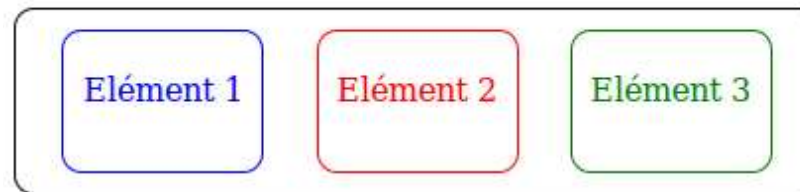


Avec Flexbox une page est vue comme un conteneur qui contient plusieurs éléments.

Dans une page Web on peut avoir plusieurs conteneurs contenant eux-mêmes d'autres conteneurs.

Conteneur



En HTML un conteneur sera une balise et chaque élément également (en général un `div`) :

```

<div id="conteneur">
  <div id="element1"></div>
  <div id="element2"></div>
  <div id="element3"></div>
</div>
  
```

G. Guillou, H. N.
Tran

Internet

Web

HTML et CSS

HTML

CSS

Structure

Flexbox

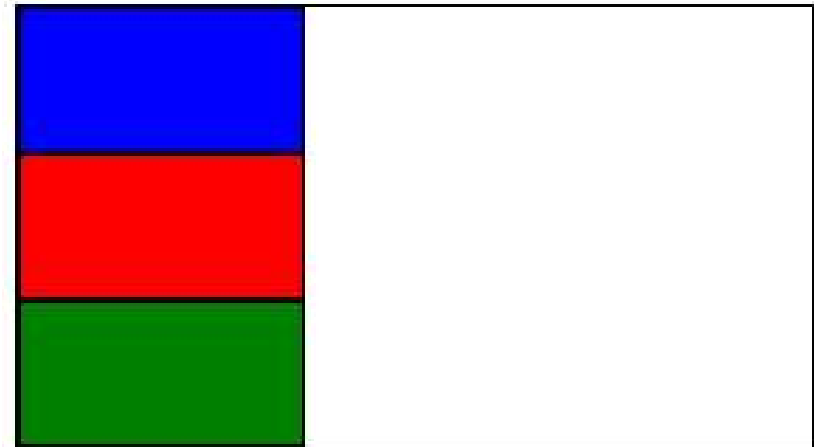
Audio et Vidéo

Media queries

JavaScript

Par défaut les éléments se mettent les uns au-dessus des autres :

```
#conteneur  
{  
  border : 1px black solid;  
}
```



G. Guillou, H. N.
Tran

Internet

Web

HTML et CSS

HTML

CSS

Structure

Flexbox

Audio et Vidéo

Media queries

JavaScript

Propriété `display : flex;`

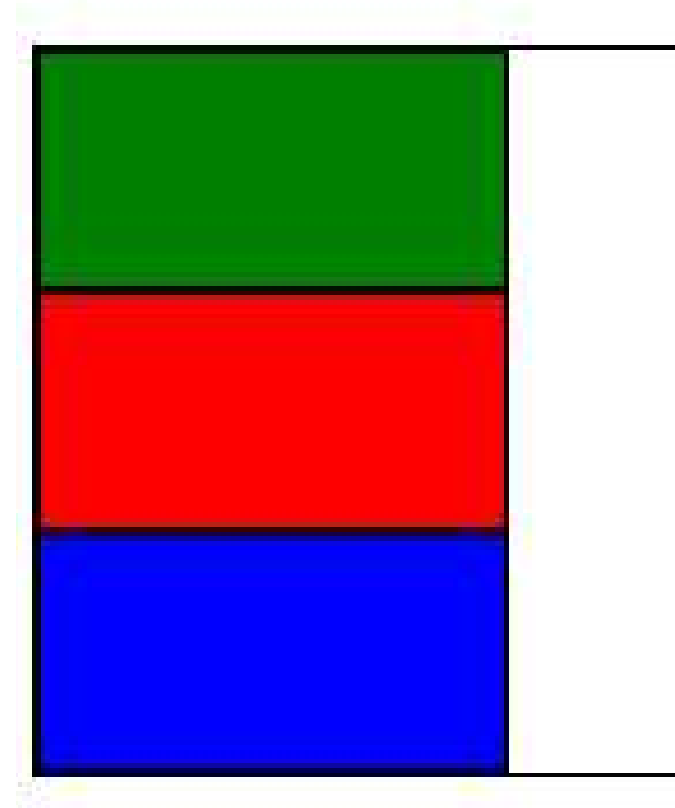
```
#conteneur  
{  
    border : 1px black solid;  
    display : flex;  
}
```

Les éléments se placent par défaut en ligne !



Propriété `flex-direction` qui peut prendre les valeurs :

- `row` en ligne (par défaut)
- `column` en colonne
- `row-reverse` en ligne inversés
- `column-reverse` en colonne renversés



`column-reverse`

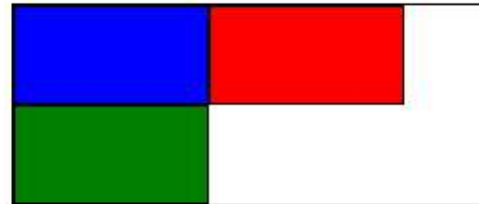
On peut changer l'ordre des blocs sans modifier le HTML !

La propriété `flex-wrap` peut prendre les valeurs :

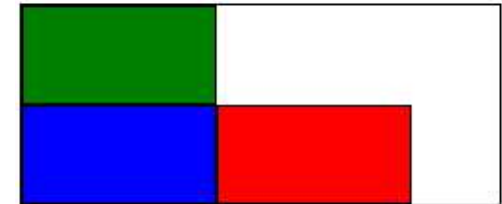
`no-wrap`



`wrap`



`wrap-reverse`



`no-wrap` a écrasé chaque bloc pour qu'ils puissent rentrer dans le conteneur.

Aligner sur l'axe principal

G. Guillou, H. N.
Tran

Internet

Web

HTML et CSS

HTML

CSS

Structure

Flexbox

Audio et Vidéo

Media queries

JavaScript

L'organisation des éléments avec Flexbox se déclare soit horizontalement (par défaut) soit verticalement.

Cette déclaration définit l'**axe principal**.

L'**axe secondaire** (cross axis) est alors l'autre axe perpendiculaire à l'axe principal. Il est donc :

- vertical si l'axe principal est horizontal
- horizontal si l'axe principal est vertical

Pour aligner sur l'axe principal, utiliser `justify-content` dont les valeurs possibles sont :

- `flex-start` : éléments alignés au début (par défaut)
- `flex-end` : éléments alignés à la fin
- `center` : éléments alignés au centre
- `space-between` : éléments répartis sur tout l'axe
- `space-around` : éléments répartis sur tout l'axe avec espace autour de chacun

Aligner sur l'axe principal : effets

G. Guillou, H. N. Tran

Internet

Web

HTML et CSS

HTML

CSS

Structure

Flexbox

Audio et Vidéo

Media queries

JavaScript



flex-start



flex-end



center



space-between



space-around

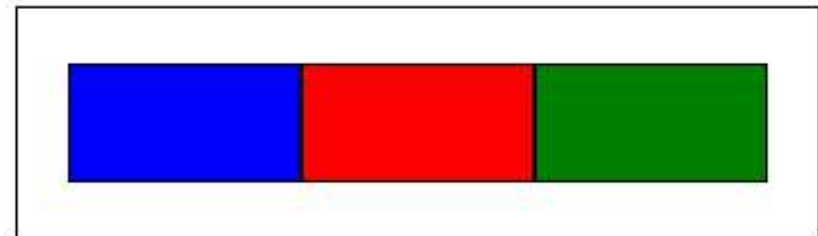
```
#conteneur
{
  border : 1px black solid;
  width : 350px;
  display : flex;
  flex-direction : row;
  justify-content : space-around;
}
```

On peut évidemment le faire aussi verticalement :
flex-direction : column;

On utilise pour cela la propriété `align-items` qui peut prendre les valeurs :

- `stretch` : les éléments s'étendent sur tout l'axe (par défaut)
- `flex-start` : alignés au début
- `flex-end` : alignés à la fin
- `center` : alignés au centre

```
#conteneur
{
  border : 1px black solid;
  width : 350px;
  height : 100px;
  display : flex;
  flex-direction : row;
  justify-content : center;
  align-items : center;
}
```



G. Guillou, H. N.
Tran

Internet

Web

HTML et CSS

HTML

CSS

Structure

Flexbox

Audio et Vidéo

Media queries

JavaScript

On peut obtenir le même effet que `space-around` sur l'axe principal et `center` sur l'axe principal avec simplement `margin : auto;` au niveau de chaque élément :

