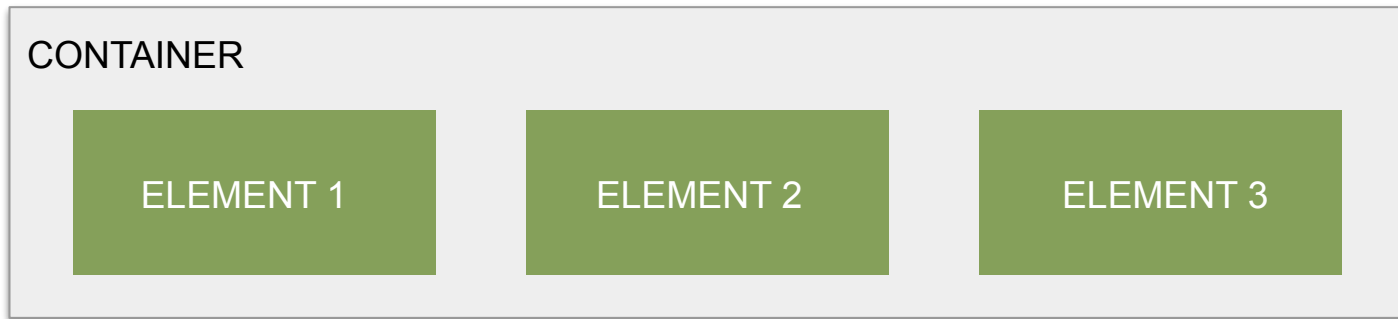
Autres propriétés

visibility	Visibilité
overflow	Comportement lors de dépassement du contenu
float	Flottant (droite, gauche)
z-index	ordre d'affichage
cursor	curseur de souris

Flexbox : boîtes flexibles

Le flexbox est modèle de disposition : alignement, direction, ordre et taille des éléments (boîtes)

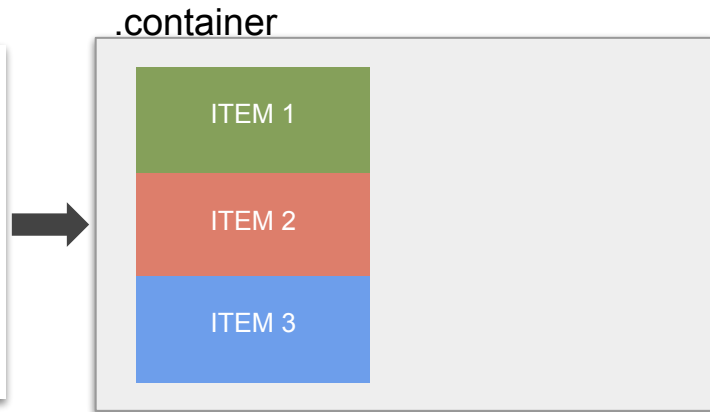
Permet de définir des conteneurs flexibles



Flex Container : display

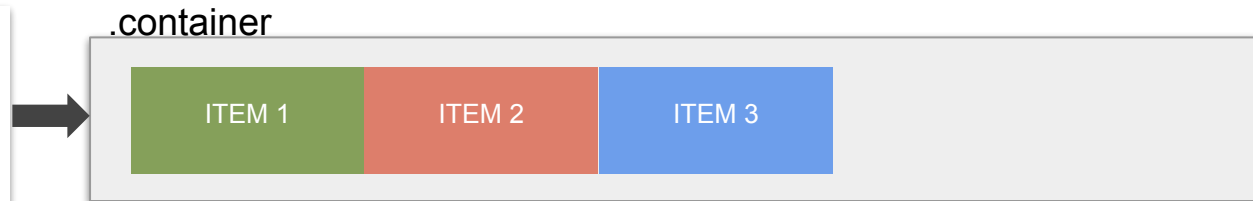
html

```
<div class="container">
  <div class="item">ITEM 1</div>
  <div class="item">ITEM 2</div>
  <div class="item">ITEM 3</div>
</div>
```



CSS

```
.container {
  display: flex;
}
```

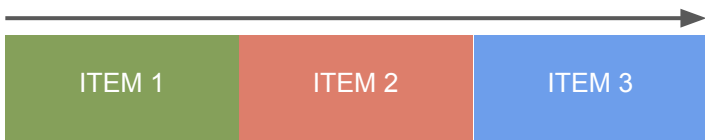


Flex Container : flex-direction

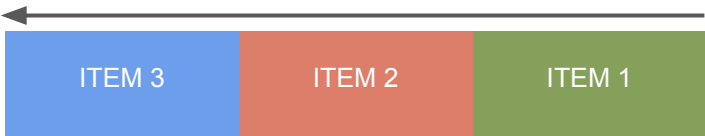
CSS

```
.container {  
  display: flex;  
  flex-direction: row | row-reverse | column | column-reverse;  
}
```

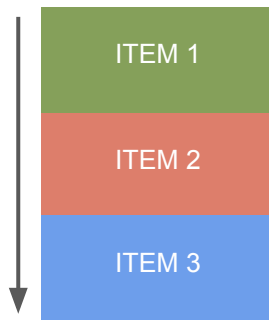
row



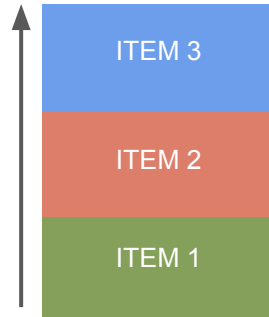
row-reverse



column



column-reverse

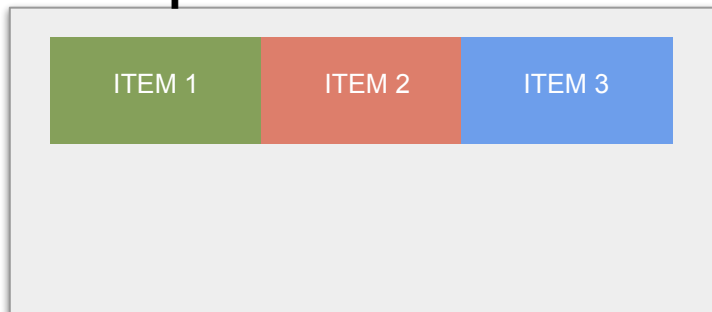


Flex Container : flex-wrap

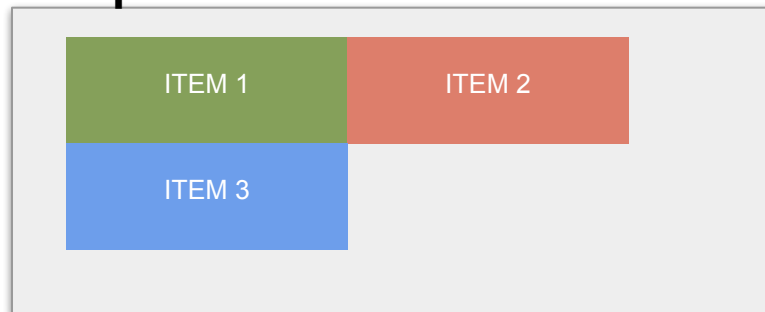
CSS

```
.container {  
  display: flex;  
  flex-wrap: nowrap | wrap | wrap-reverse;  
}
```

nowrap



wrap



Flex Container : justify-content

CSS

```
.container {  
  display: flex;  
  justify-content: flex-start | flex-end | center | space-between |  
  space-around | space-evenly;  
}
```

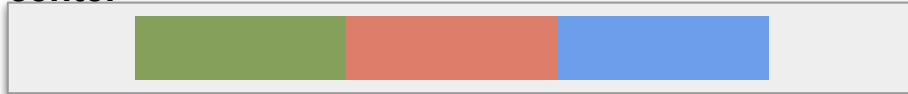
flex-start



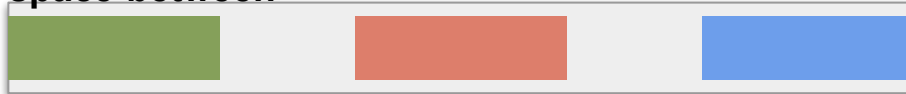
flex-end



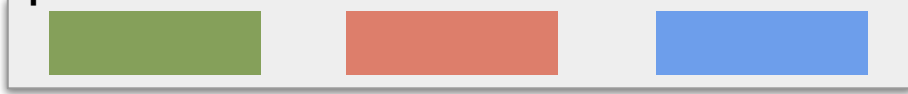
center



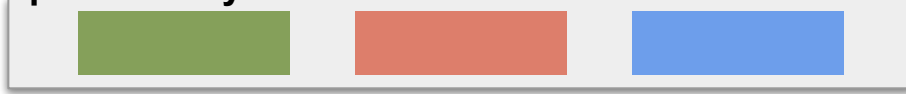
space-between



space-around



space-evenly

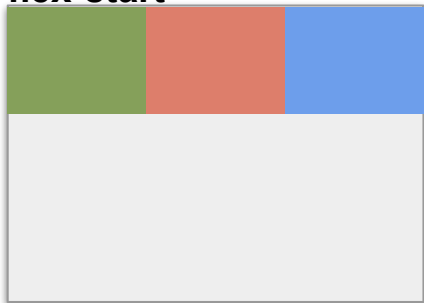


Flex Container : align-items

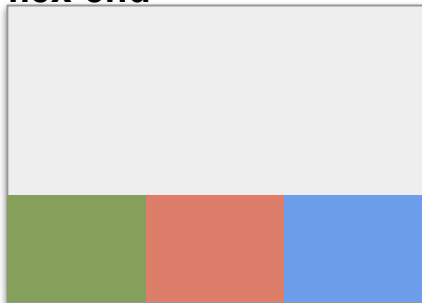
CSS

```
.container {  
  display: flex;  
  align-items: stretch | flex-start | flex-end | center;  
}
```

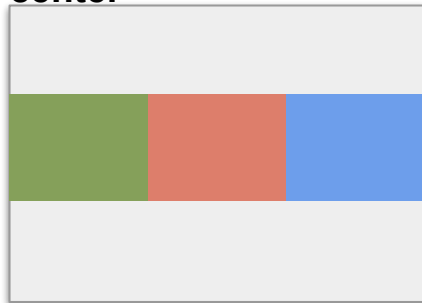
flex-start



flex-end



center



stretch

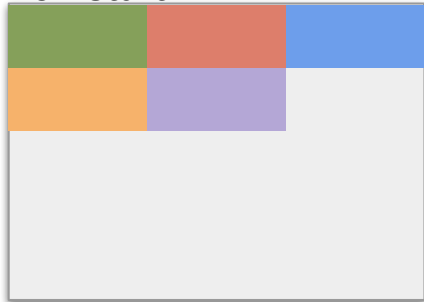


Flex Container : align-content

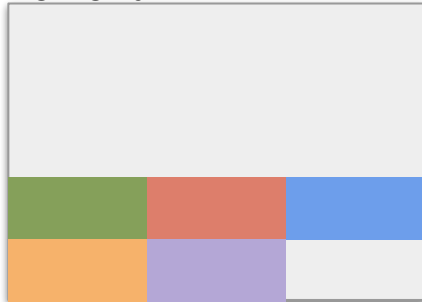
CSS

```
.container {  
  display: flex;  
  align-content: flex-start | flex-end | center | space-between |  
  space-around | space-evenly | stretch;  
}
```

flex-start



flex-end



center



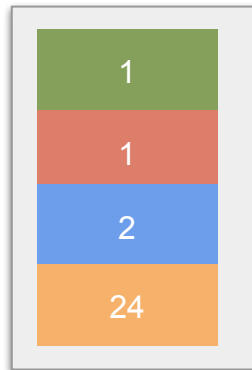
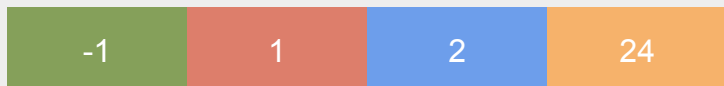
stretch



Flex items : order

CSS

```
.item {  
  order: <unEntier>;  
}
```



Flex items : flex-grow

CSS

```
.item {  
  flex-grow: <unNombre>;  
}
```

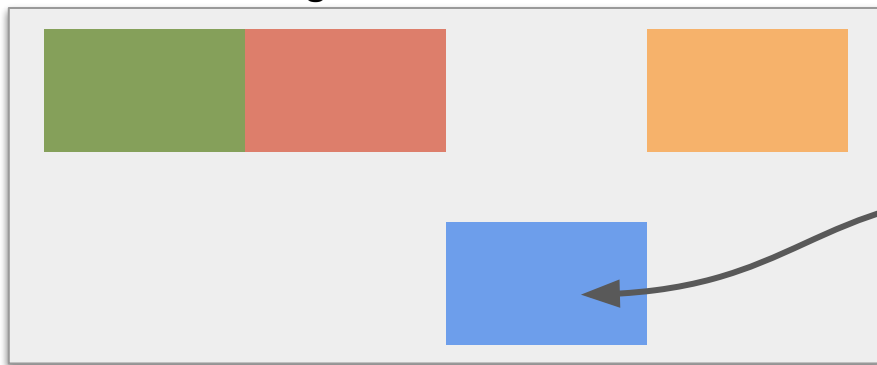


Flex items : align-self

CSS

```
.item {  
  align-self: auto | flex-start | flex-end | center | baseline | stretch;  
}
```

container → align-items : flex-start



align-self : flex-end

Responsive design

Responsive design = concept de conception Web dans lequel notre site va s'adapter à la taille de l'écran

Comment ?

Grâce aux **Media-queries**

<https://mediaqueri.es/>



Media-query

Requêtes média (media queries) : permettent de modifier l'apparence d'une page en fonction du type d'appareil et de ses caractéristiques

2 manières de procéder :

```
<link rel="stylesheet" href="style.css" />
```

```
<link rel="stylesheet" media="screen and (max-width: 800px)" href="style-mobile.css" />
```

```
@media screen and (max-width: 800px){  
    /* Propriétés CSS */  
}
```

Media-query

Adapter le design en fonction de plusieurs règles :

- Largeur, hauteur, orientation, type d'écran (screen, tv, print, ...)
- Mots clés comme only, and, not

@media + type d'écran + une ou plusieurs conditions

Le style CSS qui suit sera activé uniquement si les conditions sont remplies

```
@media screen and (max-width: 1280px)
{
    .texte {
        color: red;
    }
}
```

```
@media only screen and (max-width: 900px)
and (min-width: 600px) {
    .texte {
        color: red;
    }
}
```