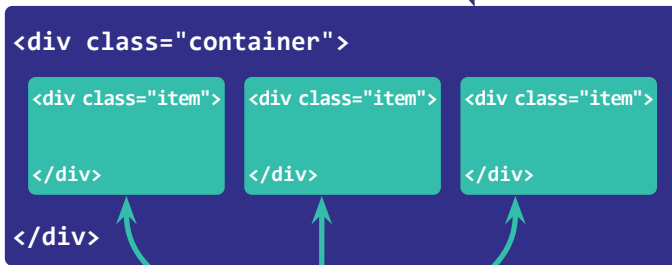


Généralités

- **Technique de mise en page** qui permet la création de dispositions complexes en CSS pour les documents HTML
- Permet l'alignement, le centrage et la distribution d'éléments selon un axe principal

parent



enfants

- **Définition d'un conteneur flexible**

```
.container{
  display: flex; /* ou inline-flex */
}
```

→ les enfants directs d'un conteneur flexible sont des éléments flexibles

- **Axes principal et secondaire**

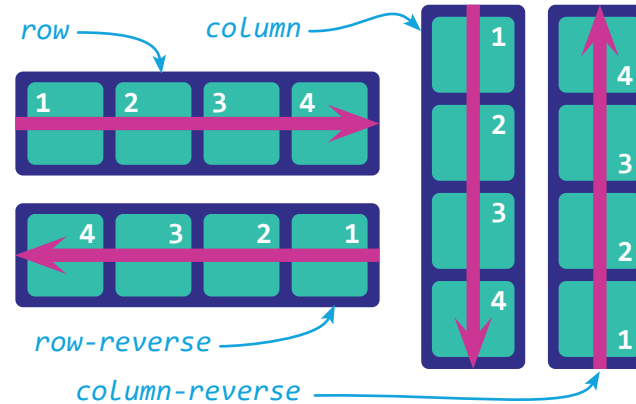
- Axe principal :
- Axe secondaire :
- Varient en fonction de la direction définie (flex-direction)



Quelques propriétés CSS flex

Propriétés s'appliquant au parent

- **flex-direction** : direction de l'axe principal
- ```
.container{
 flex-direction: row | row-reverse
 | column | column-reverse;
}
```



- **flex-wrap** : retour à la ligne

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

- **nowrap** : tous les items seront sur une ligne (par défaut)
- **wrap** : les items retourneront automatiquement à la ligne
- **wrap-reverse** : les items retourneront à la ligne du bas vers le haut
- **flex-flow**
  - Combinaison des propriétés **flex-direction** et **flex-wrap**

- **justify-content** : alignement sur l'axe principal
- **flex-start** : items au début de l'axe principal, valeur par défaut



- **flex-end** : items à la fin de l'axe principal



- **center** : items centrés sur l'axe principal



- **space-between** : items distribués uniformément sur l'axe principal, premier et dernier item au début et à la fin de la ligne



- **space-around** : items distribués uniformément sur l'axe principal avec une marge identique de chacun des côtés de chaque item



- **space-evenly** : items distribués uniformément sur l'axe principal avec un espace identique entre chaque item (et également avec les bords)



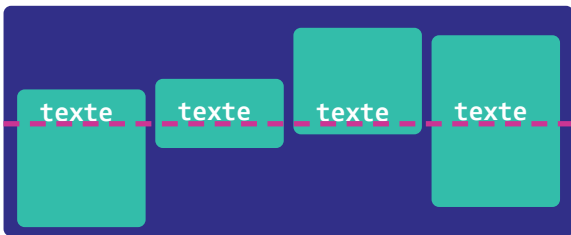
- **align-items** : alignement sur l'axe secondaire
  - **stretch** : items s'étirent pour remplir le conteneur sur l'axe secondaire (en respectant min-width et max-width), valeur par défaut



- **center** : items au centre de l'axe secondaire
- **flex-start** : items au début de l'axe secondaire



- **flex-end** : items à la fin de l'axe secondaire
- **baseline** : items alignés sur leur ligne de base



- **align-content** : alignement des lignes d'items dans le conteneur (**normal** (par défaut), **flex-start**, **flex-end**, **center**, **space-between**, **space-around**, **space-evenly**, **stretch**)
  - Uniquement avec **flex-wrap: wrap**;

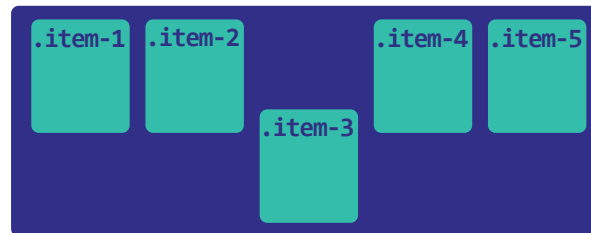
## Propriétés s'appliquant aux enfants

- **order** : ordre des items
  - `.item{ order: -n | 0 | n; }`
  - Items affichés dans l'ordre du DOM\* par défaut (**order: 0**)
  - Plus la valeur est grande plus l'affichage de l'item est décalé

- **align-self** : alignement particulier d'items sur l'axe secondaire (valeurs identiques à **align-items**)

Exemple :

```
.container{
 display: flex;
 align-items: flex-start;
}
.item-3{
 align-self: flex-end;
}
```



- **flex**
  - Combinaison des propriétés **flex-grow**, **flex-shrink** et **flex-basis**

\*DOM = Document Object Model, fournit une représentation structurée du document HTML et définit la façon dont cette structure peut être manipulée en terme de style et de contenu

- **flex-grow** : étirement d'un item dans l'espace disponible du conteneur (pas de valeur négative)
  - Par défaut les items occupent l'espace défini par leur dimension (**width / height**)



- Les items ont tous la même valeur → répartition équitable



- Un seul item a une valeur → tout l'espace disponible dans le conteneur lui est attribué



- Les items ont des valeurs différentes → répartition pondérée



- **flex-shrink** : contraction d'un item (pas de valeur négative, **1** par défaut)
- **flex-basis** : taille des items par défaut avant la distribution de l'espace disponible dans le conteneur

- **flex-basis: auto** : par défaut, les items occupent une largeur équivalente à leur contenu
- **flex-basis: 80%** : item occupe 80% du conteneur, les autres items se répartissent automatiquement

