HTML

* Balises

Il existe des balises inline comme <a> ou <span>..

Des balises block comme <p> ou <div>..

* block :

une balise de type block sur votre page web crée automatiquement un retour à la ligne avant et après. Il suffit d'imaginer tout simplement un bloc. Votre page web sera en fait constituée d'une série de blocs les uns à la suite des autres. Mais vous verrez qu'en plus, il est possible de mettre un bloc à l'intérieur d'un autre, ce qui va augmenter considérablement nos possibilités pour créer le design de notre site !

* inline :

une balise de type inline se trouve obligatoirement à l'intérieur d'une balise block. Une balise inline ne crée pas de retour à la ligne, le texte qui se trouve à l'intérieur s'écrit donc à la suite du texte précédent, sur la même ligne (c'est pour cela que l'on parle de balise « en ligne »).

* Margin et padding
* margin:

margin-top: marge extérieure en haut ;

margin-bottom: marge extérieure en bas ;

margin-left: marge extérieure à gauche ;

margin-right: marge extérieure à droite.

* padding:

padding-top: marge intérieure en haut ;

padding-bottom: marge intérieure en bas ;

padding-left: marge intérieure à gauche ;

padding-right: marge intérieure à droite.

* Créer des reports vers une portion de page

<p>

Choisir son domaine <br />

<!--Id cuisine-->

<a href="#cuisine">La cuisine</a> <br />

</p>

<p>Lignes</p>

<p>Lignes</p>

<p>Lignes</p>

<p>Lignes</p>

<p>Lignes</p>

<p>Lignes</p>

<p>Lignes</p>

<!---- Ici on plpce l'id cuisine-->

<h2 id="cuisine">La cuisine</h2>

* Nouvel onglet et mail

<!-- target="\_blank ouvre un nouvel onglet"-->

<p>Visite du site <a href="https://openclassroom.com" title="vous allez aimer" target="\_blank">OpenClassroom</a></p>

<!-- envoi mail-->

<p>envoyer un mail <a href="mailto:vlaboure@hotmail.com">envoi de mail</a></p>

* Telecharger un fichier

<p>telecharger fichier <a href="fichier.txt">fichier dans le même repertoire que le code html</a></p>

* Afficher une image

MonImage est l’image grande et monImage-mini est une vignette de l’image

<p>

afficher une image<br />

<!--alt sert pour les aveugles et les robots-->

<a href="dossier\_ou\_est\_limage/monImage.gif"><img src="dossier\_ou\_est\_limage/monImage-mini.gif" alt="mon image"></a>

</p>

On voit à l’ouverture de la page l’image src donc la vignette

* Menu de navigation

<nav>

<ul>

<li><a href="#">Accueil</a></li>

<li><a href="#">Blog</a></li>

<li><a href="#">CV</a></li>

</ul>

</nav>

* La balise

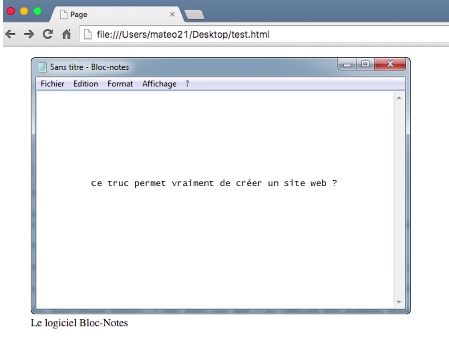
<figure>

<img…..>

<figurecaption></figurecaption>

</figure>

------------------- resultat



* <strong>

Utilisé pour indiquer l’importantce du texte

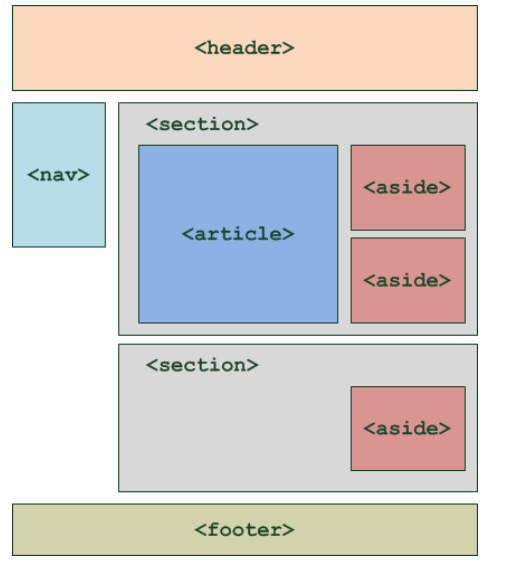
* <aside>

La balise aside est utilisée pour des info complémentaires au document

* <article>

Portion autonome de la page

* Récapitulatif :



* Les tableaux

<table>

<tr></tr> 🡪 Ligne

<td></td> 🡪 cellule

</table>

Exemple :

<table>

<tr>

<td>Carmen</td>

<td>33 ans</td>

<td>Espagne</td>

</tr>

<tr>

<td>Michelle</td>

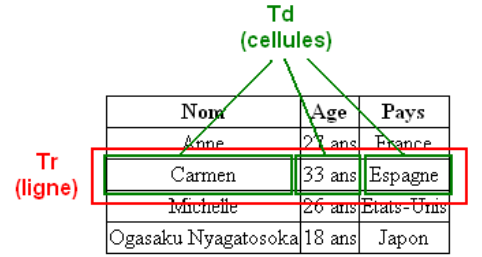
<td>26 ans</td>

<td>États-Unis</td>

</tr>

</table>

Résultat :



* Faire des bordures

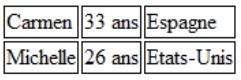
td /\* Toutes les cellules des tableaux... \*/

{

border: 1px solid black; /\* auront une bordure de 1px \*/

}

Résultat



* Bordure sans espace

table

{

border-collapse: collapse; /\* Les bordures du tableau seront collées (plus joli) \*/

}

td

{

border: 1px solid black;

}

* Pour l’en-tête on utlilise <th>

<table>

<tr>

<th>Nom</th>

<th>Âge</th>

<th>Pays</th>

</tr>

<tr>

<td>Carmen</td>

<td>33 ans</td>

<td>Espagne</td>

</tr>

<tr>

<td>Michelle</td>

<td>26 ans</td>

<td>États-Unis</td>

</tr>

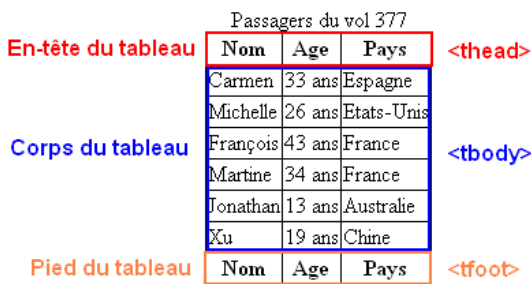
</table>

* Titre d’un tableau

<table>

<caption>Passagers du vol 377</caption>

* Tableau structuré



* Fustion ligne et colonne

<tr rowspan= « nbr de colonnes »>

<td colspan= « nbr de lignes »>

**CSS**

* Taille de texte

Utiliser l’unité en *em* permet d’avoir une taille relative à l’écran

* Polices

***Fontsquirrel.com 🡪 fontes pour sites***

On charge un fichier de polices, on dezipe on place les fichiers de police dans le un répertoire font

Dans le fichier téléchargé il y a un CSS qui contient le code à insérer dans non CSS

* *Float* CSS : un élément est flottant par rapport au suivant

Ex : texte qui entoure une image

*Clear* CSS : supprime le float

* Inserer une image de fond : background: url("soleil.png") fixed no-repeat top right;

fixed = image fixe

no-repeat = pas de repetition

top right = position

* Exemple CSS pour centrer un paragraphe entouré d’une bordure

p

{

width: 350px; /\* On a indiqué une largeur (obligatoire) \*/

margin: auto; /\* On peut donc demander à ce que le bloc soit centré avec auto \*/

border: 1px solid black;

text-align: justify;

padding: 12px;

margin-bottom: 20px;

}

* Bordures :

*border-left: 2px solid black;*

*border-radius: 10px;*

*border-radius: 10px 5px 10px 5px;*

radius : interprétation valeurs : en haut à gauche/en haut à droite/ en bas à droite/ en bas à gauche.

*box-shadow: 6px 6px 0px black;*

*text-shadow: 6px 6px 0px black;*

shadow : interprétation valeurs : le décalage horizontal de l'ombre/le décalage vertical de l'ombre/l'adoucissement du dégradé/la couleur de l'ombre.

* overflow

oververflow permet de recadrer dans les limites d’une section CSS un texte ou image qui dépasse

Les options :

* visible(par défaut) :

si le texte dépasse les limites de taille, il reste visible et sort volontairement du bloc.

* hidden:

si le texte dépasse les limites, il sera tout simplement coupé. On ne pourra pas voir tout le texte.

* scroll:

là encore, le texte sera coupé s'il dépasse les limites. Sauf que cette fois, le navigateur mettra en place des barres de défilement pour qu'on puisse lire l'ensemble du texte. C'est un peu comme un cadre à l'intérieur de la page.

* auto:

c'est le mode « pilote automatique ». En gros, c'est le navigateur qui décide de mettre ou non des barres de défilement (il n'en mettra que si c'est nécessaire). C'est la valeur que je conseille d'utiliser le plus souvent.

* word-wrap :break-word

Coupe et met à la ligne tout ce qui dépasse de la taille du bloc

* Réactions utilisateur :

*hover :* survol – lien

*active*: click – lien

*focus :* selection – util pour les zones de saisie

*visited* : site visité – lien

* Visited activ focus

a :visited 🡪 permet de changer le style d’un lien visité

a :activ 🡪 permet de changer le style d’un lien activé

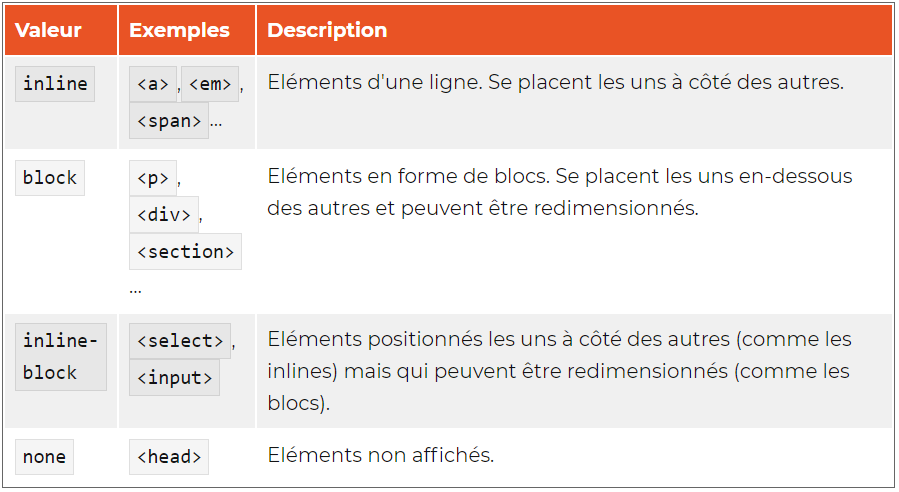
a :focus 🡪 permet de changer le focus d’un lien visité

* Display

Display : type d’affichage

Permet de modifier la façon d’afficher

Les valeurs possibles :



Quand on emploie diplay : none, pour faire réapparaitre la section dynamiquement on utilise JS.

* position

position : absolute 🡪 par rapport aux bords de l’écran (haut,bas,gauche,droite)

relative🡪par rapport à sa position dans le HTML (haut, gauche)

fixed🡪 comme absolu mais le texte reste fixe si on scrolle sur la page

* Flex

Soit un conteneur avec 3 blocs <div>

<div id="conteneur">

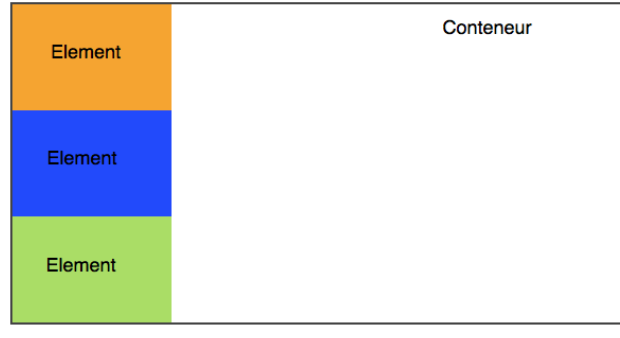
<div class="element">Elément 1</div>

<div class="element">Elément 2</div>

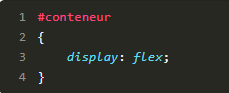
<div class="element">Elément 3</div>

</div>

Les blocs par defaut vont se placer :



Si on utilise la propriété flex dans le CSS



Alors les blocs se placeront :



Les paramètres :

* Row

Placement en ligne de gauche à droite des blocs (inline)

* row-reverse

En ligne de droite à gauche

* column

Placement en colonne haut bas

* column-reverse

Placement en colonne bas haut

* flex-wrap

permet un retour à la ligne en fin de container

nowrap : pas de retour à la ligne (par défaut)

wrap : les éléments vont à la ligne lorsqu'il n'y a plus la place

wrap-reverse : les éléments vont à la ligne lorsqu'il n'y a plus la place en sens inverse

* justify-content (aligner sur l’axe principal row-column) :

Permet de gérer l’allignement des blocs dans le container

Les options :

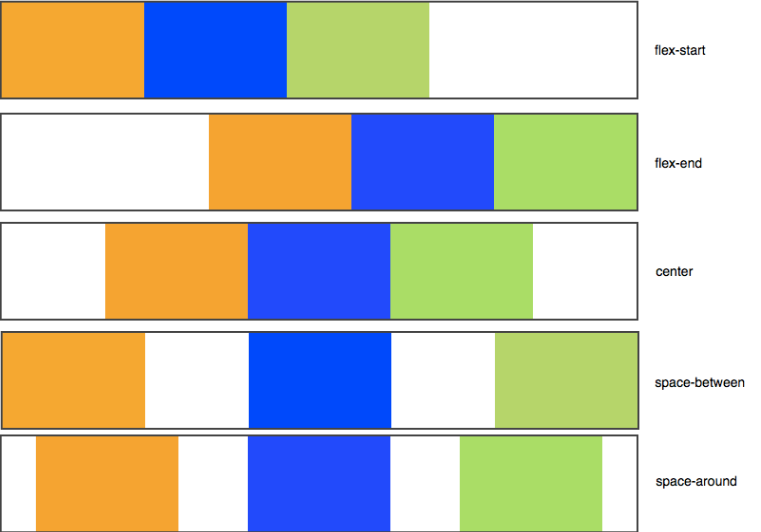
flex-start : alignés au début (par défaut)

flex-end : alignés à la fin

center : alignés au centre

space-between : les éléments sont étirés sur tout l'axe (il y a de l'espace entre eux)

space-around : idem, les éléments sont étirés sur tout l'axe, mais ils laissent aussi de l'espace sur les extrémités



* align-items (alignement sur l’axe secondaire)

si les blocs sont positionnés en row l’axe secondaire est en column

s’ils sont en column l’axe secondaire est en row

exemple :

#conteneur

{

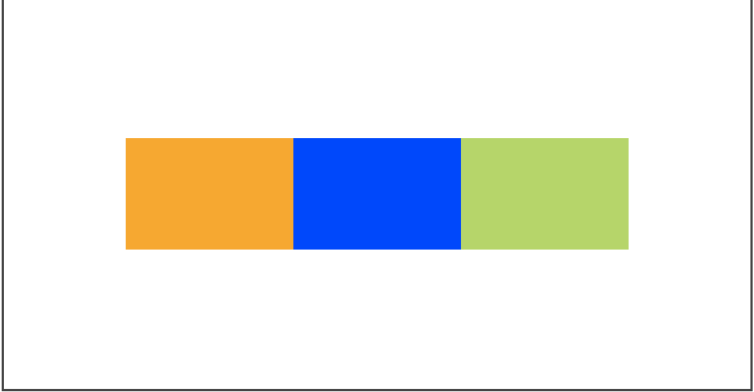
display: flex;

justify-content: center;

align-items: center;

}

Résultat :



Alignement d’un seul élément en utilisant la classe et le nr d’élément

#conteneur

{

display: flex;

flex-direction: row;

justify-content: center;

align-items: center;

}

.element:nth-child(2) /\* On prend le deuxième bloc élément \*/

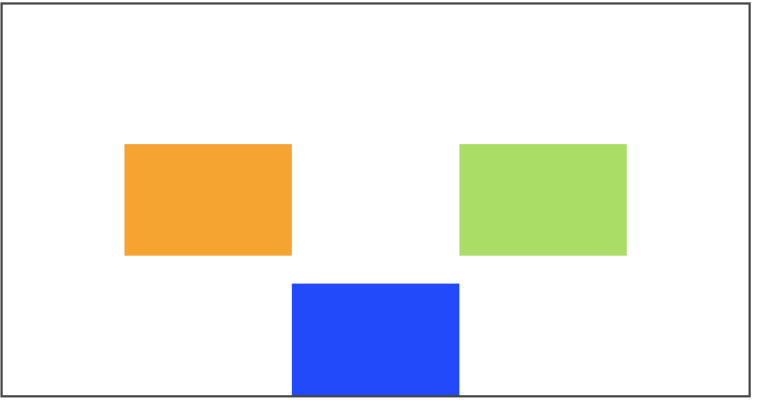
{

background-color: blue;

align-self: flex-end; /\* Seul ce bloc sera aligné à la fin \*/

}

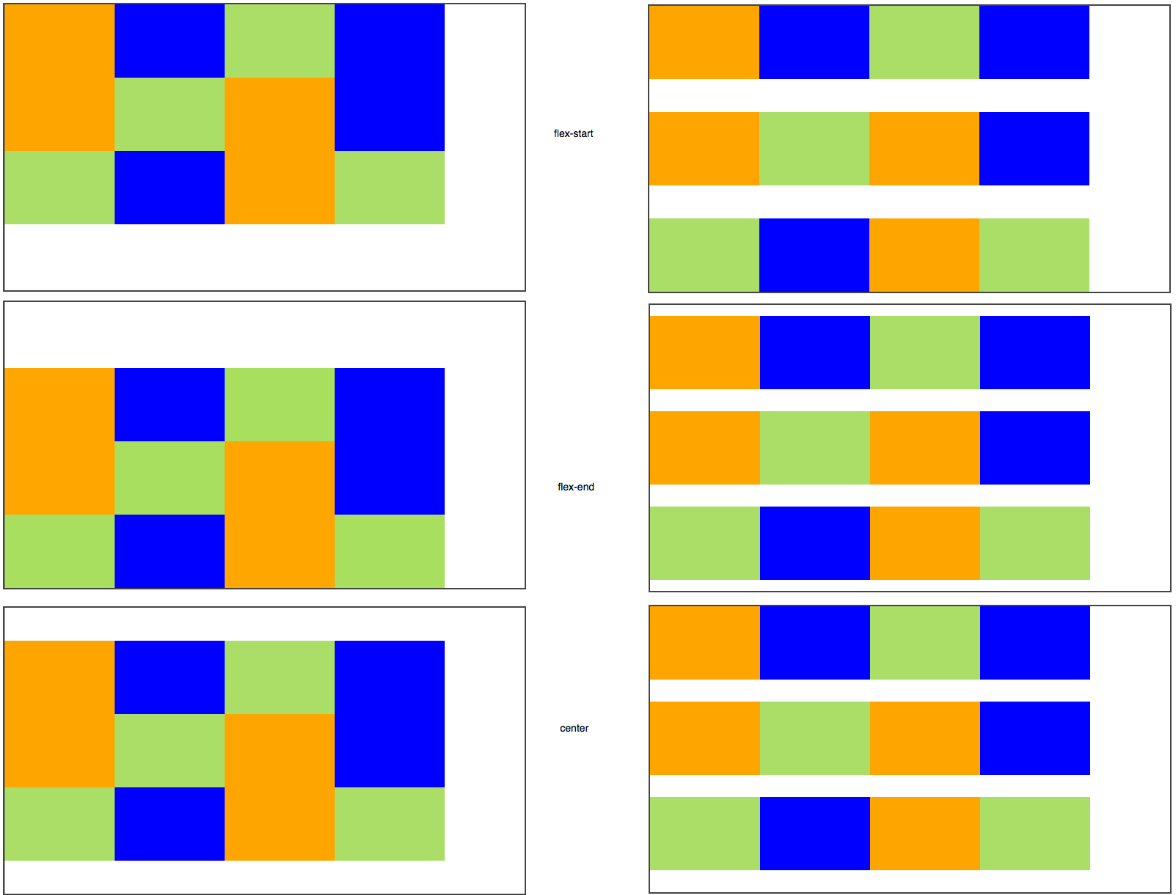
Résultat



Exemple de positionnement de blocs



Differents positionnements :



En utilisant le positionnement avec nth-child on peut modifier la position des blocs individuellement

.element:nth-child(1)

{

order: 3;

}

.element:nth-child(2)

{

order: 1;

}

.element:nth-child(3)

{

order: 2;

}

Ou modifier la taille :

.element:nth-child(2)

{

flex: 1;

}

