

# LE HTML ET LE CSS

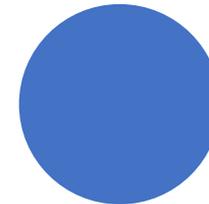


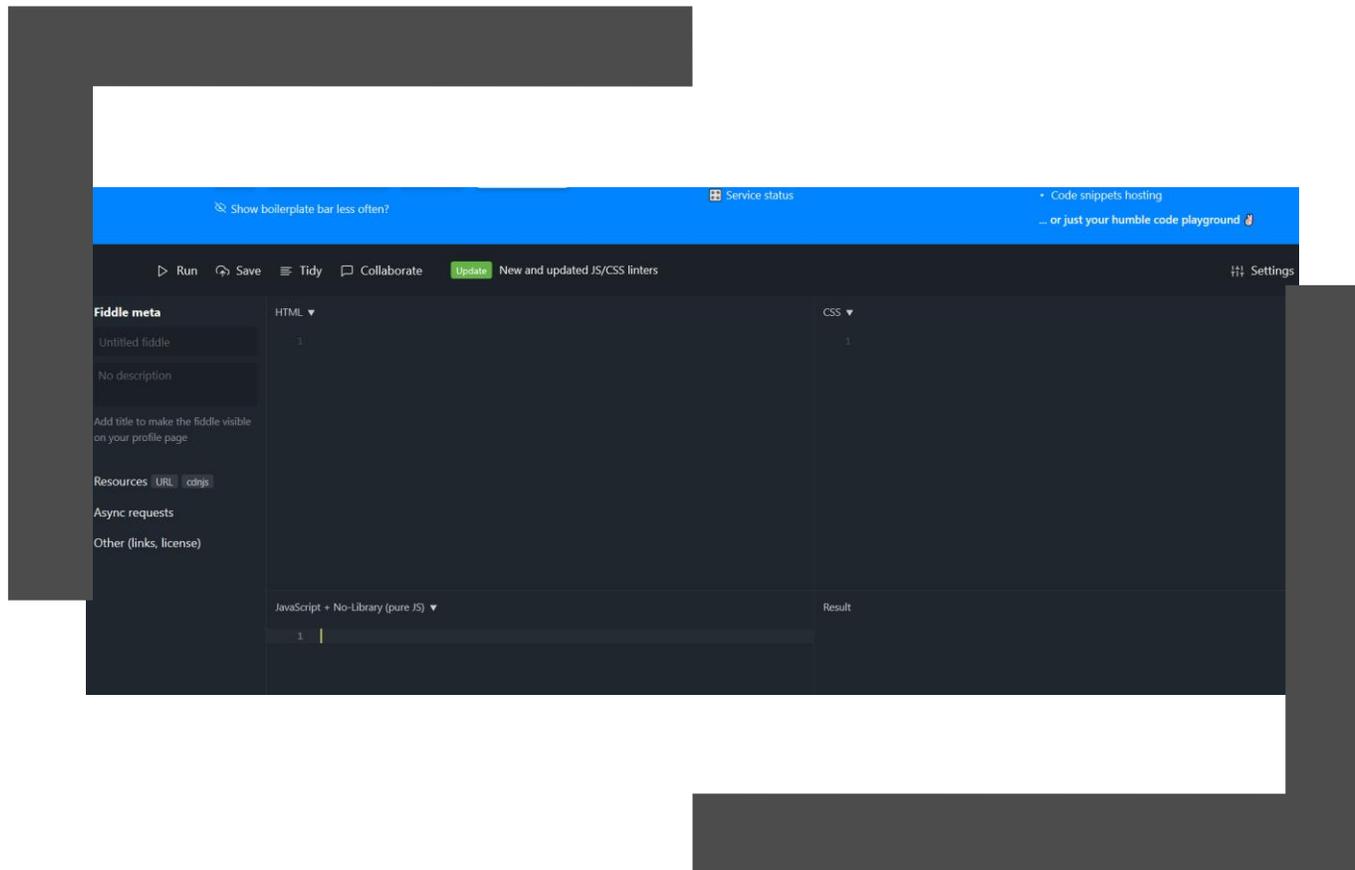
Nous allons nous intéresser à un acteur fondamental du développement web, le couple HTML+CSS (Hyper Text Markup Language et Cascading Style Sheets). Dans un premier temps, nous allons exclusivement nous intéresser au HTML.

**question1:** Qu'est-ce que le HTML? Quelle est la différence entre le langage Python et le langage HTML?

---

# HTML et CSS

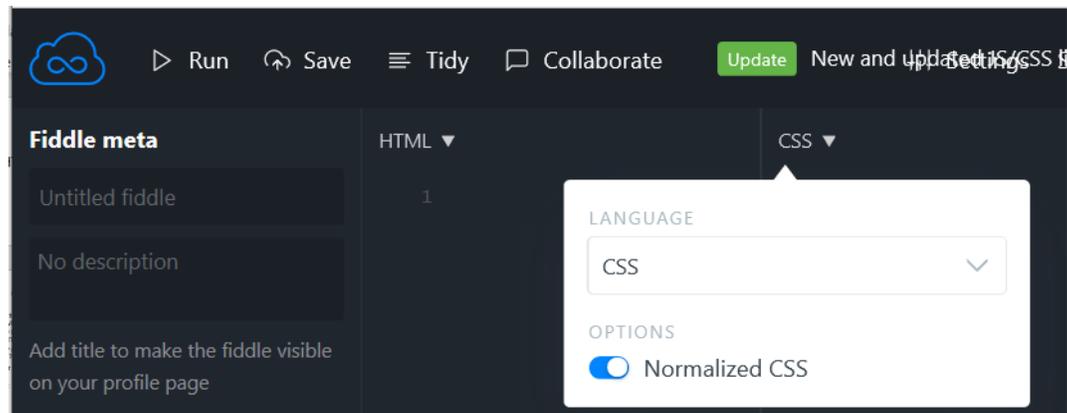




Après avoir lancé votre navigateur web, tapez, dans la barre d'adresse :

<http://jsfiddle.net/>

Vous devriez voir apparaître la fenêtre présentée à gauche.



Dans l'onglet CSS, utiliser l'onglet « normalized CSS »

```
1 <h1>
2   Page d'exemples
3 </h1>
4 <h2>
5   Première partie
6 </h2>
7 <p>
8   écrire n'importe quoi ici, en écrire plusieurs lig
9   nes, vous pouvez faire un copier coller d'autres sou
10  rces, encore et encore histoire de faire du volume
11 </p>
12 <h2>
13   Seconde partie
14 </h2>
15 <p>
16   écrire encore une fois n'importe quoi ici, peut êt
17   re un copier coller de la ligne 8.
18 </p>
```

**Question 1:** Écrivez le code HTML à gauche.

**Question 2:** Que-est ce qui s'affiche?

Les mots entre chevrons comme `<h1>` , `<p>` sont appelés des balises

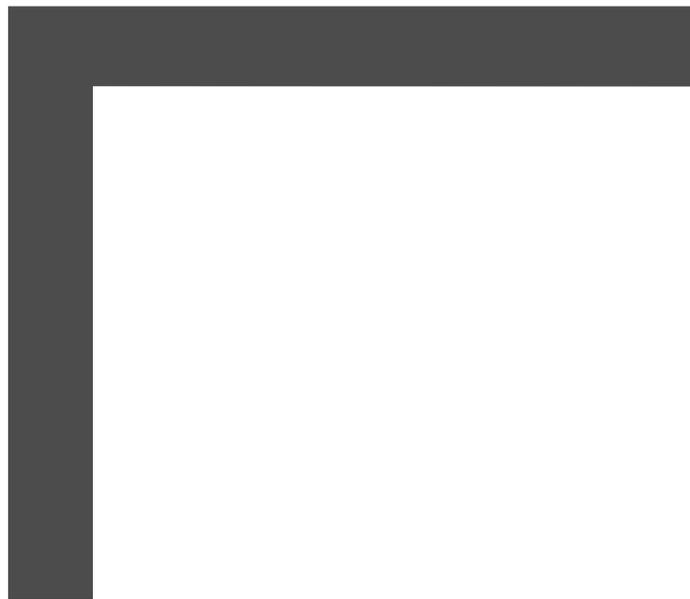
**Question 3 :** Quelle est la différence entre les balises `<h1>`, `<h2>` et `<p>`?

**Question 4:** A quoi sert `</h1>`?

```
1 <h1>
2   Page d'exemples
3 </h1>
4 <h2>
5   Première partie
6 </h2>
7 <p>
8   écrire n'importe quoi ici, en écrire plusieurs lig
9   nes, vous pouvez faire un copier coller d'autres sou
10  rces, encore et encore histoire de faire du volume
11 </p>
12 <h2>
13   Seconde partie
14 </h2>
15 <p>
16   écrire encore une fois n'importe quoi ici, peut être
17   un copier coller de la ligne 8.
18 </p>
```

**Question 5 :** Que va-t-il se passer si on tape?

**Question 6 :** Expliquer quelles sont les règles d'enchaînement des balises?



**Question 7** : A quoi peut servir les balises `<br>`?

Un peu de CSS

Taper le code CSS suivant :

```
1  h1{
2    color:green;
3    background-color:yellow;
4    font-style:italic;
5    text-align:center;
6  }
7
8  p{
9    font-family: Copperplate, Papyrus, fantasy;
10 }
```

**Question 8:** Lancer le run, à l'aide

de ce qui s'affiche, déterminer à quoi sert le CSS?

**Question 9:** Déterminer ce à quoi servent les fonctions : color; background; font-style; text-align

**Question 10 :** Combiner HTML et CSS pour afficher la mise en page ci-dessous.

*Page d'exemples*

Première partie

écrire n'importe quoi ici, en écrire plusieurs lignes, vous pouvez faire un copier coller d'autres sources, encore et encore histoire de faire du volume

Seconde partie

écrire encore une fois n'importe quoi ici, peut être un copier coller de la ligne 8.

HTML ▾

```
1 <h1>
2   Page d'exemples
3 </h1>
4 <h2 id="first">
5   Première partie
6 </h2>
```

CSS ▾

```
#first{
  text-transform: uppercase;
  font-size: 40px;
}
```

**Question 11:** A quoi peut servir la fonction id? quel est son intérêt lors de la création d'un site WEB?

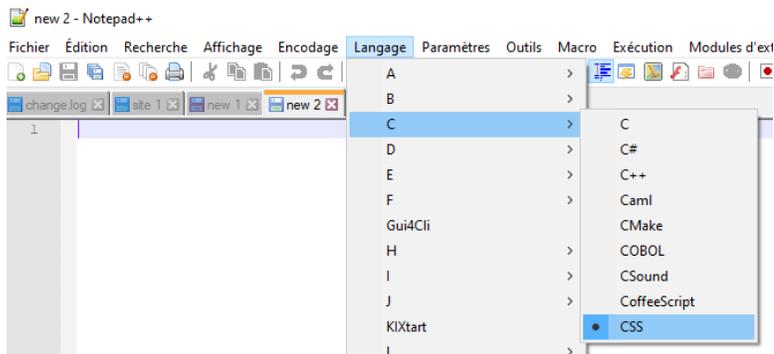
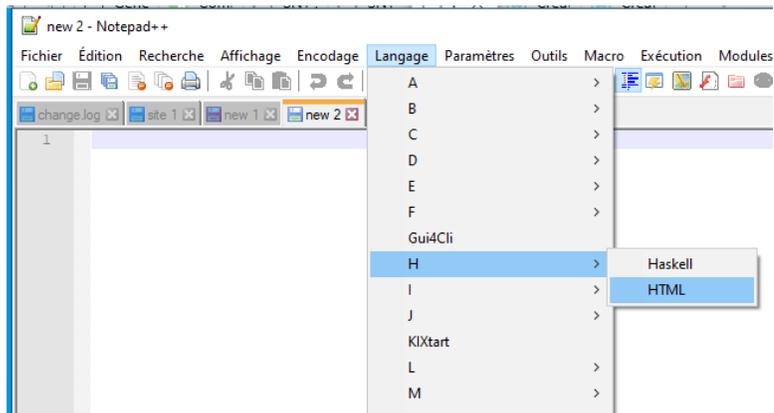
*Page d'exemples*

**PREMIÈRE PARTIE**

écrire n'importe quoi ici, en écrire plusieurs lignes, vous pouvez faire un copier coller d'autres sources, encore et encore histoire de faire du volume

**Seconde partie**

écrire encore une fois n'importe quoi ici, peut être un copier coller de la ligne 8.



---

JSFIDDLE est un très bel outil, mais il ne peut pas être utilisé pour la réalisation d'un vrai site internet.

Plutôt que de commencer un projet à 0 nous allons utiliser ce qu'on appelle un template constitué de 2 fichiers : un fichier qui contiendra du HTML (**index.html**) et un fichier qui contiendra du CSS (**style.css**).

Pour cela on peut utiliser le notepad++ qui est présent dans la suite logicielle de votre ordinateur ou **sublime text** que vous avez pu télécharger ici : <https://www.sublimetext.com/download> et que vous deviez installer sur votre ordinateur avant le cours d'aujourd'hui.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width">
6     <title>exemple intro</title>
7     <link href="style.css" rel="stylesheet" type="text/css" />
8   </head>
9   <body>
10    <h1>          Page d'exemples          </h1>
11    <h2 id="first"> Première partie </h2>
12    <p>
13      écrire n'importe quoi ici, en écrire plusieurs lignes,
14      vous pouvez faire un copier coller d'autres sources,
15      encore et encore histoire de faire du volume
16    </p>
17    <h2> Seconde partie    </h2>
18    <p>
19      écrire encore une fois n'importe quoi ici,
20      peut être un copier coller de la ligne 8.
21    </p>
22  </body>
23 </html>
```

Voici l'exemple de départ si on veut qu'il soit exploitable par le navigateur sans qu'on passe par fiddle.

Entre les balises <body> c'est un copier/coller de ce qu'on avait tout à l'heure.

Mais il y a toute une zone qui n'existait pas avant : <head> qui renseigne le navigateur sur le fonctionnement de la page. Notamment où trouver les informations css (dans un fichier appelé style.css, situé dans le même répertoire qu'index.html)

Le tout est enveloppé par une balise <html>

```

1 h1{
2   color:green;
3   background-color:yellow;
4   font-style:italic;
5   text-align:center;
6 }
7
8 p{
9   font-family: Copperplate, Papyrus, fantasy;
10 }
11
12 h2{
13   color:purple;
14   background-color:grey;
15   font-style:bold;
16   text-align:right;
17 }
18
19 #first{
20   text-transform: uppercase;
21   font-size: 40px;
22 }

```

Ça c'est le fichier style.css

Rien à signaler de ce côté : c'est un copier/coller de ce que nous avons sur fiddle.

Nom	Modifié le	Type	Taille
 index.html	29/12/2021 09:57	Chrome HTML Do...	1 Ko
 style.css	29/12/2021 08:56	Document de feui...	0 Ko

Pour pouvoir observer le rendu de notre mini projet on peut faire un double click sur index.html

Vu l'icône associée au fichier, ça va lancer celui-ci avec Chrome, chez vous ça va peut être utiliser un autre navigateur



## Retour sur la tête du fichier html

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width">
6     <title>exemple intro</title>
7     <link href="style.css" rel="stylesheet" type="text/css" />
8   </head>
```

Vous remarquerez que les balises <meta> ou <link> sont dites singletons ou à fermeture automatique. Elles sont comme <br/>, pas besoin de les fermer.

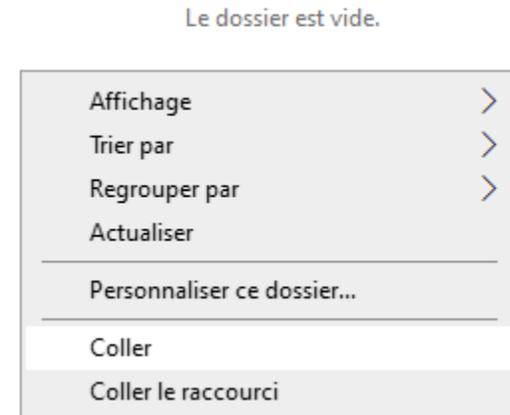
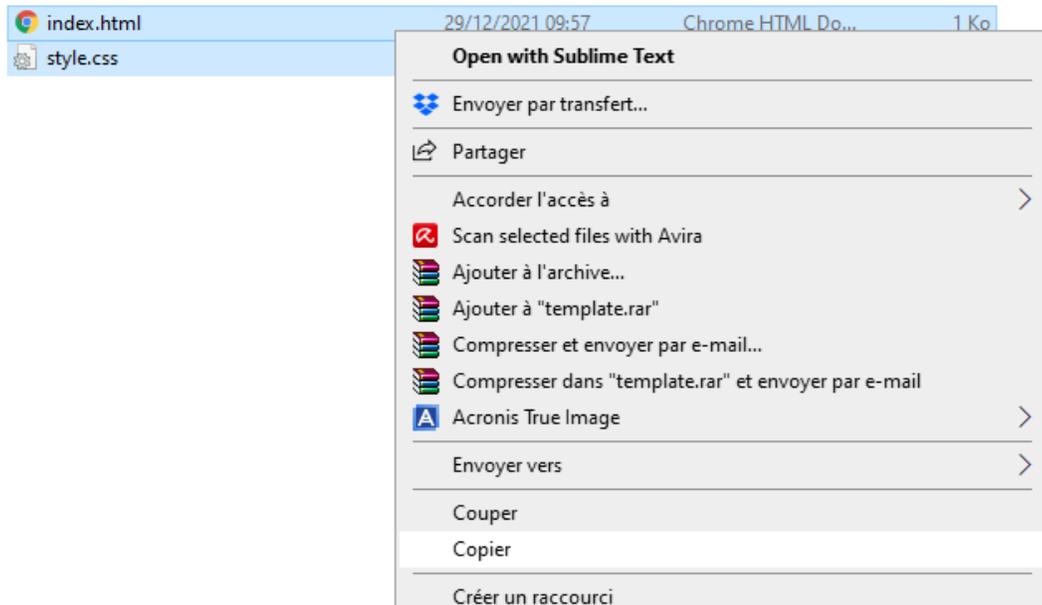
charset=« utf-8 » fait référence au jeu de caractère que l'on utilise pour l'affichage de la page. Sans cette information, l'ordinateur ne sera que faire des accents typiques de la langue française et affichera des symboles bizarre à la place.

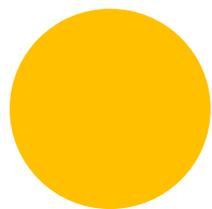
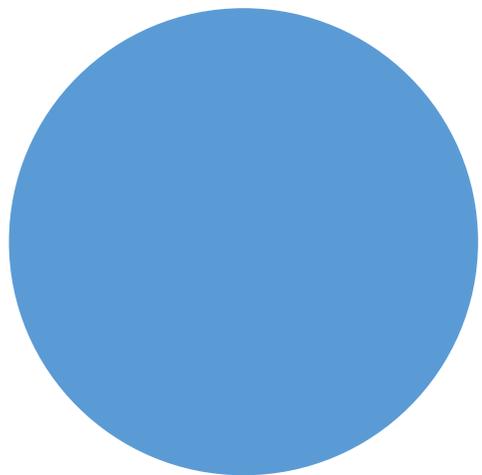
<title> permet de choisir ce qu'affiche l'onglet de la page dans le navigateur : exemple intro

Bonus : viewport correspond à l'ajustement automatique de la fenêtre d'affichage aux dimensions de l'écran.

[https://www.w3schools.com/css/css\\_rwd\\_viewport.asp](https://www.w3schools.com/css/css_rwd_viewport.asp)

Pour la suite il vous faudra créer un nouveau répertoire pour chaque nouveau projet et y copier les deux fichiers contenus dans le répertoire template (on peut les sélectionner puis d'un click droit choisir l'option « copier », puis aller dans le répertoire du nouveau projet et d'un click droit choisir l'option « coller »)





D'AUTRES BALISES



## La balise <a>

L'acronyme HTML correspond à **Hyper Text** Markup Language.

La balise <a> correspond à cette caractéristique la plus fondamentale du HTML : la possibilité de tisser des liens entre les différentes pages. On clique sur un mot, une photo, un bouton et nous voilà transporté à un autre endroit.

Lien inter-page : on écrit <a href="adresse de la page">

Exemple : `<a href="https://www.trivago.fr">clique ici si tu es comme moi</a>`

Lien intra : on écrit <a href="#identité de la balise où l'on veut aller">

Exemple : `<a href="#rendu">rendus</a>`

Permettra d'aller à la partie de la page dont le code html sera d'identité "rendu" par exemple:

```
<div id="rendu" class="bloc">
```

## La balise <a>

Le but de l'activité va être de créer votre page de fan. Ça peut être pour un groupe de musique, un cinéaste, un homme politique, un sportif, à vous de voir.

Vous allez avoir besoin de beaucoup de liens :

- des liens internes qui vous permettront à terme de vous servir du menu pour accéder aux différentes parties de votre longue page.
- Des liens externes qui permettront à l'utilisateur d'aller vers d'autres pages dédiées à l'objet de votre fanatisme, ou à des sujets connexes.

Commencez par créer un nouveau répertoire pour votre projet, vous y copierez le contenu du dossier « template » puis dans le fichier index.html vous allez créer plusieurs sections <div> :

<div id=« menu"> qui contiendra le menu de votre page </div>

<div id="identite"> qui contient les informations de base sur l'artiste, taille, poids, couleurs des yeux, genre, ... </div> ,

<div id=« faitdarmes"> qui va lister les choses pour lesquelles cette personne est connue (album, match, film,...) </div>

<div id="galerie"> galerie photo </div>

...

<div id=« liensexternes">qui donnera une liste de lien vers d'autres pages parlant du même sujet </div>

Puis dans la partie menu, vous allez créer des liens internes pour permettre à l'utilisateur d'accéder rapidement à aux différentes parties de la page (que vous venez de définir)

Puis dans la partie lien externes vous allez mettre une liste de liens vers d'autres sites.

## <img> La balise image

Une page web sans image c'est un peu morne, donc on va changer ça.

La syntaxe est ``

Concernant l'adresse, soit l'image est sur votre site, voire même dans le répertoire contenant votre projet et dans ce cas vous pouvez vous contenter d'écrire le nom de l'image.

Comme vous pouvez le voir l'image est bien trop grande pour ma page, heureusement il est possible d'adapter sa taille :

```

```

C'est peut être un peu radical, mais surtout les proportions ont été modifiées donc on préférera :

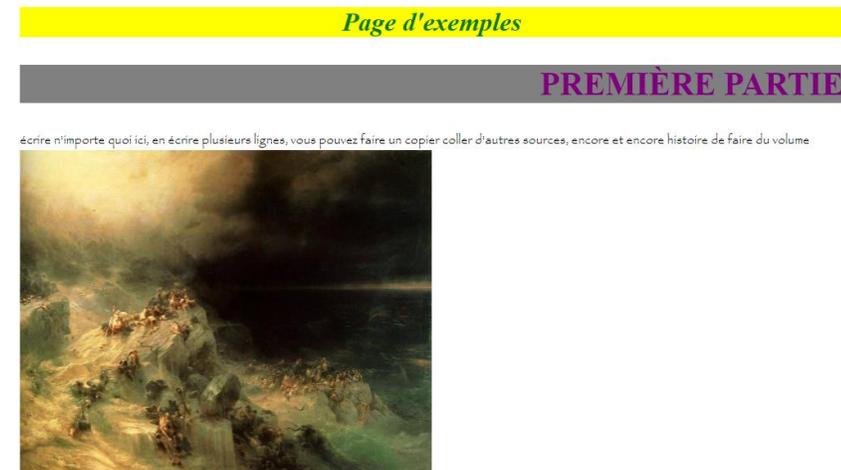
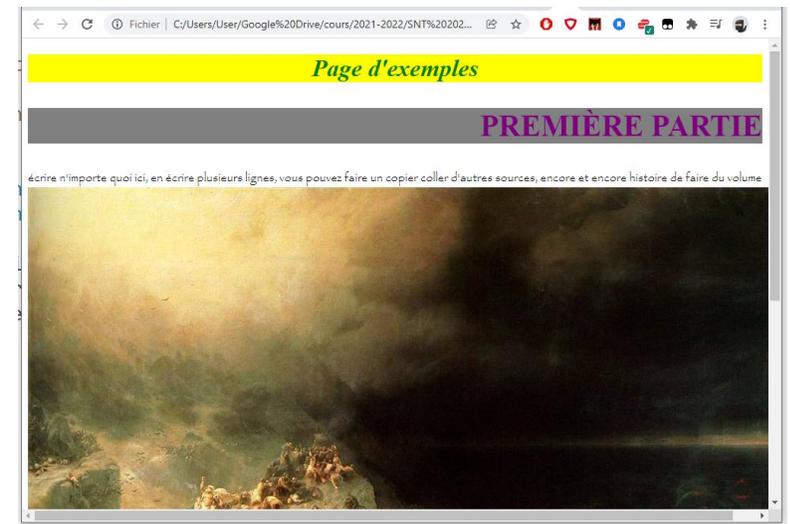
```

```

Qui va se caller sur la largeur de la page et afficher l'image de telle sorte que sa largeur soit 50% de celle de la page. C'est comme si on avait écrit `height="auto"`

Pour en savoir plus : <https://imagekit.io/blog/how-to-resize-image-in-html/>

Et en français : <https://developer.mozilla.org/fr/docs/Web/HTML/Element/Img>



## Les balises form, input et button

Les formulaires sont des éléments importants des sites internet, ils permettent à l'utilisateur de transmettre des informations. Un formulaire devra être délimité par une balise form (même si ce n'est pas une obligation) :

```
<form>
```

```
.....  
</form>
```

Il existe différentes balises permettant de construire un formulaire, notamment la balise input. Cette balise possède un attribut type qui lui permet de jouer des rôles très différents.

La balise button nous sera aussi d'une grande utilité.

Pour en savoir plus :

<https://developer.mozilla.org/fr/docs/Web/HTML/Element/Input>  
<https://developer.mozilla.org/fr/docs/Web/HTML/Element/Input/button>

Voici un exemple d'utilisation

ENQUETE

nom :

prénom :

âge :

pour les vacances cliquer sur ce bouton

contrelesvacances cliquer sur ce bouton

```
Éléments Console Débugueur Réseau Perform
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8" />
    <title>Voici mon site</title>
  </head>
  <body>
    <h>ENQUETE</h>
    <form>
      <p>
        nom :
        <input type="text" />
      </p>
      <p>
        prénom :
        <input type="text" />
      </p>
      <p>
        âge :
        <input type="text" />
      </p>
      <p>
        pour les vacances cliquer sur ce bouton
        <input type="checkbox" />
      </p>
      <p>
        contrelesvacances cliquer sur ce bouton
        <input type="checkbox" />
      </p>
      <button>pour valider Cliquez ici !</button>
    </form>
  </body>
</html>
```

### **Les balises div et span**

Ces 2 balises très utilisées (surtout la balise div), sont des balises dites “génériques”.

À quoi servent-elles alors ?

- À organiser la page, à regrouper plusieurs balises dans une même entité.
- Et par le biais d’un attribut de classe ou d’identité, d’associer à la zone couverte des propriétés CSS de présentation.

Quelle est la différence entre ces deux balises?

Parce que div est une balise de type “block” et que span est une balise de type “inline” !

Au moment de l’interprétation du code HTML, les contenus des balises de type "block" (p, div,h1,...) se placent automatiquement sur la page web les uns en dessous des autres.

Les contenus des balises de type "inline" (strong, img, span, a) se placent sur la page web les uns à côté des autres.

**Copier et coller dans un nouveau fichier le code suivant, pour tester la page web qui en découle**

```
<!doctype html>
<html lang="fr">
  <head>
    <meta charset="utf-8">
    <title>block/inline</title>
  </head>
  <body>
    <div>div est une balise de type "block"</div>
    <p>la balise p est une autre balise de type block</p>
    <span>En revanche, span est une balise de type "inline"</span>
    <a href="https://www.google.fr">Et voici une autre balise de type
"inline"</a>
    <h1>h1 est bien une balise de type "block"</h1>
    <span>la malheureuse balise span est "obligée" de se placer en
dessous du block </span>
    <span>mais la balise span se placera sur la même ligne à la suite
de la span précédente.</span>
  </body>
</html>
```

## Les listes avec les balises <ul> <ol> <li>

Pour pouvoir proposer des listes sur l'écran, il faut réfléchir pour savoir si l'on veut qu'elles soient ordonnées c'est-à-dire avec les items numérotés ou pas : juste des points.

Dans le premier cas la listes sera encadré par les bornes <ol> et </ol> dans le second par <ul> et </ul> .

Puis on encadrera chaque item par les bornes <li> ... </li>

```
<ul>
  <li>padding-left : marge gauche du padding Css.</li>
  <li>padding-right : marge droite du padding Css.</li>
  <li>padding-top : marge haute du padding Css.</li>
  <li>padding-bottom : marge basse du padding Css.</li>
</ul>
```

- padding-left : marge gauche du padding Css.
- padding-right : marge droite du padding Css.
- padding-top : marge haute du padding Css.
- padding-bottom : marge basse du padding Css.

```
<ol>
  <li>margin-left : marge gauche du margin Css.</li>
  <li>margin-right : marge droite du margin Css.</li>
  <li>margin-top : marge haute du margin Css.</li>
  <li>margin-bottom : marge basse du margin Css.</li>
</ol>
```

1. margin-left : marge gauche du margin Css.
2. margin-right : marge droite du margin Css.
3. margin-top : marge haute du margin Css.
4. margin-bottom : marge basse du margin Css.

## Les balises <ul> <ol> <li>

### Mise en pratique

Il sera intéressant dans votre projet d'utiliser les deux types de listes. Ça peut être intéressant pour faire par exemple le classement de vos albums préféré de l'artiste (liste ordonnée) ou pour faire une liste d'informations non hiérarchisées comme la taille le poids, la couleur des yeux, ...

## Les tableaux : <table> <tr> <td>

Les tableaux peuvent être pratiques pour ranger des informations (et aussi des images).

On utilise les balises <table> ... </table> pour encadrer le tableau dans sa globalité.

Puis on y va ligne par ligne, et :

chaque ligne est encadrée par <tr>...</tr>

chaque élément est encadré par <td>...</td>

## CSS

```
table{  
  border-collapse: collapse;}  
td, th /* Mettre une bordure sur les td ET les th */  
{  
  border: 1px solid black;}
```

## HTML

```
<table>  
  <caption>Passagers du vol 377</caption>  
  <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>  
  
  <tr>
```

Passagers du vol 377

Nom	Âge	Pays
Carmen	33 ans	Espagne
Michelle	26 ans	États-Unis
François	43 ans	France
Martine	34 ans	France
Jonathan	13 ans	Australie
Xu	19 ans	Chine

Plus sur les tableaux :

<https://developer.mozilla.org/fr/docs/Web/HTML/Element/table>

## Encadrer vos blocks à l'aide du CSS

Une page dont les informations s'étalent du bord gauche au bord droit, ce n'est ni beau ni agréable à consulter. Il peut être intéressant de gérer les zones dans lesquelles on va s'exprimer.



Pour chaque bloc on peut agir sur les marges extérieures (margin), le contour (border), et le rembourrage intérieur (padding).

Pour les marges on peut contrôler les quatre dimensions généralement dans le sens des aiguilles d'une montre à partir du haut (autrement dit : haut, droit, bas, gauche)

Pour la bordure on peut arrondir les coins à l'aide de « border-radius » suivi du rayon de l'arrondi, exprimé généralement en pixels. On peut aussi contrôler l'épaisseur de la bordure, et son style.

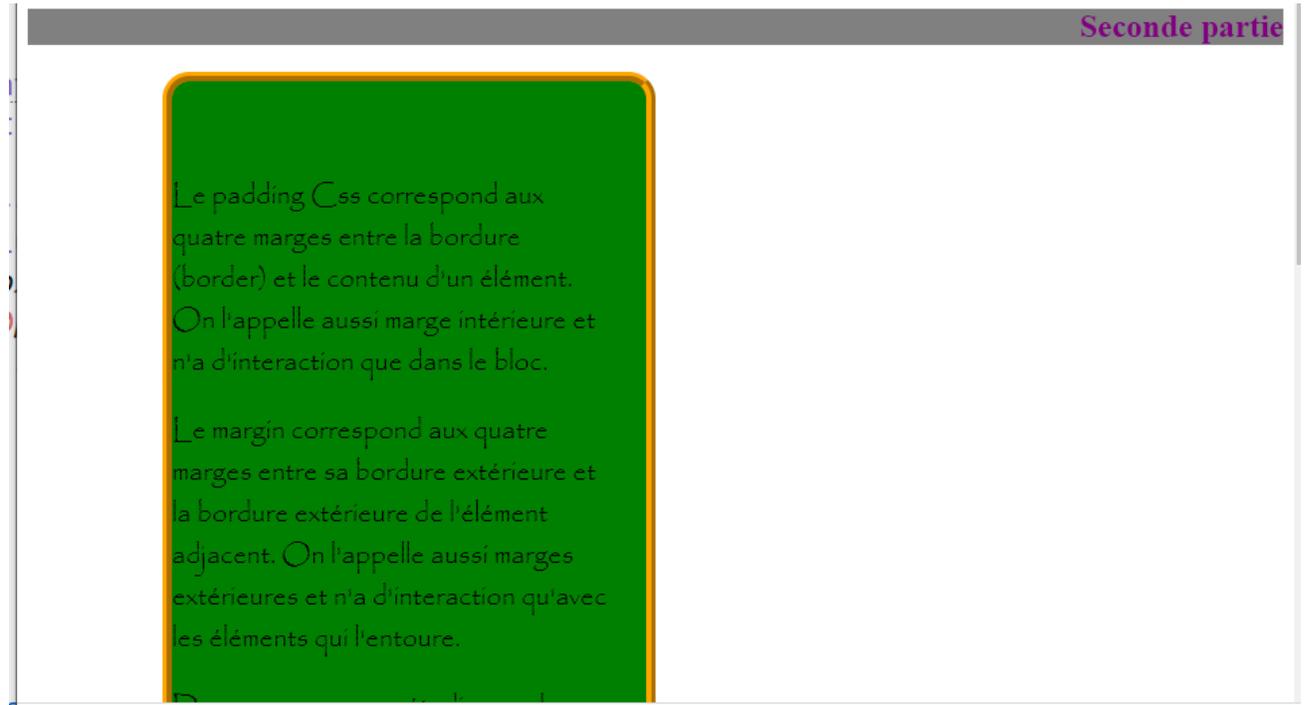
<https://www.zonecss.fr/cours-css/les-marges-css-avec-padding-et-margin.html>

[https://www.w3schools.com/css/css\\_border.asp](https://www.w3schools.com/css/css_border.asp)

# Encadrer vos blocks à l'aide du CSS

## Exemple

```
#second{  
  margin-right : 50%;  
  margin-left : 100px;  
  margin-top : 20px;  
  background-color: green;  
  border-radius: 20px;  
  border : 7px ridge orange;  
  padding: 50px 20px 10px 0px;  
  font-size: 20px;  
}
```



On remarquera ici que notre cadre s'arrête à mi page à droite : `margin-right : 50%`;

Il commence aussi à 100 pixel du bord droit (`margin-left : 100px`) et 20pixel du block qui le précède (`margin-top : 20px` ; )

Pour ce qui est du padding plutôt que de préciser `padding-top`, `padding-right` ... j'ai mis tous mes paramètres dans le bon ordre (voir diapositive précédente) à la suite de « `padding`: ».



MAINTENANT VOUS AVEZ QUELQUES BASES  
POUR ÉLABORER VOTRE PROPRE SITE WEB.



EVIDEMMENT IL EST RECOMMANDÉ D'ENRICHIR VOTRE  
SITE AVEC D'AUTRES CONNAISSANCES QUE VOUS AUREZ  
GLANÉ AILLEURS (AVEC VOTRE MOTEUR DE RECHERCHE,  
DÉCRIVEZ CE QUE VOUS VOULEZ FAIRE PUIS ÉCRIVEZ HTML  
ET VOUS TROUVEREZ SANS DOUTE VOTRE BONHEUR.