



WEB - Conception d'un gabarit CSS

[Mise à jour le 1/5/2022]

- **Sources**
 - MDN web docs
 - [La mise en page avec le CSS](#)
 - [Introduction à la mise en page en CSS](#)
- **Ressources**
 - CSS facile ! - [Propriétés CSS](#)
- **Liens connexes**
 - [WEB - Les bases du HTML](#)
 - [WEB - Les bases du CSS](#)

-
- **Mots-clés** : boîte, contenu, marge interne et externe, bord, positionnement.
-

Introduction

Un **gabarit css** se compose de la **structure** HTML et de la **présentation** CSS d'un site Web.
Il y a 2 choix concernant la structure générale du site :

- Structure fluide s'étirant sur toute la largeur de l'écran (non traité ici)
- **Structure fixe (largeur fixe, centrée ou non)**



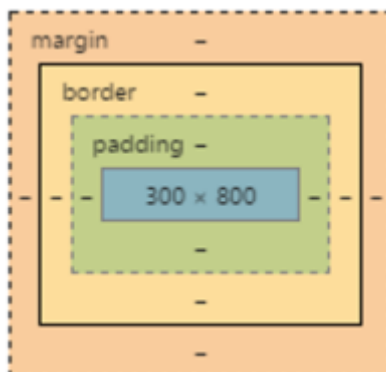
Pour réaliser le gabarit d'un site, il est nécessaire de connaître le modèle des boîtes CSS et de disposer d'une méthode de conception. Ces deux points sont décrits dans la suite du document.

1. Comprendre le modèle de boîte

1.1 Présentation



En **HTML**, chaque élément est considéré comme une **boîte**. Aux dimensions induites par son **contenu** s'ajoutent souvent les espaces de **marges externes** (**margin**) ou **internes** (**padding**) et une **bordure** (**border**).

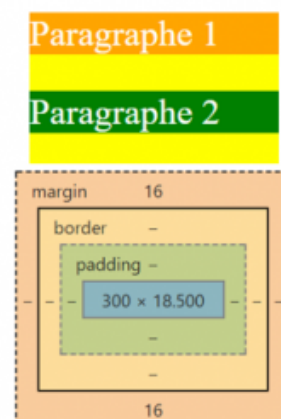


Les propriétés **border**, **margin** et **padding** ne sont pas obligatoires. Non renseignés, elles prendront une valeur par défaut, toujours **nulle** pour les éléments en ligne. En revanche **parmi les éléments de type bloc, seul <div> n'a pas de marge par défaut**.

Exemple

```
index1.html
<body>
  <div id="page">
    <p id="p1">Paragraphe 1</p>
    <p id="p2">Paragraphe 2</p>
  </div>
</body>

style1a.css
#page {
  width: 300px;
  height: 200px;
  background-color: yellow;
}
#p1 {
  background-color: orange;
}
#p2 {
  background-color: green;
}
```



Les **paragraphes** s'affichent les **un en dessous des autres**, l'espace les séparant correspond à leurs marges externes (16px dans l'exemple ci-dessus).

1.2 Reset CSS



Pour supprimer les marges de tous les éléments, il suffit d'écrire le code suivant dans une feuille de style.

*.css

```
* {  
  /* Reset CSS */  
  margin : 0px ;  
}
```



Lors de la mise en page, la suppression des marges de tous les éléments permet de **ne pas être gêné par des dimensions non choisies**.

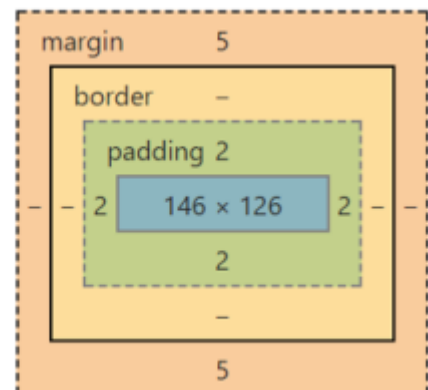
1.3 Calcul de la place occupée



La place occupée par une boîte est calculée en ajoutant les marges internes (**padding**), externes (**margin**) et l'épaisseur de la bordure (**border**).

Exemple : La boîte ci-contre occupe :

- une largeur $l = 2 * 2 + 146 = 150\text{px}$ et
- une hauteur $h = 2 * 5 + 2 * 2 + 126 = 140\text{px}$



2. Méthode proposée pour réaliser un gabarit (dimensionné en pixels)

"Un gabarit, souvent nommé template, est un patron de mise en page où l'on place images et textes. Aussi, il est souvent utilisé de manière répétitive pour créer des documents présentant une même structure." Wikipédia.



Conseil : afficher le code HTML, le code CSS et le résultat dans le navigateur sur le même écran pour faciliter la conception.

2.1 Création de la maquette graphique

On **dessine** la **maquette (Fig1a)** du site à la main et on **dimensionne** les boîtes (**Fig1b**). Les dimensions sont choisies en tenant compte de leur contenu.

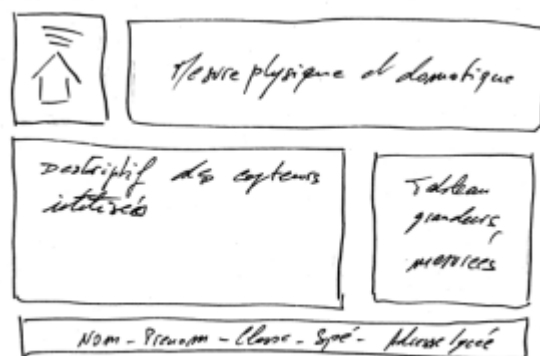


Fig1a

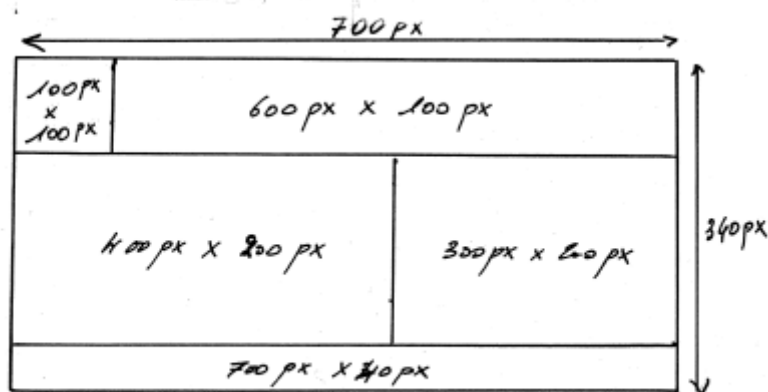


Fig1b

2.2 Mise en page

1. On **crée** la **structure** de la page (code **HTML**).
2. On présente le contenu de la page avec **une** ou plusieurs feuilles de style (code **CSS**) en procédant par étape.

Étape 1 - Faire apparaître les blocs

Supprimer les **marges extérieures par défaut** de toutes les balises (**margin**). Déclarer et dimensionner les **conteneurs** (header, footer, section, div, etc.). Faire apparaître les conteneurs avec des couleurs (**Fig2**).



Exemple partiel

*.css

```
/* Etape 1 */
* {
  /* Reset CSS */
  margin: 0px;
}

div#page { /* Dimensionnement et visualisation
du conteneur page */
  /* Etape 1 */
  background-color: yellow;
  width: 700px;
  height: 340px;
}

div#logo {}

div#banner {}

div#content {}

div#aside {}

div#footer {}
```

Étape 2 - Placer les blocs

Placer les **conteneurs** conformément à la maquette graphique. (**Fig3**)

Exemple partiel



Fig3

*.CSS

```
div#logo {
    /* Etape 1 */
    background-color: lightblue;
    width: 100px;
    height: 100px;
    /* Etape 2 */
    /* On fait flotter la boîte à gauche */
    float: left;
}

div#banner {
    /* Etape 1 */
    background-color: lightgreen;
    width: 600px;
    height: 100px;
    /* Etape 2 */
    /* On fait flotter la boîte à gauche */
    float: left;
}

div#content {
    /* Etape 1 */
    background-color: lightgray;
    width: 400px;
    height: 200px;
    /* Etape 2 */
    /* On fait flotter la boîte à gauche */
    float: left;
    /* mais en dessous de banner */
    clear: both;
}
```

Étape 3 - Eloigner les blocs (éventuellement)

Affiner le positionnement des blocs en plaçant



Fig4

Exemple

*.css

```
div#logo {
    /* Etape 1 */
    background-color: lightblue;
    /* width est calculé en fonction de margin : 100-2x5 */
    width: 90px;
    /* height est calculé en fonction de margin : 100-2x5 */
    height: 90px;
    /* Etape 2 */
    float: left;
    /* Etape 3 */
    margin: 5px;
}

div#banner {
    /* Etape 1 */
    background-color: lightgreen;
    /* width est calculé en fonction de margin : 600-2x5 */
    width: 590px;
    /* height est calculé en fonction de margin : 100-2x5 */
    height: 90px;
    /* Etape 2 */
    float: left;
    /* Etape 3 */
    margin: 5px;
}
```



Pour conserver l'occupation initiale des boîtes sur la page, il faut soustraire les marges au contenu.

Étape 4 - Simuler le contenu textuel

Simuler le rendu avec du texte "**Lorem Ipsum**" et placer les marges intérieures (**padding**). (Fig5)

Exemple



Fig5

*.CSS

```
div#content {
  /* Etape 1 */
  background-color: lightgray;
  /* A l'étape 3, width est calculé en fonction de margin :
  400-2x5 */
  width: 390px;
  height: 200px;
  /* Etape 2 */
  clear: both;
  float: left;
  /* Etape 3 */
  margin-left: 5px;
  margin-right: 5px;
  /* Etape 4 */
  padding: 10px;
  /* Redimensionnement automatique de la boîte */
  /* pour conserver la place occupée initialement */
  box-sizing: border-box;
  text-align: justify;
  font-size: 16px;
}
```

Étape 5 - Faire les finitions



Fig6

Remplacer le texte “Lorem Ipsum” et faire les **finitions** (bords arrondis, bordures, taille de texte, police de caractère factoriser le code, etc.) (**Fig6**).

Exemple partiel

*.css

```
body {
  /* Etape 4 */
  font-family: "Trebuchet MS", Arial, Helvetica,
  sans-serif;
  /* Etape 5 */
  background-color: rgb(65, 106, 242, 0.2);
  font-size: 14px;
}

div#page {
  /* Etape 1 */
  width: 700px;
  /* Etape 5 */
  /* height: 340px; est supprimé à l'étape 5 */
  /* margin centre la page */
  margin: 0px auto;
}
```

Astuces



- Pour **centrer verticalement** le texte dans une boîte, remplacer height par **line-height**.
- Pour **centrer horizontalement** le texte dans une boîte, utiliser : **text-align:center**;
- Pour **centrer horizontalement** une boîte sur l'écran (par exemple la page), utiliser : **margin: 0 auto**;
- Pour **redimensionner automatiquement** une boîte après lui avoir appliqué du padding, utiliser : **box-sizing:border-box**;
- Pour **centrer une image**, utiliser **margin:auto; display:flex**;

3. Code de la démonstration

Télécharger le code de la démonstration "**Comprendre le modèle de boîte**"

From:

<http://webge.fr/dokuwiki/> - **WEBGE Wikis**

Permanent link:

<http://webge.fr/dokuwiki/doku.php?id=web:positioncss&rev=1652175514>

Last update: **2022/05/10 11:38**

