

FeEXⁱB_eE
=====
III

1) Le conteneur (l'élément parent), se met en **display: flex;** ~~ainsi que ses enfants à déplacés.~~

2) Ensuite on lui rajoute une autre propriété.

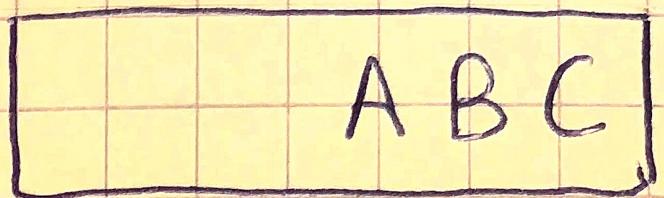
Par exemple pour lui demander d'aligner ses enfants on lui dira :

justify-content :

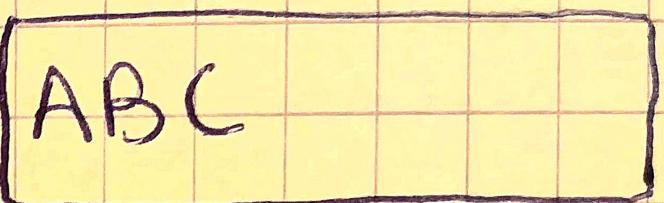
3) A côté de justify-content on va rajouter la valeur.

Celle-ci peut être

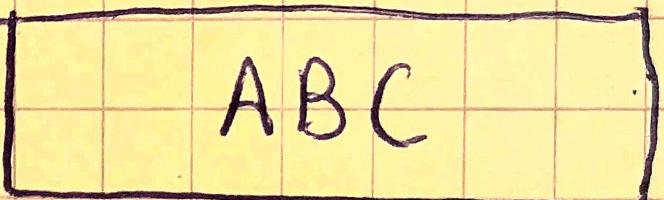
flex-end;



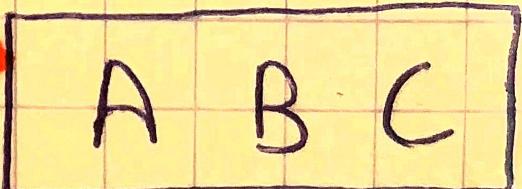
flex-start;



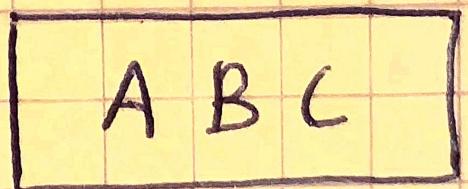
center;



space-between; →



space-around;



space-evenly;



Les 3 derniers se ressemblent.

4) A la place de la propriété justify-content, on peut utiliser (ou en plus) **flex-direction**:

5) on lui rajoute une valeur →

La valeur peut-être :

row-
(par défaut)

| | | |
|---|---|---|
| 1 | 2 | 3 |
|---|---|---|

column-

| |
|---|
| 1 |
| 2 |
| 3 |

| |
|---|
| 3 |
| 2 |
| 1 |

Column-reverse;

row-reverse;

| | | |
|---|---|---|
| 3 | 2 | 1 |
|---|---|---|

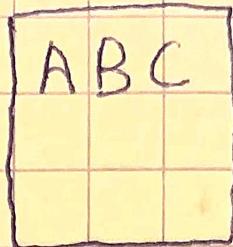
6) la propriété align-items:

permet de choisir l'alignement sur la hauteur.

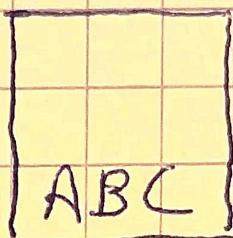
7) On lui rajoute une

valeur qui peut être :

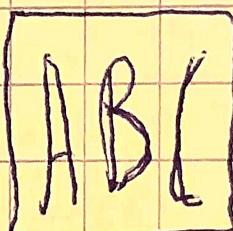
flex-start;



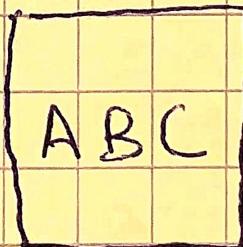
flex-end;



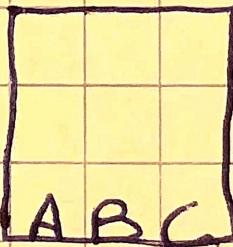
~~flex~~ stretch;



center;



baseline;

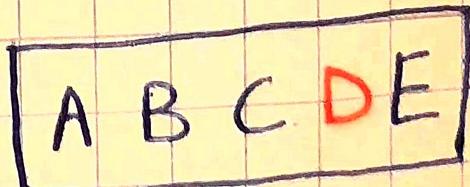


⚠ Quand le flex-direction est en column, le justify-content et le align-items s'inversent.

8) Si on veut changer l'ordre d'un seul élément, on utilise la propriété **order**:

9) On lui rajoute un chiffre sachant que 0 est sa position actuelle, 1 signifie qu'il va se positionner 1 place vers la droite et -1 le place vers la gauche.

ici D à une valeur de 0.



Si je veux mettre **D** à la place de **A**. J'écris dans le CSS :

```
.B {  
    order: -3;  
}
```

Si je veux mettre **D** à la place de **E**. J'écris :

```
order: 1;
```

10) Il y a une autre propriété que je peux utiliser sur des éléments individuels, c'est la

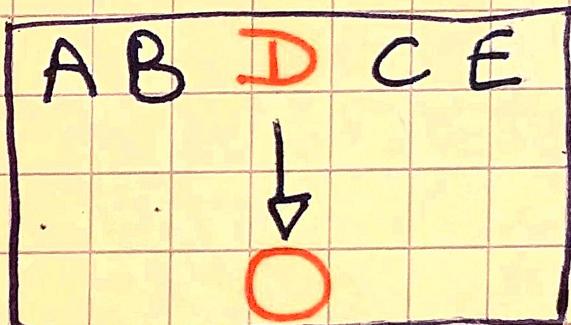
propriété align-self :
qui comporte les mêmes
valeurs que align-items :
sauf qu'elle cible l'UN élément

.conteneur {

display: flex;

align-items: flex-start;

}



.D {

align-self: flex-end;

}

11) Il y a la propriété **flex-wrap**: qui signifie que tous les éléments sont en flex.

12) **flex-wrap**: a plusieurs valeur comme celles-ci :
wrap; qui va à la ligne
wrap-reverse; qui va à la ligne dans le sens inverse
nowrap; ne va pas à la ligne.

13) La propriété flex-flow

flow: est le raccourcis de flex-direction et de flex-wrap. Elle accepte les valeurs des deux autres propriétés.

ex:

p {

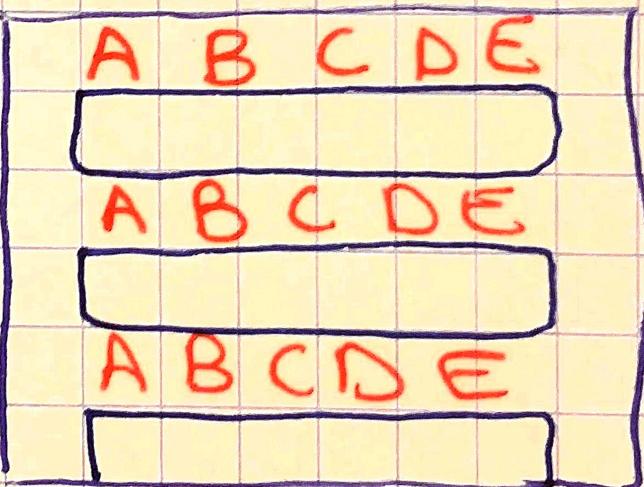
display: flex;

flex-flow: row wrap;

}

14) La propriété align-content : sert à définir l'espace entre des éléments alignés sur plusieurs lignes.

ex:



Si je veux mettre mes lignes d'éléments A,B,C,D,E dans les blocs bleus, j'utiliserais align-content :

align-content: peut avoir ces valeurs :

flex-start; les lignes vont vers le haut du conteneur.

flex-end; les lignes vont vers le bas du conteneur.

center- au centre.

space-between; un espace égale entre elles. Collés aux bords

space-around; un espace égale autour d'elles.

stretch; elles s'étirent sur tout le conteneur.