

HTML 5

- Squelette d'une page HTML
- Balises de base
- Tableaux

Introduction

Le langage HTML (*HyperText Markup Language*) est un langage de formatage de texte adapté à la conception de pages web.

Ce n'est pas un langage de programmation, mais un langage de mise en forme de données : une page Web décrit une structure statique, tandis qu'un programme est un processus dynamique.

Pas besoin d'avoir des bases de programmation pour écrire du HTML.

Le HTML définit la structure logique d'un document, permettant ensuite à un navigateur (Chrome, Firefox, etc.) de rendre visuellement (mais aussi auditivement ou tactilement, pour les non-voyants) le document.

la norme actuelle recommandée par le W3C est le HTML 5

L'élément de base d'un site est la page. C'est un fichier dont l'extension est normalement .html (ou .htm) et qui contient du code HTML.

Le HTML ne s'occupe que de définir la structure logique de votre document. Pour l'aspect visuel (mise en page, couleurs, fontes, etc.), c'est le domaine des feuilles de style.

HTML - Principes fondamentaux : Les balises

Les noms des balises et des attributs sont en minuscules.

On écrit : `<p>` et plus `<P>`.

Les valeurs des attributs sont entre doubles quotes.

On écrit : `<p align="center">` et plus `<p align=center>`.

Tout attribut doit impérativement avoir une valeur.

On écrit : `<table border="1">` et plus `<table border>`.

Toute balise ouvrante doit être refermée.

On écrit : `<p>blabla</p>` et plus `<p>blabla`.

Toutes les balises sont ouvrantes.

On écrit : `
` et plus `
`

On écrit `<hr width="50%" />` et plus `<HR WIDTH=50%>`.

Les balises doivent être correctement imbriquées.

On écrit : `<p><i>blabla</i></p>` et pas `<p><i>blabla</p></i>`.

Squelette d'une page HTML - DTD

Toute page HTML est contenue entre les 2 balises <html> et </html>.

1- Le DTD

Toute page HTML doit commencer par une déclaration de *Document Type Definition* (DTD), qui précise au navigateur quelle version de HTML est utilisée :

HTML 5

```
<!DOCTYPE html>
```

Squelette d'une page HTML - Langue

2. Langue de la page

```
<html lang="fr" />
```

fr pour le français, **en** pour l'anglais, **es** pour l'espagnol, **ar** pour l'arabe. Déclarer la langue utilisée est utile pour les synthétiseurs vocaux des navigateurs pour aveugles (qui en déduisent comment prononcer). Pour les navigateurs standards qui peuvent en déduire s'il faut utiliser des guillemets anglo-saxons (") ou français (« »).

Pour l'instant, la page-squelette ressemble à :

```
<!DOCTYPE html>
<html lang="fr" />

<!-- contenu de la page. -->

</html>
```

Squelette d'une page HTML - head

Le rôle des en-têtes dans une page HTML est de donner des méta-informations plus ou moins importantes.

Exp: l'identité de l'auteur, le titre de la page, l'encodage, la langue utilisée, etc.

Les en-têtes sont déclarées à l'intérieur de l'élément **<head>**.

Squelette d'une page HTML - title

Le titre est affiché par les navigateurs dans la barre. Il est également utilisé par les moteurs de recherche. Un bon titre est informatif, tout en restant relativement court. Il se déclare dans l'élément **<title>** qui est *obligatoire*.

```
<title> mon titre </title>
```


Squelette d'une page HTML - meta

Les **meta** sont facultatives, mais importantes à la fois pour le navigateur qui a des renseignements sur la manière dont afficher la page, et pour les moteurs de recherche, qui s'en servent pour indexer vos pages:

Auteur de la page

```
<meta name="author" content="Hmida Hamid" />
```

Vous pouvez déclarer plusieurs auteurs en séparant les noms par des virgules.

Squelette d'une page HTML - meta

Encodage de la page

```
<meta charset="UTF-8" />
```

Ceci déclare que votre page est du HTML respecte l'encodage UTF-8 (le plus utilisé). La déclaration de l'encodage est importante pour le navigateur.

Squelette d'une page HTML - meta

Mots clefs

```
<meta name="keywords" content=" mot-clef1, mot-clef2, mot-clef3" />
```

Cette balise *meta* est utilisée par certains moteurs de recherche pour indexer les pages. Ne mettez que les mots-clefs les plus significatifs, les moteurs n'aiment pas les listes trop longues.

Description

```
<meta name="description" content=" un texte de description de votre page" />
```

Les références externes

Elles sont utilisées afin d'indiquer qu'un autre document est lié au document en cours.

Cela peut être un script, une feuille de style ou encore un icône.

Référence à un script

Pour faire référence à un script, il faut utiliser la balise `<script>`.

```
<script src=javascript/mon_script.js"></script>
```

4.2. La balise <link>

La balise <link> permet de faire référence à de nombreux types de documents, mais les principaux sont les feuilles de style (CSS) et l'icône.

- Pour un fichier CSS:

```
<link rel="stylesheet" href="css/fichier.css" />
```

- Pour un icône:

```
<link rel="icon" href="images/icon.png" />
```

Squelette d'une page HTML - body

Le corps de la page est ce que le lecteur va voir *via* son navigateur.

C'est le contenu entre balises **<body>**.

Squelette d'une page HTML

```
<!doctype html>
<html lang="fr">
<head>
  <meta charset="utf-8">
  <title>Titre de la page</title>
  <link rel="stylesheet" href="style.css">
  <script src="script.js"></script>
</head>
<body>
  ...
  <!-- Le reste du contenu -->
  ...
</body>
</html>
```

Balise lien: <a>

<a> Insère un **lien** ou un **point d'ancrage**.

Exemple :

- Un lien :

```
<a href="https://iwa.fst.ac.ma">DCESS IWA</a>
```

- Un point d'ancrage :

```
<a href="#intro">Introduction</a>
```

```
<!-- Plus loin dans le code -->
```

```
<a id="intro"></a>
```


Balise saut de ligne: `
`

La balise `
` provoque un saut de ligne.

Exemple :

Du texte `
`
La suite ...

Balises gras et mise en emphase

Balise **** : mise en gras d'un texte.

Exemple :

Filière **SMI**.

Balise **** : mise en emphase Met en emphase du texte.

Exemple :

Vive SMI et **moi** aussi !

Balises paragraphe <p> et trait horizontal

Balise <p> : paragraphe

Définit un paragraphe.

Exemple :

```
<p>Un paragraphe</p>
```

Balise <hr /> : trait horizontal

Insère un trait horizontal.

Exemple :

```
<p>Premier paragraphe</p>  
<hr />  
<p>Deuxième paragraphe</p>
```

Balise ``

Couplée à du CSS, la balise *span* met en forme du texte.

Exemple :

```
<span style="text-decoration:line-through">  
  Du texte barré  
</span>  
et du texte normal.
```

Balises des titres <h1> à <h6>

Balises <h1> à <h6> : entête de paragraphe

Utilité :

Définit un entête de paragraphe de niveau X (X pouvant aller de 1 à 6).

Exemple :

```
<h1>Titre 1</h1>  
<h2>Titre 2</h2>  
<h3>Titre 3</h3>  
<h4>Titre 4</h4>  
<h5>Titre 5</h5>  
<h6>Titre 6</h6>
```

Balises puces et numéros

La balise Définit une liste.

La balise *ul* étant une liste à puces.

La balise *ol* étant une liste ordonnée avec des chiffres.

Exemple :

```
<ul>
  <li>Menu</li>
  <li>Entrée</li>
  <li>Plat principal</li>
  <li>Plat secondaire</li>
  <li>Dessert (important)</li>
</ul>
```

Balises image

Insère une image.

Exemple :

```

```

Balise <pre>

Définit un bloc où le texte sera affiché tel qu'il figure dans le fichier source.

Exemple :

```
<pre>
#include <stdio.h>
int main(void)
{
    printf("Hello, world!\n");
    return 0;
}
</pre>
```


Balise <div>

La balise <div> divise un document

Joue le rôle de "conteneur" en permettant de structurer la page en plusieurs blocs.

Exemple :

```
<div>  
  <div> Le haut de page </div>  
  <div> Le contenu principal de la page </div>  
  <div> Le bas de page </div>  
</div>
```

Balise tableau <table>

Les tableaux sont décrits par l'élément <table>.

Un tableau simple

<table> définit l'ensemble du tableau. À l'intérieur, il faut décrire les lignes du tableau et ses cellules.

L'élément <tr> décrit les lignes, et <td> les cellules.

On déclare dans l'ordre <table>, puis <tr> et enfin <td>.

```
<table>
  <tr><td> Cellule 1 </td><td> Cellule 2 </td></tr>
  <tr><td> Cellule 3 </td><td> Cellule 4 </td></tr>
  <tr><td> Cellule 5 </td><td> Cellule 6 </td></tr>
</table>
```

Ce qui donne :

Cellule 1 Cellule 2

Cellule 3 Cellule 4

Cellule 5 Cellule 6

Balise tableau <table>

```
<table border=1>  
  <caption>Ceci est un tableau</caption>  
  <tr><th> Colonne gauche </th><th> Colonne droite</th></tr>  
  <tr><td> Cellule 1 </td><td> Cellule 2</td></tr>  
  <tr><td> Cellule 3 </td><td> Cellule 4</td></tr>  
  <tr><td> Cellule 5 </td><td> Cellule 6</td></tr>  
</table>
```

Ceci est un tableau

Colonne gauche	Colonne droite
Cellule 1	Cellule 2
Cellule 3	Cellule 4
Cellule 5	Cellule 6

Tableaux évolués

Les balises **<thead>**, **<tbody>** et **<tfoot>** sert à différencier le contenu des entêtes.

<thead>: C'est elle qui contient l'entête du tableau, en général, la partie du tableau dont les cellules sont définies par des **<th>**.

Type de balise : table-header-group

Attributs pouvant être employés : scope, lang, dir, title.

<tfoot>: Elle a la même fonction que **<thead>**, mais cette fois-ci, pour le pied du tableau.

Type de balise : table-footer-group

Attributs pouvant être employés : lang, dir, title.

<tbody>: Elle permet de délimiter la partie du tableau contenant uniquement les données.

Type de balise : table-row-group

Attributs pouvant être employés : lang, dir, title.

Tableaux – fusion des cellules

l'attribut **colspan**: pour fusionner des colonnes l'attribut **rowspan** : pour fusionner des lignes

```
<table>
<caption>Le titre du tableau </caption>
<thead> <!-- En-tête du tableau -->
    <tr><th> Colonne gauche </th><th> Colonne droite</th></tr>
</thead>
<tfoot> <!-- Pied de tableau -->
    <tr><th>pied gauche</th><th>pied droit</th> </tr>
</tfoot>

<tbody> <!-- Corps du tableau -->
    <tr><td colspan="2"> Cellule 1 et Cellule 2</td></tr>
    <tr><td rowspan="2"> Cellule 3 et Cellule 5 </td><td> Cellule 4</td></tr>
    <tr><td> Cellule 6</td></tr>
</tbody>
</table>
```

Le titre du tableau

Colonne gauche	Colonne droite
Cellule 1 et Cellule 2	
Cellule 3 et Cellule 5	Cellule 4
	Cellule 6
pied gauche	pied droit

Ingénierie du Web Avancé (IWA)

Module 1 : Initiation web: HTML – CSS – PHP

HTML 5

- **Formulaire**

<form> :C'est la balise principale du formulaire

```
<p>Texte avant le formulaire</p>
<form method="post" action="traitement.php">
<p>Texte à l'intérieur du formulaire</p>
</form>
<p>Texte après le formulaire</p>
```

method : cet attribut indique par quel moyen les données vont être envoyées **method="get"** : les informations seront envoyées dans l'adresse

de la page (http://...),

method="post" : elle permet d'envoyer un grand nombre d'informations. Les données saisies dans le formulaire ne transitent pas par la barre d'adresse.

action : c'est l'adresse de la page ou du programme qui va traiter les informations

Formulaire: Zone de texte et libellé

Zone de texte monoligne

```
<input type="text" name="pseudo" />
```

Libellé

```
<label>Votre pseudo</label> : <input type="text" name="pseudo" />
```

Pour lier le label au champ, il faut lui donner un attribut for qui a la même valeur que l'id du champ

```
<label for="pseudo">Votre pseudo</label> : <input type="text" name="pseudo" id="pseudo" />
```


Formulaire: Quelques attributs supplémentaires

On peut agrandir le champ avec `size`.

- On peut limiter le nombre de caractères que l'on peut saisir avec **`maxlength`**.
- On peut pré-remplir le champ avec une valeur par défaut à l'aide de **`value`**.
- On peut donner une indication sur le contenu du champ avec **`placeholder`**. Cette indication disparaîtra dès que le visiteur aura cliqué à l'intérieur du champ.

```
<label for="pseudo">Votre pseudo :</label>  
<input type="text" name="pseudo" id="pseudo" placeholder="Ex : toto" size="30"  
maxlength="10" />
```

Formulaire: Zone de mot de passe

```
<label for="pseudo">pseudo :</label>  
  <input type="text" name="pseudo" id="pseudo" /> <br />  
<label for="pass">mot de passe :</label>  
  <input type="password" name="pass" id="pass" />
```

Formulaires: Zone de texte multiligne

```
<label for="comment">Commentaire</label><br />  
<textarea name="ameliorer" id="comment"></textarea>
```

Formulaire: Checkbox

Cochez les aliments que vous aimez manger :

`<input type="checkbox" name="frites" id="frites" /> <label for="frites">Frites</label>
`

`<input type="checkbox" name="steak" id="steak" /> <label for="steak">`

`Steak haché</label>
`

`<input type="checkbox" name="epinards" id="epinards" /> <label for="epinards">`

`Epinards</label>
`

`<input type="checkbox" name="huitres" id="huitres" /> <label for="huitres">Huitres</label>`

Cochez les aliments que vous aimez manger :



Frites



Steak haché



Epinards



Huitres

Formulaire: Zone d'options

Veillez indiquer la tranche d'âge dans laquelle vous vous situez :

<input type="radio" name="age" value="moins15" id="moins15" />

**<label for="moins15">Moins de 15 ans</label>
**

<input type="radio" name="age" value="medium15-25" id="medium15-25" />

**<label for="medium15-25">15-25 ans</label>
**

<input type="radio" name="age" value="medium25-40" id="medium25-40" />

**<label for="medium25-40">25-40 ans</label>
**

<input type="radio" name="age" value="plus40" id="plus40" />

<label for="plus40">Encore plus vieux que ça ?!</label>

Veillez indiquer la tranche d'âge dans laquelle vous vous situez :

- ☐ Moins de 15 ans
- ☐ 15-25 ans
- ☒ 25-40 ans
- ☐ Encore plus vieux que ça ?!

Formulaire: Listes déroulantes

```
<form method="post" action="traitement.php">
<p> <label for="pays">Dans quel pays habitez-vous ?</label><br />
<select name="pays" id="pays">
  <option value="france">France</option>
  <option value="espagne">Espagne</option>
  <option value="italie">Italie</option>
  <option value="royaume-uni">Royaume-Uni</option>
  <option value="canada">Canada</option>
  <option value="etats-unis">États-Unis</option>
  <option value="chine">Chine</option>
  <option value="japon">Japon</option>
</select>
</p>
</form>
```

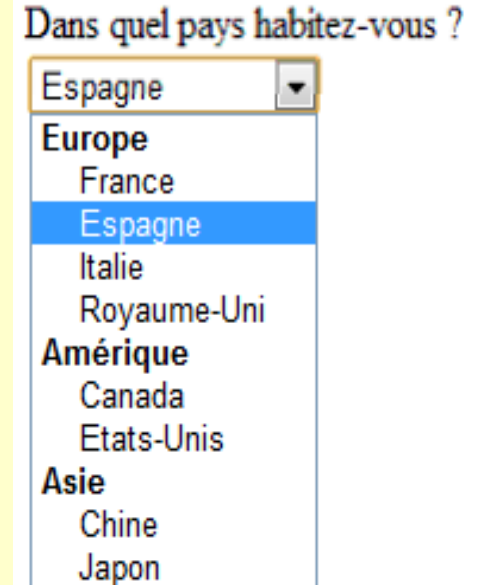
Dans quel pays habitez-vous ?



Formulaire: Listes déroulantes

```
<label for="pays">Dans quel pays habitez-vous ?</label><br />
<select name="pays" id="pays">
<optgroup label="Europe">
  <option value="france">France</option>
  <option value="espagne">Espagne</option>
  <option value="italie">Italie</option>
  <option value="royaume-uni">Royaume-Uni</option> </optgroup>
<optgroup label="Amérique">
  <option value="canada">Canada</option>
  <option value="etats-unis">Etats-Unis</option> </optgroup>
  <optgroup label="Asie"> <option value="chine">Chine</option>
  <option value="japon">Japon</option> </optgroup>
</select>
```

Dans quel pays habitez-vous ?



Formulaire: Autres options

Sélectionner automatiquement un champ

```
<input type="text" name="prenom" id="prenom" autofocus />
```

Rendre un champ obligatoire

```
<input type="text" name="prenom" id="prenom" required />
```

Le bouton d'envoi

```
<input type="submit" value="Envoyer" />
```



Formulaires: zones de saisie enrichies

```
<input type="email" />
```

dsqfdfsdfsp



Un champ e-mail mal renseigné est entouré de rouge dans Firefox
les navigateurs mobiles sur iPhone et Android, affichent un clavier adapté à la saisie d'e-mail

```
<input type="url" />
```



Clavier de saisie d'URL sur iPhone

Formulaires: zones de saisie enrichies

Téléphone

```
<input type="tel" />
```

Nombre

```
<input type="number" />
```

Un curseur

```
<input type="range" />
```

Couleur

```
<input type="color" />
```

Date

```
<input type="date" />
```

Recherche

```
<input type="search" />
```

CSS 3

- **Fonctionnement**
- **Arrière plan**
- **Police des caractères**

Cascading Style Sheets

Les feuilles de style en cascade CSS « **Cascading Style Sheets** » (CSS) est un outil pour la présentation des sites Web. Elles peuvent faire gagner beaucoup de temps, et elles permettent de créer des sites Web de façon entièrement nouvelle.

L'utilisation de CSS **nécessite une connaissance de base de HTML.**

Comment CSS fonctionne-t-il ?

La syntaxe CSS de base

Exp: le fond d'une page Web rouge :

Avec **HTML** :

```
<body bgcolor="#FF0000">
```

Avec **CSS** :

```
body {background-color: #FF0000;}
```

```
selector {property: value;}
```

↑
À quelle(s)
balise(s) HTML
la propriété
s'applique
(p.ex. "body")

↑
La propriété, par
exemple, la couleur
d'arrière-plan
("background-color")

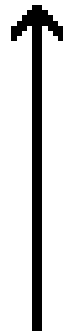
↖ La valeur de la
propriété de couleur
d'arrière-plan, par
exemple, le rouge
("#FF0000")

Modèle fondamental de CSS

```
selector {property: value;}
```



À quelle(s)
balise(s) HTML
la propriété
s'applique
(p.ex. "body")



La propriété, par
exemple, la couleur
d'arrière-plan
("background-color")



La valeur de la
propriété de couleur
d'arrière-plan, par
exemple, le rouge
("#FF0000")

Méthodes d'appliquer CSS à un document HTML

Il existe 3 façons

Méthode 1 : Dans la ligne (l'attribut style)

Avec l'attribut HTML style.

Exp :

```
<html>
<head>
  <title>Exemple</title>
</head>
<body style="background-color: #FF0000;">
  <p>Cette page est rouge</p>
</body>
</html>
```

Méthodes d'appliquer CSS à un document HTML

Méthode 2 : Interne (l'élément style)

Une autre méthode consiste à inclure le code CSS avec la balise HTML `<style>`.

Exp:

```
<html>
<head>
  <title>Exemple</title>
  <style type="text/css">
    body {background-color: #FF0000;}
  </style>
</head>
<body>
  <p>Cette page est rouge</p>
</body>
</html>
```


Méthodes d'appliquer CSS à un document HTML

Méthode 3 : Externe (un lien vers une feuille de style)

La méthode recommandée est celle avec un lien vers feuille de style externe.

Une feuille de style externe est un fichier texte ayant l'extension « .css ».

Exemple:



On crée un lien depuis (default.htm) vers la feuille de style (style.css). Ce lien peut être créé en une ligne de code HTML :

```
<link rel="stylesheet" href="style/style.css" />
```

La ligne de code doit s'inscrire entre les balises <head> et </head>.

Couleurs et arrières-plans

La couleur d'avant-plan : la propriété 'color'

La propriété **color** décrit la couleur d'avant-plan d'un élément.
Par exemple, les titres rouge foncé. <h1>.

```
h1 { color: #ff0000; }
```

La propriété 'background-color'

La propriété **background-color** décrit la couleur d'arrière-plan.
Pour changer la couleur d'arrière-plan d'une page entière, il faudrait l'appliquer à l'élément <body>.
Exp: Les couleurs d'arrière-plan différentes sont appliquées aux éléments <body> et <h1>.

```
body { background-color: #FFCC66; }  
h1 { color: #990000;  
    background-color: #FC9804; }
```

Images d'arrière-plan [background-image]

background-image : insérer une image d'arrière-plan.

```
body
{ background-color: #FFCC66;
  background-image: url("butterfly.gif");
}

h1
{ color: #990000;
  background-color: #FC9804;
}
```

Arrière-plan [background-repeat]

Les quatre valeurs de background-repeat.

Valeur	Description
Background-repeat: repeat-x	L'image se répète horizontalement
background-repeat: repeat-y	L'image se répète verticalement
background-repeat: repeat	L'image se répète horizontalement et verticalement
background-repeat: no-repeat	L'image ne se répète pas

Arrière-plan [background-repeat]

Pour éviter la répétition d'une image d'arrière-plan:

```
body {  
  background-color: #FFCC66;  
  background-image: url("butterfly.gif");  
  background-repeat: no-repeat;  
}  
h1 {  
  color: #990000;  
  background-color: #FC9804;  
}
```

Arrière-plan [background-attachment]

background-attachment: pour fixer ou défiler l'image d'arrière-plan avec l'élément conteneur.

Valeur	Description
Background-attachment: scroll	L'image défile avec la page (non bloquée)
Background-attachment: fixed	L'image est bloquée

Arrière-plan [background-attachment]

Pour bloquer l'image d'arrière-plan.

```
body {  
    background-color: #FFCC66;  
    background-image: url("butterfly.gif");  
    background-repeat: no-repeat;  
    background-attachment: fixed;  
}  
h1 {  
    color: #990000; background-color: #FC9804;  
}
```

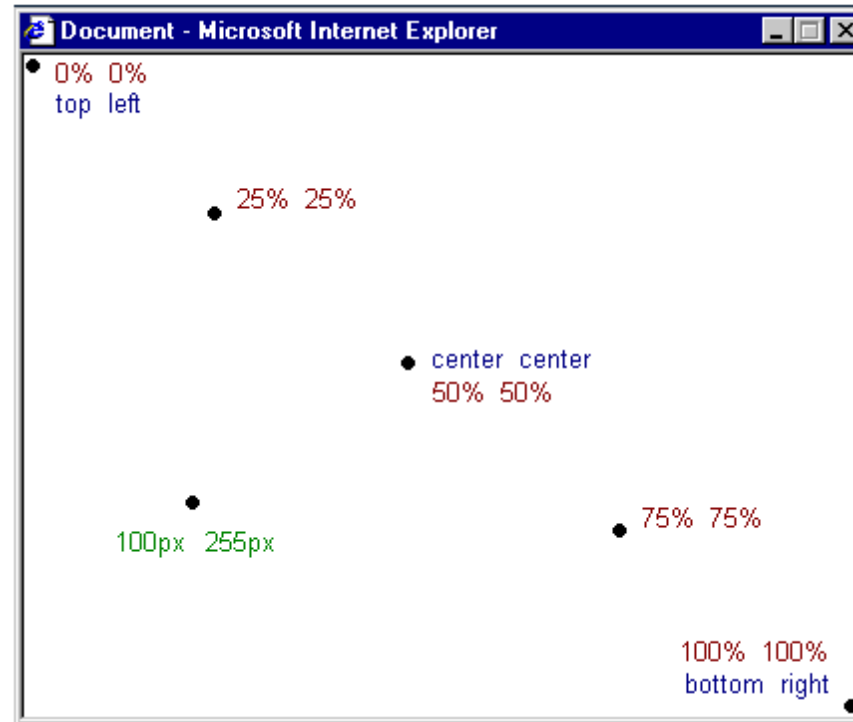
Positionner une image d'arrière-plan [background-position]

Par défaut, l'image d'arrière-plan se positionne dans le coin supérieur gauche de l'écran.

background-position : permet de changer la position prédéfinie et de placer l'image d'arrière-plan n'importe où à l'écran.

Exemple: la valeur '100px 200px' positionne l'image d'arrière-plan à 100px depuis la gauche et à 200px depuis le haut de la fenêtre du navigateur.

Les coordonnées peuvent s'exprimer en pourcentages de la largeur de l'écran, ou en unités fixes (pixels, centimètres, etc.), ou on peut utiliser les mots-clés "top", "bottom", "center", "left" ou "right".



Positionner une image d'arrière-plan [background-position]

Valeur	Description
background-position: 2cm 2cm	L'image est positionnée à 2 cm de la gauche et à 2 cm du haut de la page
background-position: 50% 25%	L'image est positionnée au centre et à un quart de la page vers le bas
background-position: top right	L'image est positionnée au coin supérieur droit de la page

Positionner une image d'arrière-plan [background-position]

L'exemple de code à suivre positionne l'image d'arrière-plan dans le coin inférieur droit de la page :

```
body {  
    background-color: #FFCC66;  
    background-image: url("butterfly.gif");  
    background-repeat: no-repeat;  
    background-attachment: fixed;  
    background-position: right bottom;  
}  
h1 {  
    color: #990000; background-color: #FC9804;  
}
```

On peut obtenir le même résultat avec background en une seule ligne de code :

```
background: #FFCC66 url("butterfly.gif") no-repeat fixed right bottom;
```

font-family : indique la liste prioritaire des polices à utiliser pour l'affichage d'un élément donné ou d'une page Web. Si la première police de la liste n'est pas disponible sur l'ordinateur utilisé pour accéder au site, la police suivante est essayée et ainsi de suite.

Polices [font-family]

La différence entre les polices :

Times New Roman

Garamond

Georgia

Ces trois familles de polices appartiennent au genre **serif**. Elles se caractérisent par leurs « empattements »

Trebuchet

Arial

Verdana

Ces trois familles de polices appartiennent au genre **sans serif**. Elles se caractérisent par l'absence d'« empattements »

Courier

Courier New

Andale Mono

Ces trois familles de polices appartiennent au genre **monospace**. Elles se caractérisent par une chasse fixe

Polices [font-family]

Il est recommandé de terminer la liste par une famille générique. Ainsi, la page s'affichera au moins avec une police de la même famille si aucune des polices indiquées n'était disponible.

Exp :

```
h1 {font-family: arial, verdana, sans-serif;}  
h2 {font-family: "Times New Roman", serif;}
```

Style des polices [font-style]

La propriété font-style définit le style de la police concernée être **normal**, **italic** ou **oblique**.

Exp:

```
h1 {font-family: arial, verdana, sans-serif;}  
h2 {font-family: "Times New Roman", serif;  
    font-style: italic;}
```

Variantes de polices [font-variant]

La propriété font-variant sert à choisir entre les variantes **normal** ou **small-caps** d'une police.

Une police **small-caps** utilise des lettres en capitales de taille réduite à la place des lettres en minuscules.

Exp. :

Sans Book SC
ABCABC

Sans Bold SC
ABCABC

Serif Book SC
ABCABC

Serif Bold SC
ABCABC

```
h1 {font-variant: small-caps;}  
h2 {font-variant: normal;}
```

Graisse des polices [font-weight]

Font-weight décrit le degré de graisse de la police. Elle peut avoir une graisse **normal** ou **bold**.

Exp:

```
p {font-family: arial, verdana, sans-serif;}  
td {font-family: arial, verdana, sans-serif;  
    font-weight: bold;}
```


Taille des polices [font-size]

Le corps d'une police se règle par la propriété font-size.

On peut choisir parmi beaucoup d'unités différentes (par exemple, pixels et pourcentages) pour décrire les corps des polices.

Exp :

```
h1 {font-size: 30px;}  
h2 {font-size: 12pt;}  
h3 {font-size: 120%;}  
p {font-size: 1em;}
```

Il existe une différence essentielle entre les quatre unités précédentes. Les unités « px » et « pt » font que le corps de la police est absolu, tandis que « % » et « em » permettent à l'utilisateur de l'ajuster si nécessaire. Il est souhaitable utiliser des unités ajustables telles que « % » ou « em ».

Concision [font]

Avec la propriété raccourcie font, il est possible de couvrir toutes les différentes propriétés de police en un coup.

Exp:

```
p { font-style: italic;  
font-weight: bold;  
font-size: 30px;  
font-family: arial, sans-serif; }
```

Avec la propriété raccourcie, on peut simplifier le code :

```
p { font: italic bold 30px arial, sans-serif; }
```

L'ordre des valeurs de la propriété font est le suivant :

```
font-style | font-variant | font-weight | font-size | font-family
```

Indentation du texte [text-indent]

La propriété **text-indent** permet d'ajouter une touche d'élégance aux paragraphes de texte en appliquant une indentation à la première ligne du paragraphe.

Exp:

```
p { text-indent: 30px; }
```

Alignement du texte [text-align]

La propriété CSS text-align correspond à l'attribut **align** utilisé dans le HTML. Le texte peut être aligné à gauche (**left**), à droite (**right**) ou centré (**center**). La valeur **justify** étirera chaque ligne de sorte que les marges à gauche et à droite soient droites toutes les deux.

Exp:

```
th { text-align: right; }  
td { text-align: center; }  
p { text-align: justify; }
```

Décoration du texte [text-decoration]

La propriété **text-decoration** permet d'ajouter des « effets » différents au texte. Par exemple, on peut souligner le texte, le barrer, ou tracer un trait au-dessus.

Exp:

```
h1 { text-decoration: underline; }  
h2 { text-decoration: overline; }  
h3 { text-decoration: line-through; }
```

Espacement des lettres [letter-spacing]

L'espacement entre les caractères du texte peut être défini avec la propriété letter-spacing.
La valeur de la propriété est celle de l'espacement désiré.

Exp:

```
h1 { letter-spacing: 6px; }  
p { letter-spacing: 3px; }
```

Transformation du texte [text-transform]

La propriété **text-transform** contrôle la capitalisation du texte. On peut choisir une capitalisation initiale, une mise en majuscules ou une mise en minuscules, indépendamment de l'aspect du texte original dans le code HTML.

Exp: le mot « titre » que l'on peut présenter à l'utilisateur comme « TITRE » ou « Titre ».

La propriété **text-transform** admet quatre valeurs possibles :

Capitalize: Elle capitalise la première lettre de chaque mot.

Uppercase: Elle convertit toutes les lettres en majuscules.

Lowercase: Elle convertit toutes les lettres en minuscules.

None: Aucune transformation n'est appliquée.

Exp:

```
h1 { text-transform: uppercase; }  
li { text-transform: capitalize; }
```

CSS 3

- **Personnalisation des liens**
- **Identification et le regroupement d'éléments**
- **Modèle des boîtes**
- **Balises structurantes de HTML5**

Personnalisation des liens

CSS permet de définir les propriétés du lien différemment, selon que le lien est **visité**, **non visité**, **activé**, ou si le **curseur le survole**. Cela permet d'ajouter des effets plaisants et utiles aux sites Web. Pour contrôler ces effets, on utilise ce qu'on appelle des pseudo-classes.

Une pseudo-classe permet de tenir compte de conditions et événements différents pour la définition d'une propriété sur une balise HTML.

Exp: les liens sont définis en HTML avec des balises `<a>`. On peut donc utiliser **a** comme sélecteur dans CSS :

```
a { color: blue; }
```

Personnalisation des liens

Un lien est susceptible d'avoir plusieurs états.

Exp: il peut être visité ou non. On peut utiliser des pseudo-classes pour assigner des styles différents aux liens visités et non visités.

```
a:link { color: blue; }  
a:visited { color: red; }
```

On Utilise **a:link** et **a:visited** pour les liens non visités et pour ceux visités. Les liens actifs ont pour pseudo-classe **a:active**, et la pseudo-classe **a:hover** intervient lorsque le curseur survole le lien.

Personnalisation des liens

La pseudo-classe **:link**

utilisée pour les liens menant aux pages que l'utilisateur n'a pas visitées.

Exp:

```
a:link { color: #6699CC; }
```

La pseudo-classe **:visited**

Utilisée pour les liens menant aux pages que l'utilisateur a visitées.

Exp:

```
a:visited { color: #660099; }
```

La pseudo-classe **:active**

La pseudo-classe :active est utilisée pour les liens qui sont activés.

Exp:

```
a:active { background-color: #FFFF00; }
```

La pseudo-classe **:hover**

Utilisée lorsque le pointeur de la souris survole un lien.

Exp:

```
a:hover { color: orange; font-style: italic; }
```

Personnalisation des liens

Exp 1: l'espacement entre les lettres

On peut ajuster l'espacement entre les lettres avec la propriété **letter-spacing**.

Cela s'applique aux liens pour un effet spécial :

```
a:hover { letter-spacing: 10px;  
          font-weight:bold;  
          color:red; }
```

Exp2: MAJUSCULES et minuscules

La propriété **text-transform** permet de basculer entre des lettres majuscules et minuscules. Elle peut aussi servir pour créer un effet sur les liens :

```
a:hover { text-transform: uppercase;  
          font-weight:bold;  
          color:blue;  
          background-color:yellow; }
```

Les deux exemples donnent un aperçu des possibilités de combinaison des différentes propriétés presque illimitées.

Personnalisation des liens: Supprimer le trait sous les liens

```
a { text-decoration:none; }
```

On peut fixer la valeur de **text-decoration** en même temps que celles d'autres propriétés pour les quatre pseudo-classes.

```
a:link { color: blue;  
         text-decoration:none; }  
a:visited { color: purple;  
            text-decoration:none; }  
a:active { background-color: yellow;  
           text-decoration:none; }  
a:hover { color:red;  
          text-decoration:none; }
```

Identification et le regroupement d'éléments (class et id)

Utilisation des attributs **class** et **id**.

Comment donner à un titre particulier une autre couleur que celle des autres titres du site Web ? Comment regrouper les liens en différentes catégories et donner à chacune un style spécial ? Solution: Le regroupement d'éléments dans une classe.

Exp :

```
<p>Pays africains:</p>
<ul>
  <li><a href="ma.htm">Maroc</a></li>
  <li><a href="al.htm">Algérie</a></li>
  <li><a href="tu.htm">Tunisie</a></li>
</ul>
<p>Pays européens:</p>
<ul>
  <li><a href="fr.htm">France</a></li>
  <li><a href="es.htm">Espagne</a></li>
  <li><a href="it.htm">Italie</a></li>
</ul>
```

Identification et le regroupement d'éléments : class

Nous voulons ensuite que les liens des pays africains soient jaune, que ceux des pays européens soient rouge, et que les autres liens de la page Web restent bleu.

Pour y parvenir, nous répartissons les liens en deux catégories. On assigne une classe à chaque lien avec l'attribut **class**.

```
<p>Pays africains:</p>
<ul>
  <li><a href="ma.htm" class="af">Maroc</a></li>
  <li><a href="al.htm" class="af">Algérie</a></li>
  <li><a href="tu.htm" class="af">Tunisie</a></li>
</ul>
<p>Pays européens:</p>
<ul>
  <li><a href="fr.htm" class="eu">France</a></li>
  <li><a href="es.htm" class="eu">Espagne</a></li>
  <li><a href="it.htm" class="eu">Italie</a></li>
</ul>
```

Fichier HTML

```
a { color: blue; }
a.af { color: #FFBB00; }
a.eu { color: #800000; }
```

Fichier CSS

Identification d'un élément avec id

on utilise l'attribut id pour identifier un élément unique parmi plusieurs éléments

On ne peut pas avoir deux éléments avec le même id dans le document.

Exp :

```
<h1>Chapitre 1</h1>
<h2>Chapitre 1.1</h2>
<h2>Chapitre 1.2</h2>
<h1>Chapitre 2</h1>
<h2>Chapitre 2.1</h2>
<h3>Chapitre 2.1.2</h3>
```

Il pourrait s'agir des titres d'un document partagé en chapitres ou en paragraphes. On assigne un **id** à chaque chapitre comme ceci :

```
<h1 id="c1">Chapitre 1</h1>
<h2 id="c1-1">Chapitre 1.1</h2>
<h2 id="c1-2">Chapitre 1.2</h2>
<h1 id="c2">Chapitre 2</h1>
<h2 id="c2-1">Chapitre 2.1</h2>
<h3 id="c2-1-2">Chapitre 2.1.2</h3>
```

Le titre du chapitre 1.2 apparaît en rouge:

```
#c1-2 { color: red; }
```


Regroupement d'éléments (span et div)

Les éléments `` et `<div>` servent à regrouper et structurer un document et seront souvent utilisés avec les attributs `class` et `id`.

Dans cette leçon, nous verrons de plus près l'utilisation des deux éléments ``

Le regroupement avec ``

L'élément `` est un élément neutre, qui n'ajoute rien au document. Mais CSS peut utiliser l'élément `` pour ajouter des caractéristiques visuelles à des parties spécifiques du texte des documents.

Exp:

```
<p>Tôt couché et tôt levé fortifie, enrichit et rend avisé.</p>
```

Regroupement d'éléments

Pour accentuer en rouge des parties de cette citation . Dans ce but, on utilise la balise . Chaque **span** reçoit un attribut **class** défini dans la feuille de style :

```
<p>Tôt couché et tôt levé  
<span class="benefit">fortifie</span>,  
<span class="benefit">enrichit</span>  
et <span class="benefit">rend avisé</span>.</p>
```

Fichier HTML

```
span.benefit { color:red; }
```

Fichier CSS

On peut utiliser l'attribut **id** pour ajouter du style aux éléments , tant que vous appliquez un id unique à chacun des trois éléments .

Regroupement d'éléments <div>

 s'utilise dans un élément de type bloc, <div> sert à regrouper un ou plusieurs éléments de type bloc.

Le regroupement avec <div> fonctionne plus ou moins de la même façon.

Exp: Deux listes de présidents des États-Unis répartis selon leur affiliation politique :

```
<div id="democrats">
<ul>    <li>Jimmy Carter</li>
        <li>Bill Clinton</li>
        <li>Barack Obama</li>
        <li>Joe Biden</li>
</ul>
</div>
<div id="republicans">
<ul>    <li>Ronald Reagan</li>
        <li>George Bush</li>
        <li>George W. Bush</li>
        <li>Donald Trump</li>
</ul>
</div>
```

```
#democrats { background:blue; }
#republicans { background:red; }
```

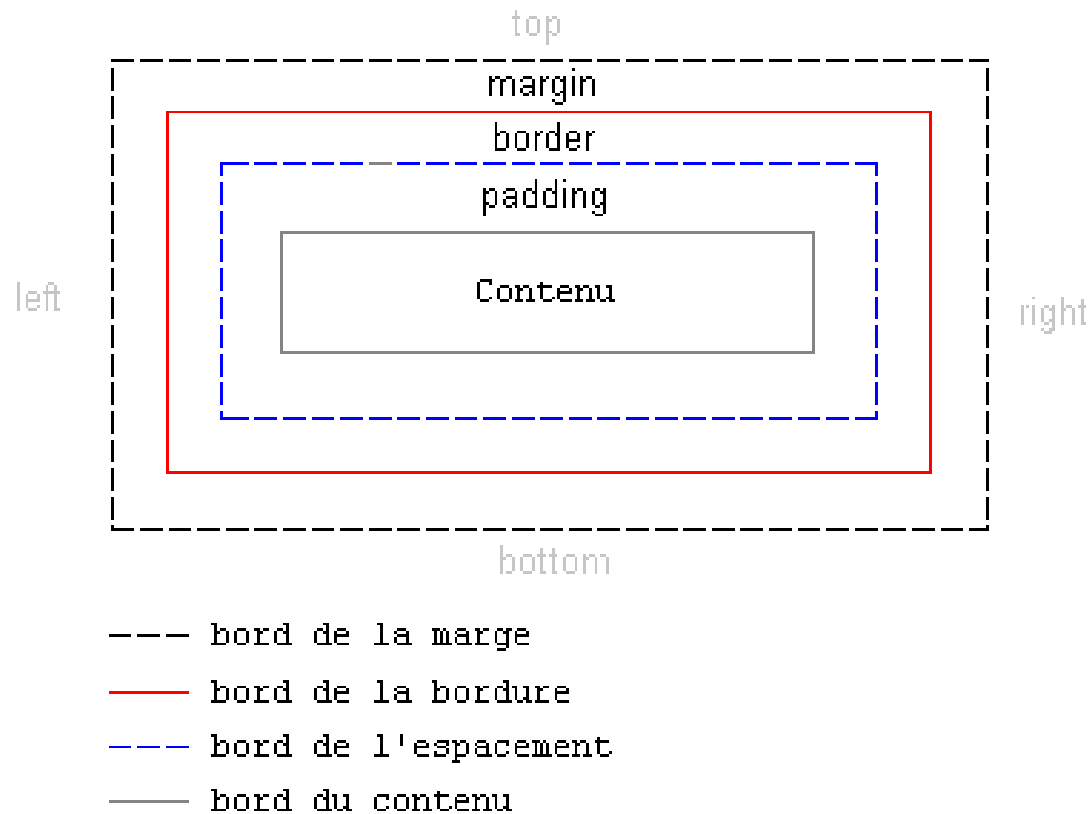
Fichier CSS

Fichier HTML

Modèle des boîtes

Le modèle des boîtes de CSS décrit les boîtes générées pour les éléments HTML. Il contient également des options détaillées pour l'ajustement des marges, des bordures, des espacements et du contenu de chaque élément.

Le modèle des boîtes dans CSS

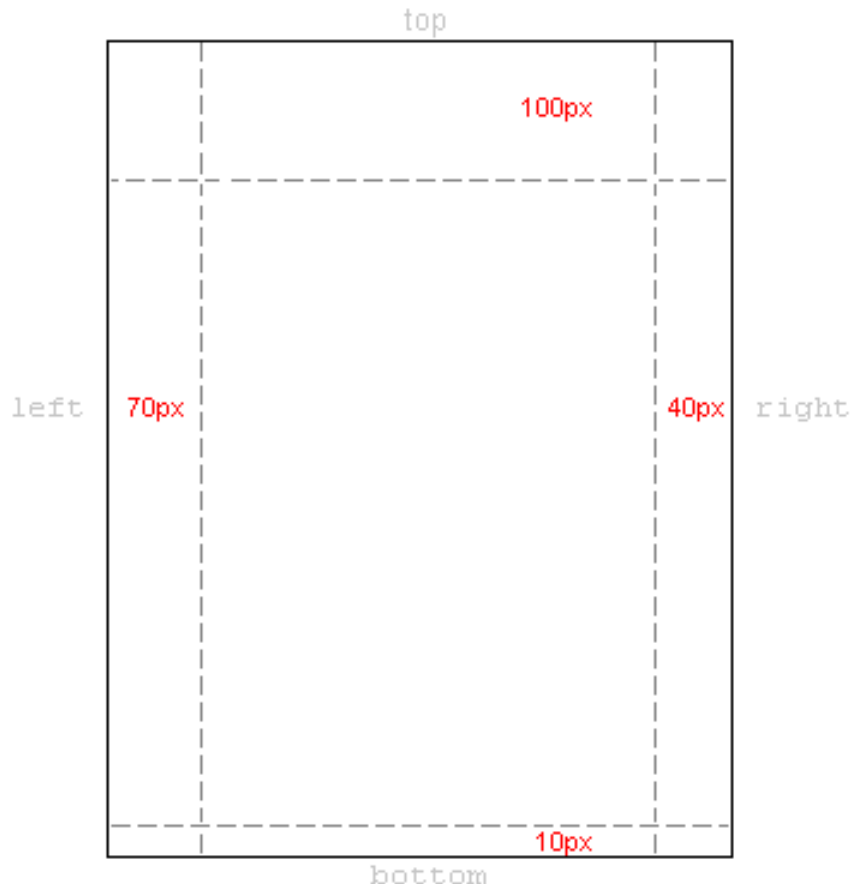


Margin et padding

Fixer la marge d'un élément

Un élément a quatre côtés : droit ("right"), gauche ("left"), supérieur ("top") et inférieur ("bottom"). La marge est la distance entre chaque côté et l'élément avoisinant (ou les bordures du document).

Exp: les marges de l'élément <body>.



```
body { margin-top: 100px;  
        margin-right: 40px;  
        margin-bottom: 10px;  
        margin-left: 70px; }
```

Ou bien :

```
body { margin: 100px 40px 10px 70px; }
```

Exp2: Ajout de marges dans le paragraphe

```
body { margin: 100px 40px 10px 70px; }  
p { margin: 5px 50px 5px 50px; }
```

Fixer l'espacement dans un élément

L'espacement peut aussi être assimilé à un « remplissage ». Cela peut se comprendre dans la mesure où l'espacement n'affecte pas la distance de l'élément aux autres éléments mais définit seulement la distance interne entre la bordure et le contenu de l'élément.

Exp:

```
h1 { background: yellow; }  
h2 { background: orange; }
```

En définissant l'espacement des titres, on change le remplissage autour du texte dans chaque titre :

```
h1 {   background: yellow;  
      padding: 20px 20px 20px 80px; }  
h2 {   background: orange;  
      padding-left: 120px; }
```

Types de bordures [border-style]



8 types de bordures sont possibles.

On peut utiliser les valeurs "none" ou "hidden" si on ne veut aucune bordure.

Couleur des bordures [border-color]

Définit la couleur d'une bordure.

Exemples:

"#123456"

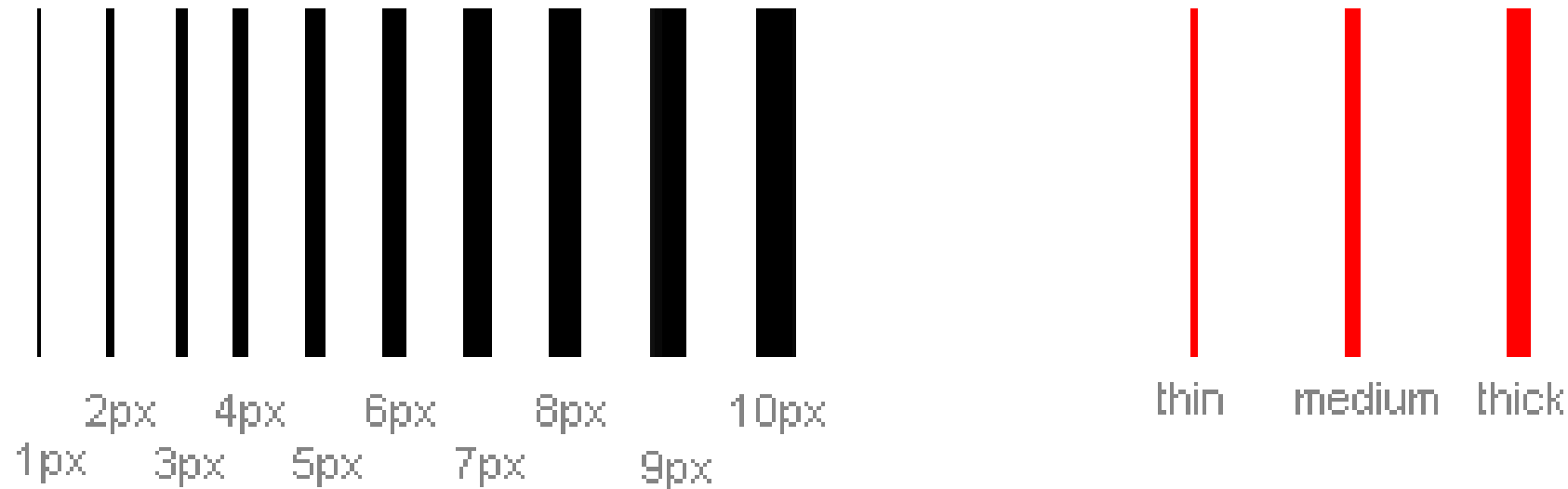
"rgb(123,123,123)"

"yellow"



Couleur des bordures [border-width]

Définie par la propriété border-width, dont les valeurs peuvent être **thin**, **medium** et **thick**, ou une valeur numérique en pixels.



Bordures [border]

Exp:

On définit les bordures différentes pour <h1>, <h2>, et <p>. Le résultat montre les possibilités :

```
h1 { border-width: thick;
      border-style: dotted;
      border-color: gold; }
h2 { border-width: 20px;
      border-style: outset;
      border-color: red; }
p { border-width: 1px;
     border-style: dashed;
     border-color: blue; }
ul { border-width: thin;
      border-style: solid;
      border-color: orange; }
```

Bordures [border]

Exp: Déclaration des propriétés spéciales pour les bordures supérieures, inférieures, gauches ou droites :

```
h1 {  
    border-top-width: thick;  
    border-top-style: solid;  
    border-top-color: red;  
  
    border-bottom-width: thick;  
    border-bottom-style: solid;  
    border-bottom-color: blue;  
  
    border-right-width: thick;  
    border-right-style: solid;  
    border-right-color: green;  
  
    border-left-width: thick;  
    border-left-style: solid;  
    border-left-color: orange; }
```

Bordures [border]

Concision [border]

On peut concentrer plusieurs propriétés en une seule avec border :

```
p {    border-width: 1px;  
      border-style: solid;  
      border-color: blue; }
```

Il peut se concentrer en :

```
p { border: 1px solid blue; }
```

Hauteur et largeur des éléments

Il est possible de définir la hauteur et la largeur d'un élément.

Fixer la largeur [width]

La propriété **width** permet de définir la largeur d'un élément.

Exp :

```
div.box { width: 200px;  
          border: 1px solid black;  
          background: orange; }
```

Fichier CSS

```
<body>  
<div class="box">Il était une fois , ... </div>  
</body>
```

Fichier HTML

Fixer la hauteur [height]

Dans l'exemple précédent, la hauteur de la boîte fixée par son contenu. On peut affecter la hauteur d'un élément avec la propriété **height**.

```
div.box { height: 500px; width: 200px;  
          border: 1px solid black;  
          background: orange; }
```

Éléments flottants

Un élément peut avoir une position flottante à droite ou à gauche avec la propriété **float**.

Exp:

```
<div id="picture">  
  
</div>  
<p>biographie ...</p>
```

Pour que l'image flotte à gauche et que le texte l'entoure, il suffit de définir la largeur de la boîte entourant l'image puis de donner à la propriété **float** la valeur "**left**" :

```
#picture { float:left;  
            width: 100px; }
```

Éléments flottants

Les flottants peuvent servir pour créer les colonnes d'un document.

Exp:

```
<div id="column1">
    <p>bla bla bla...</p>
</div>
<div id="column2">
    <p>bla bla bla...</p>
</div>
<div id="column3">
    <p>bla bla bla...</p>
</div>
```

Une fois fixée la largeur des colonnes (par exemple à 33%), on fait flotter chaque colonne à gauche en définissant la propriété **float** :

```
#column1 { float:left; width: 33%; }
#column2 { float:left; width: 33%; }
#column3 { float:left; width: 33%; }
```

La propriété **float** admet les valeurs "**left**", "**right**" ou "**none**".

La propriété clear

Sert à contrôler le comportement des éléments qui viennent après les éléments flottants dans le document. Par défaut, les éléments à suivre remontent pour combler l'espace disponible libéré lorsqu'une boîte flotte d'un côté.

La propriété **clear** admet les valeurs "**left**", "**right**", "**both**" ou "**none**". Le principe est tel que, par exemple, si la valeur de clear d'une boîte est fixée à "both", la bordure supérieure de la marge de cette boîte se trouvera toujours sous la bordure de la marge inférieure des boîtes flottantes éventuelles venant du dessus.

```
<div id="picture">  
  
</div>  
<h1>Said Youssef</h1>  
<p class="floatstop">biographie ...</p>
```

Pour empêcher que le texte ne flotte à côté de l'image, nous pouvons ajouter le code CSS suivant :

```
#picture { float:left; width: 100px; }  
.floatstop { clear:both; }
```


Balises structurantes de HTML5

Dans HTML5, des nouvelles balises sont introduites pour structurer les pages.

`<header>` : l'en-tête

La plupart des sites web possèdent un en-tête (header). On y trouve le plus souvent un logo, une bannière, ...

```
<header>
<!-- Placez ici le contenu de l'en-tête de votre page -->
</header>
```

`<footer>` : le pied de page

le pied de page (footer) se trouve en général tout en bas du document. On y trouve des informations comme des liens de contact, le nom de l'auteur, les mentions légales, etc.

```
<footer>
<!-- Placez ici le contenu du pied de page -->
</footer>
```

Principaux liens de navigation <nav>

La balise <nav> doit regrouper tous les principaux liens de navigation du site. Vous y placerez par exemple le menu principal de votre site.

Généralement, le menu est réalisé sous forme de liste à puces à l'intérieur de la balise

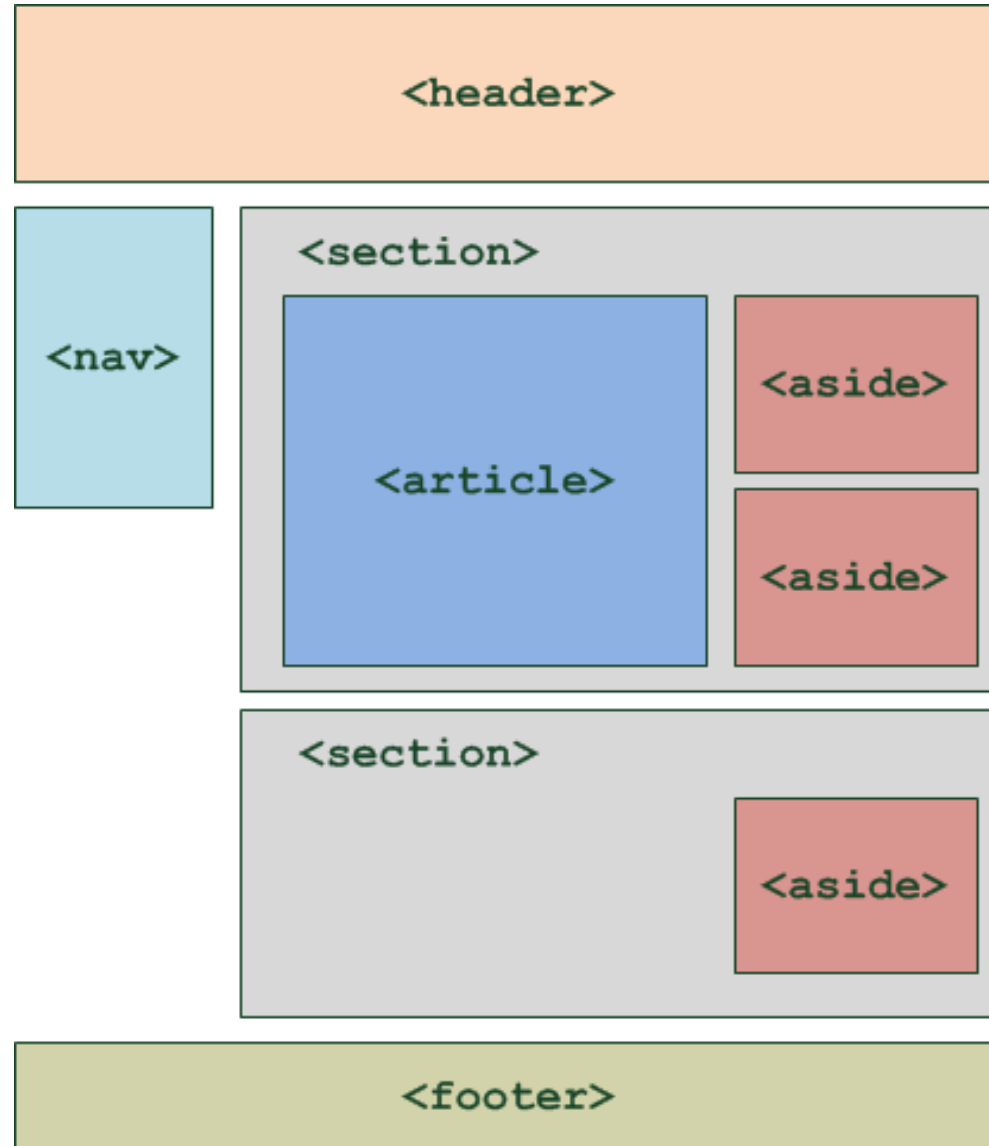
```
<nav>
<ul>
<li><a href="index.html">Accueil</a></li>
<li><a href="forum.html">Forum</a></li>
<li><a href="contact.html">Contact</a></li>
</ul>
</nav>
```

Article indépendant <article>

La balise <article> sert à englober une portion généralement autonome de la page. C'est une partie de la page qui pourrait ainsi être reprise sur un autre site. C'est le cas par exemple des actualités (articles de journaux ou de blogs).

```
<article>  
<h1>Mon article</h1>  
<p>Bla bla bla bla</p>  
</article>
```

Exemple d'organisation de la page



Exemple d'utilisation des balises structurantes

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8" />
<title>Fac Tetouan</title>
<link rel="stylesheet" href="style.css" />
</head>
<body>
<header>
<h1>Université Abdelmalek Essaadi</h1>
<h2>Faculté des Sciences</h2>
</header>
<nav>
<ul>
<li><a href="#">Accueil</a></li>
<li><a href="#">Licences</a></li>
<li><a href="#">DCESS</a></li>
</ul>
</nav>
```

```
<section>
<aside>
<h1>Conférence sur les énergies renouvelables</h1>
<p>Une conférence sur les énergies renouvelables est organisée en ligne</p>
</aside>
<article>
<h1>Ouverture d'une nouvelle formation</h1>
<p>La faculté des Sciences de Tétouan a ouvert une nouvelle formation sur les
technologies de pointe</p>
</article>
</section>
<footer>
<p>Copyright FS - Tous droits réservés<br />
<a href="#">contacter nous</a></p>
</footer>
</body>
</html>
```

Exemple d'utilisation des balises structurantes

```
nav
{
float: left; width: 150px;
border: 1px solid black;
}

section
{
margin-left: 170px; border: 1px solid blue;
}

footer
{
text-align: center;
border: 1px solid blue;
}
```

Intégration des équations mathématiques

Utilisation de la librairie MathJax

Intégration des équations mathématiques avec MathJax

[MathJax](#) est une librairie Javascript, compatible avec tous les navigateurs, simple et très évoluée pour le rendu des expressions mathématiques en HTML.

Le codage simple et le contenu des expressions devient ainsi aisément modifiable en cas d'erreur ou de besoin.

La librairie JavaScript MathJax est appelée avec la syntaxe :

```
<head>...  
<script async="true"  
src="https://cdn.jsdelivr.net/npm/mathjax@2/MathJax.js?config=AM\_CHTML">  
</script>  
...</head>
```

<https://www.mathjax.org>

Intégration des équations mathématiques avec MathJax

Syntaxe

```
<!DOCTYPE html>
<html lang="fr"/>
<head>
<meta charset="utf-8">
<script async="true" src="https://cdn.jsdelivr.net/npm/mathjax@2/MathJax.js?config=AM_CHTML"> </script>
</head>
<body>
```

```
<div class="cmath">
  `f(x)=x^2 + 2x + 1 = 0`
  <br />Sa fonction dérivée :<br />
  `f^(x)=2x + 2`
</div>
```

$$f(x) = x^2 + 2x + 1 = 0$$

Sa fonction dérivée :

$$f'(x) = 2x + 2$$

Intégration des équations mathématiques avec MathJax

Syntaxe (suite)

```
<div class="cmath">
`frac(df(x))(dx) =
  lim_(h->0)(f(x+h)-f(x))/h`
<br />
`f(t) = a_(0)/2 +
  sum_(n=1)^oo [a_(n) cos(nt) + b_(n) sin(nt)]`
<br />
`x^' = frac(-b -sqrt(Delta))(2a)`
</div>
```

$$\frac{df(x)}{dx} = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$$
$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} [a_n \cos(nt) + b_n \sin(nt)]$$
$$x' = \frac{-b - \sqrt{\Delta}}{2a}$$

Intégration des équations mathématiques avec MathJax

Utilisation des matrices

```
<div class="cmath">
```

```
`B = ( ( 3 , 0 , 3 ),  
      ( 0 , 2 , 4 ),  
      ( 0 , 2 , 5 ) )  
      +  
      ( ( 3 , 0 , 3 ),  
        ( 2 , 7 , 4 ),  
        ( 1 , 2 , 4 ) )`
```

```
<br />
```

```
`[bbrho] = (( 1/2 , 0 ),  
            ( 0 , 1/2 ))`
```

```
</div>
```

$$B = \begin{pmatrix} 3 & 0 & 3 \\ 0 & 2 & 4 \\ 0 & 2 & 5 \end{pmatrix} + \begin{pmatrix} 3 & 0 & 3 \\ 2 & 7 & 4 \\ 1 & 2 & 4 \end{pmatrix}$$
$$[\rho] = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix}$$

Intégration des équations mathématiques avec MathJax

Les expressions peuvent être aussi "in line" dans un paragraphe en utilisant des balises `span`.

```
<div class="cmath">  
<p>Une expression :  
<span> `sum_(i=1)^n i^3=((n(n+1))/2)^2`</span>  
au sein d'une phrase.</p>  
</div>
```

Une expression : $\sum_{i=1}^n i^3 = \left(\frac{n(n+1)}{2} \right)^2$ au sein d'une phrase.

Intégration des équations mathématiques avec MathJax

Centrer les équations sur le signe =

```
<div class="cmath">
`{
  ( f(x) ,=, frac(x+1)(x-2) ) ,
  ( f^(x) ,=, frac((x-2)-(x+1))((x-2)^2)),
  ( ,=, - frac(3)(x^2 - 4x + 4))
:}`
</div>
```

$$\begin{aligned} f(x) &= \frac{x+1}{x-2} \\ f'(x) &= \frac{(x-2) - (x+1)}{(x-2)^2} \\ &= -\frac{3}{x^2 - 4x + 4} \end{aligned}$$

Intégration des équations mathématiques avec MathJax

Numéroter Les équations

math.css

```
.numero {  
  position: absolute;  
  left: 600px;}
```

```
<div class="cmath">
```

```
`cos^2 \theta + sin^2 \theta = 1 class{numero}{(6)}`
```

```
<br />
```

```
`cos^2 \theta - sin^2 \theta = cos 2\theta class{numero}{(7)}`
```

```
<br />
```

```
`cos^2 \alpha + sin^2 \alpha = 1 class{numero}{(8)}`
```

```
</div>
```

$$\cos^2 \theta + \sin^2 \theta = 1$$

(6)

$$\cos^2 \theta - \sin^2 \theta = \cos 2\theta$$

(7)

$$\cos^2 \alpha + \sin^2 \alpha = 1$$

(8)

Intégration des équations mathématiques avec MathJax

Encadrer les équations

```
<!DOCTYPE html>
<html lang="fr"/>
<head>
<link rel="stylesheet" href="math.css">
<meta charset="utf-8">
<script async="true" src="https://cdn.jsdelivr.net/npm/mathjax@2/MathJax.js?config=AM_CHTML"> </script>
</head>
<body>
```

```
<div class="cmath">
`class{highlight}
  { e^x = lim_(n->oo) (1 + x/n)^n }`
</div>
```

math.css

```
.highlight {
border: 2px solid #DD4A68;
padding: 8px;
margin-right: 4px; }
```

$$e^x = \lim_{n \rightarrow \infty} \left(1 + \frac{x}{n}\right)^n$$

Ingénierie du Web Avancé (IWA)

Module 1 : Initiation web: HTML – CSS – PHP

Frameworks CSS



Bootstrap



tailwindcss



Pure.CSS



BULMA.io



Skeleton

A dead simple, responsive boilerplate.



Ant Design

Tachyons



Materialize

Frameworks CSS

Les frameworks CSS sont des feuilles de style CSS prêtes à l'emploi ou des bibliothèques CSS avec des designs web déjà codés qui offrent aux développeurs web un certain soulagement, car ils leur épargnent un certain travail de développement du code.

Ingénierie du Web Avancé (IWA)

Module 1 : Initiation web: HTML – CSS – PHP

Framework Bootstrap



Bootstrap - Qu'est-ce que Bootstrap

- Bootstrap est un framework gratuit pour un développement Web plus rapide et plus facile
- Bootstrap comprend des modèles de conception basés sur HTML et CSS pour la typographie, les formulaires, les boutons, les tableaux, la navigation, les images et bien d'autres, ainsi que des plugins JavaScript facultatifs
- Bootstrap vous donne également la possibilité de créer facilement des conceptions réactives

Bootstrap – sites web

https://www.youtube.com/watch?v=T-6dbrYUpXg&list=PLBOL5k415M_WJeZyalQeWrWtH3OHDHxCu (Fr)

https://www.youtube.com/watch?v=dKVX22GR7zQ&list=PL4cUxeGkcC9joIM91nLzd_qaH_AimmdAR (Eng)

<https://www.youtube.com/watch?v=A2ToRN09XLY&list=PLxbVBWjVdAEjom8KOV1c9aXMCz4IE0OZy> (Ar)

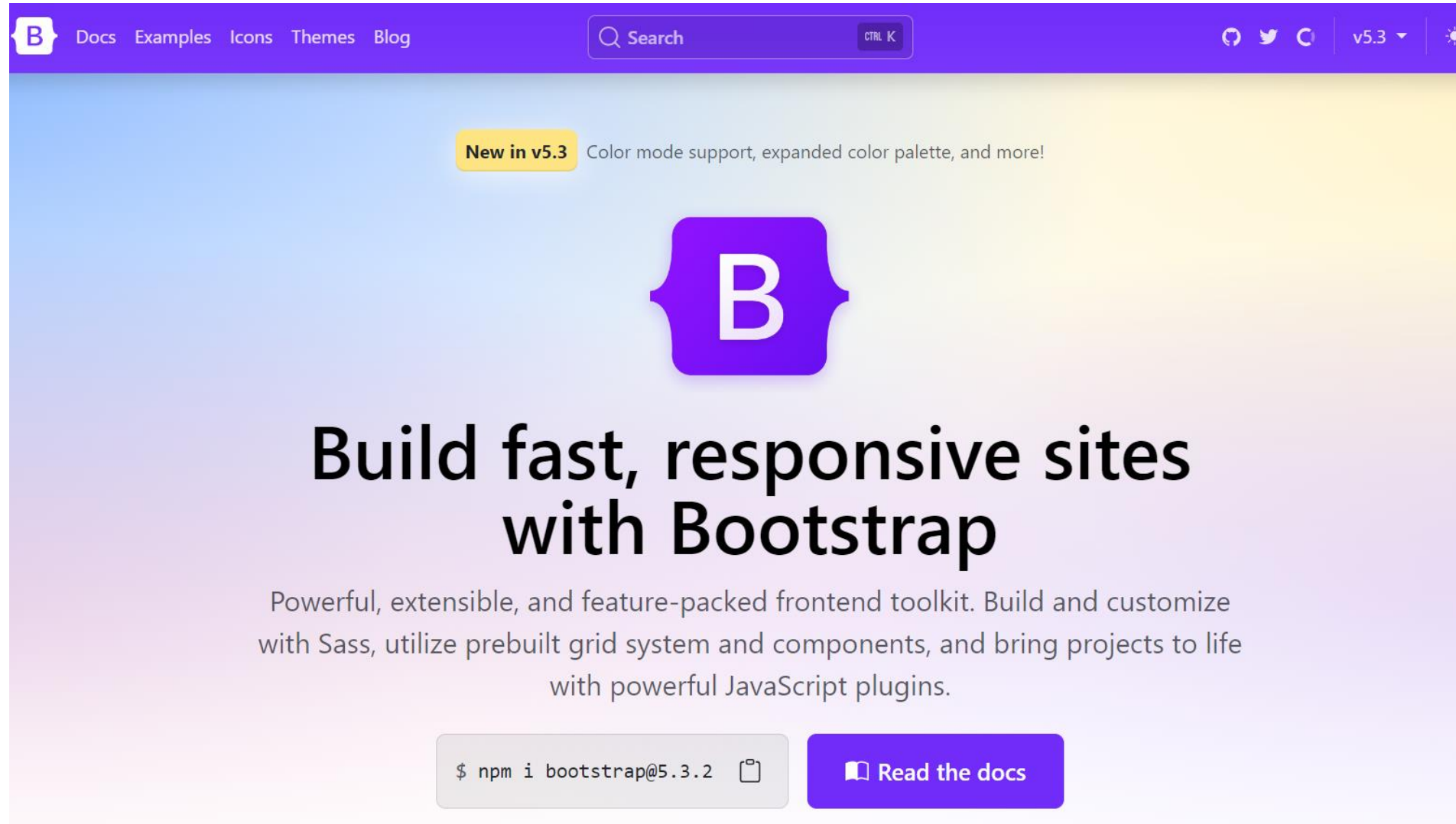
<https://openclassrooms.com/en/courses/7542506-creez-des-sites-web-responsives-avec-bootstrap-5>



<https://www.booglit.com/formation-gratuite-bootstrap-5>

Bootstrap –Site officiel

<https://getbootstrap.com>



Bootstrap –Site officiel - Documentation

<https://getbootstrap.com/docs/5.3/getting-started/introduction/>

B

Docs

Examples

Icons

Themes

Blog

Q

Search

CTRL K

v5.3

Getting started

Introduction

Download

Contents

Browsers & devices

JavaScript

Webpack

Parcel

Vite

Accessibility

RFS

RTL

Contribute

Customize

Overview

Sass

Options

Color


Color modes

Components

Get started with Bootstrap

View on GitHub

Bootstrap is a powerful, feature-packed frontend toolkit. Build anything—from prototype to production—in minutes.



\$350K+ in prizes and expert-run workshops. Register to help build the future of Web3.

ads via Carbon

Quick start

Get started by including Bootstrap's production-ready CSS and JavaScript via CDN without the need for any build steps. See it in practice with this [Bootstrap CodePen demo](#).

1. Create a new **index.html** file in your project root. Include the `<meta name="viewport">` tag as well for [proper responsive behavior](#) in mobile devices.

<!doctype html>

On this page

Quick start

CDN links

Next steps

JS components

Important globals

HTML5 doctype

Viewport meta

Box-sizing

Reboot

Community

جامعة عبد المومن
IWA
Ingénierie du Web Avancé
Bac+5

Module 1 : Initiation web: HTML – CSS – PHP / par Prof. Youssef Zaz

Include Bootstrap's CSS and JS

Nouveau fichier sur VSC, puis !, ligne CSS de bootstrap puis ligne JS avant </body>

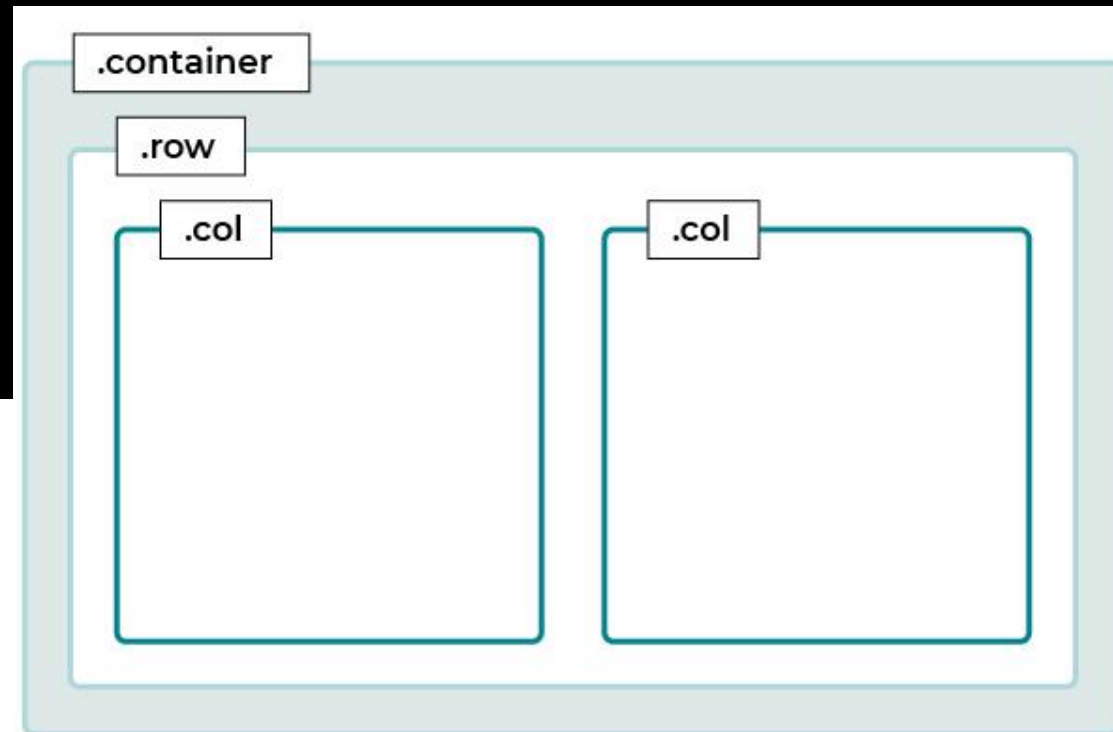
```
<!doctype html>
<html lang="fr">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Bootstrap demo</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
T3c6CoIi6uLrA9TneNEoa7RxnatzjcDSCmG1MXxSR1GAsXEV/Dwwykc2MPK8M2HN" crossorigin="anonymous">
  </head>
  <body>
    <h1>Hello, world!</h1>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"
integrity="sha384-C6RzsynM9kWDrmNeT87bh950GNyZPhcTNXj1NW7RuBCsyN/o0jlpcV8Qyq46cDfL"
crossorigin="anonymous"></script>
  </body>
</html>
```

Bootstrap – class container et container-fluid

```
<body>
  <div class="container-fluid">
    <h1>Hello, world!</h1>
    <p>Lorem, ipsum dolor sit amet consectetur adipisicing elit. Commodi doloremque
cumque repellat laborum sit modi optio iusto omnis minima, laboriosam, harum deleniti
culpa eaque officiis consequuntur dolore magnam expedita iste! Lorem ipsum dolor sit
amet consectetur adipisicing elit. Ab, quia provident? Qui odit molestiae obcaecati
quos? Adipisci qui porro perspiciatis beatae! Veritatis sit modi pariatur iusto
commodi harum provident mollitia.</p>
  </div>
```


Bootstrap - grid

```
<body>
  <div class="container">
    <div class="row">      <!--pour ajouter une ligne à la grille-->
      <div class="col">    <!--pour ajouter une colonne à la grille-->
        Première colonne
      </div>
      <div class="col">
        2ème colonne
      </div>
    </div>
  </div>
```



Bootstrap - grid

Créer 12 colonnes avec l'assistant de VSC: **div.col{\$}*12**

```
<body>
  <div class="container">
    <div class="row">
      <div class="col">1</div>
      <div class="col">2</div>
      <div class="col">3</div>
      <div class="col">4</div>
      <div class="col">5</div>
      <div class="col">6</div>
      <div class="col">7</div>
      <div class="col">8</div>
      <div class="col">9</div>
      <div class="col">10</div>
      <div class="col">11</div>
      <div class="col">12</div>
    </div>
  </div>
```

1

2

3

4

5

6

7

8

9

10

11

12

Bootstrap - grid

```
<div class="container">
  <div class="row">
    <div class="col border">1</div>
    <div class="col border">2</div>
    <div class="col border">3</div>
    <div class="col border">4</div>
    <div class="col border">5</div>
    <div class="col border">6</div>
    <div class="col border">7</div>
    <div class="col border">8</div>
    <div class="col border">9</div>
    <div class="col border">10</div>
    <div class="col border">11</div>
    <div class="col border">12</div>
  </div>
  <div class="row">
    <div class="col-4 border">A</div>
    <div class="col-8 border">B</div>
  </div>
</div>
```

1	2	3	4	5	6	7	8	9	10	11	12
A				B							

Bootstrap – grille responsive

```
<div class="container">
  <div class="row">
    <div class="col">
      <h2>Colonne 1</h2>
      <p>Lorem ipsum dolor sit amet co
adipisicing elit. Illum </p>
    </div>
    <div class="col">
      <h2>Colonne 2</h2>
      <p>Lorem ipsum dolor sit amet consectetur
adipisicing elit. Illum nostrum </p>
    </div>
    <div class="col">
      <h2>Colonne 3</h2>
      <p>Lorem ipsum dolor sit amet consectetur
adipisicing elit. Illum nostrum </p>
    </div>
    <div class="col">
      <h2>Colonne 4</h2>
      <p>Lorem ipsum dolor sit amet consectetur
adipisicing elit. Illum </p>
    </div>
  </div>
</div>
```

Colonne 1

Lorem ipsum dolor sit
amet consectetur
adipisicing elit. Illum
nostrum ullam iure
reiciendis ipsa
pariatur non? Omnis

Colonne 2

Lorem ipsum dolor sit
amet consectetur
adipisicing elit. Illum
nostrum ullam iure
reiciendis ipsa
pariatur non? Omnis

Colonne 3

Lorem ipsum dolor sit
amet consectetur
adipisicing elit. Illum
nostrum ullam iure
reiciendis ipsa
pariatur non? Omnis

Colonne 4

Lorem ipsum dolor sit
amet consectetur
adipisicing elit. Illum
nostrum ullam iure
reiciendis ipsa
pariatur non? Omnis

Bootstrap – grille responsive

Bootstrap>Docs>Layout>Breakpoint

```
<div class="row">
  <div class="col-12 col-md-6 col-lg-3">
    <h2>Colonne 1</h2>
    <p>Lorem ipsum dolor sit amet consectetur
    adipisicing elit.</p></div>
  <div class="col-md-6 col-lg-3">
    <h2>Colonne 2</h2>
    <p>Lorem ipsum dolor sit amet consectetur
    adipisicing elit.</p></div>
  <div class="col-md-6 col-lg-3">
    <h2>Colonne 3</h2>
    <p>Lorem ipsum dolor sit amet consectetur
    adipisicing elit.</p></div>
  <div class="col-md-6 col-lg-3">
    <h2>Colonne 4</h2>
    <p>Lorem ipsum dolor sit amet
    consectetur adipisicing elit.</p></div>
</div>
```

Breakpoint	Class infix	Dimensions
Extra small	<i>None</i>	<576px
Small	sm	≥576px
Medium	md	≥768px
Large	lg	≥992px
Extra large	xl	≥1200px
Extra extra large	xxl	≥1400px

Bootstrap – grille responsive

Sur ecran d'ordinateur

Colonne 1

Lorem ipsum dolor sit amet
consectetur adipisicing elit.

Colonne 2

Lorem ipsum dolor sit amet
consectetur adipisicing elit.

Colonne 3

Lorem ipsum dolor sit amet
consectetur adipisicing elit.

Colonne 4

Lorem ipsum dolor sit amet
consectetur adipisicing elit.

Sur ecran de tablette

Colonne 1

Lorem ipsum dolor sit amet consectetur
adipiscing elit.

Colonne 3

Lorem ipsum dolor sit amet consectetur
adipiscing elit.

Colonne 2

Lorem ipsum dolor sit amet consectetur
adipiscing elit.

Colonne 4

Lorem ipsum dolor sit amet consectetur
adipiscing elit.

Sur ecran de smartphone

Colonne 1

Lorem ipsum dolor sit amet consectetur adipisicing elit.

Colonne 2

Lorem ipsum dolor sit amet consectetur adipisicing elit.

Colonne 3

Lorem ipsum dolor sit amet consectetur adipisicing elit.

Colonne 4

Lorem ipsum dolor sit amet consectetur adipisicing elit.

Bootstrap – Texte/Typographie

On utilise h1 ... h6 classes sur les éléments pour les faire se comporter comme des en-têtes :

```
<p class="h1">h1 Bootstrap Titre 1</p>  
<p class="h2">h2 Bootstrap Titre 2</p>  
<p class="h3">h3 Bootstrap Titre 3</p>  
<p class="h4">h4 Bootstrap Titre 4</p>  
<p class="h5">h5 Bootstrap Titre 5</p>  
<p class="h6">h6 Bootstrap Titre 6</p>
```

h1 Bootstrap Titre 1

h2 Bootstrap Titre 2

h3 Bootstrap Titre 3

h4 Bootstrap Titre 4

h5 Bootstrap Titre 5

h6 Bootstrap Titre 6

Bootstrap – Couleurs du texte

Les classes pour les couleurs du texte sont : `.text-muted`, `.text-primary`, `.text-success`, `.text-info`, `.text-warning`, `.text-danger`, `.text-secondary`, `.text-white`, `.text-dark`, `.text-body` (couleur du corps par défaut/souvent noir) et `.text-light`:

```
<div class="conteneur mt-3">
  <h2>Couleurs contextuelles</h2>
  <p class="text-muted">texte est coupé.</p>
  <p class="text-primary">texte est important.</p>
  <p class="text-success">texte indique le succès.</p>
  <p class="text-info">texte d'information.</p>
  <p class="text-warning">texte d'avertissement.</p>
  <p class="text-danger">texte de danger.</p>
  <p class="text-secondary">Texte secondaire.</p>
  <p class="text-dark">texte est gris foncé.</p>
  <p class="text-body">Couleur par défaut.</p>
</div>
```

texte est coupé.

texte est important.

texte indique le succès.

texte d'information.

texte d'avertissement.

texte de danger.

Texte secondaire.

texte est gris foncé.

Couleur par défaut.

Bootstrap – Couleurs d'arrière-plan

Les classes pour les couleurs d'arrière-plan sont : `.bg-primary`, `.bg-success`, `.bg-info`, `.bg-warning`, `.bg-danger`, `.bg-secondary`, `.bg-dark` et `.bg-light`.

```
<div class="conteneur mt-3">
  <h2>Arrière-plans contextuels</h2>
  <p class="bg-primary text-white">texte est important.</p>
  <p class="bg-success text-white">texte de succès.</p>
  <p class="bg-info text-white">texte d'information.</p>
  <p class="bg-warning text-white">texte d'avertissement.</p>
  <p class="bg-danger text-white">texte de danger.</p>
  <p class="bg-Secondary text-white">fond secondaire.</p>
  <p class="bg-dark text-white">fond gris foncé.</p>
  <p class="bg-light text-dark">fond gris clair.</p>
</div>
```

texte est important.

texte de succès.

texte d'information.

texte d'avertissement.

texte de danger.

fond gris foncé.

fond gris clair.

Bootstrap – Tableaux

La classe table ajoute un style de base à une table :

```
<table class="table">
  <thead>
    <tr><th>Firstname</th><th>Lastname</th><th>Age</th></tr>
  </thead>
  <tbody>
    <tr><td>Mohamed</td><td>KINANI</td><td>25</td></tr>
    <tr><td>Laila</td><td>SAFIR</td><td>23</td></tr>
    <tr><td>Sami</td><td>MANSOURI</td><td>24</td></tr>
  </tbody>
</table>
```

Firstname	Lastname	Age
Mohamed	KINANI	25
Laila	SAFIR	23
Sami	MANSOURI	24

Bootstrap – Les tableaux - Lignes rayées

```
<table class="table table-striped">
  <thead>
    <tr><th>Firstname</th><th>Lastname</th><th>Age</th></tr>
  </thead>
  <tbody>
    <tr><td>Mohamed</td><td>KINANI</td><td>25</td></tr>
    <tr><td>Laila</td><td>SAFIR</td><td>23</td></tr>
    <tr><td>Sami</td><td>MANSOURI</td><td>24</td></tr>
  </tbody>
</table>
```

Firstname	Lastname	Age
Mohamed	KINANI	25
Laila	SAFIR	23
Sami	MANSOURI	24

Bootstrap – Les tableaux - classes

Classe	La description
<code>table-striped</code>	Tableau rayée
<code>table table-dark</code>	table sombre
<code>table-borderless</code>	supprime les bordures du tableau
<code>table-hover</code>	ajoute un effet de survol (couleur de fond gris) sur les lignes du tableau
<code>table-bordered</code>	Ajoute les bordures

Bootstrap – Les tableaux - classes contextuelles

Les classes contextuelles peuvent être utilisées pour colorer l'ensemble du tableau (`<table>`), les lignes du tableau (`<tr>`) ou les cellules du tableau (`<td>`).

Classe	La description
.table-primary	Bleu : indique une action importante
.table-success	Vert : indique une action réussie ou positive
.table-danger	Rouge : indique une action dangereuse ou potentiellement négative
.table-info	Bleu clair : indique une action ou un changement informatif neutre
.table-warning	Orange : indique un avertissement qui peut nécessiter votre attention
.table-active	Gris : applique la couleur de survol à la ligne ou à la cellule du tableau
.table-secondary	Gris : Indique une action un peu moins importante
.table-light	Fond de table ou de rangée de table gris clair
.table-dark	Fond de table ou de rangée de table gris foncé

Bootstrap – Tableaux réactifs

La classe `.table-responsive` ajoute une barre de défilement au tableau en cas de besoin (lorsqu'elle est trop grande horizontalement) :

```
<div class="table-responsive">
  <table class="table">
    ...
  </table>
</div>
```

Vous pouvez également décider quand le tableau doit avoir une barre de défilement, en fonction de la largeur de l'écran :

Classe	Largeur de l'écran
<code>.table-responsive-sm</code>	< 576 pixels
<code>.table-responsive-md</code>	< 768 pixels
<code>.table-responsive-lg</code>	< 992 pixels
<code>.table-responsive-xl</code>	< 1200px
<code>.table-responsive-xxl</code>	< 1400px

Bootstrap – Images

Coins arrondis

```

```



Cercle

```

```



La vignette

```

```



Bootstrap – Alignement des images

Centrer une image en ajoutant les classes utilitaires `.mx-auto`(margin:auto) à l'image

Faire flotter une image à gauche avec la classe *float-start* ou à droite avec *float-end* ou au centre *mx-auto* :

```

```



Alignement à gauche

```

```



Image centrée

```

```



Alignement à droite

Les images réactives s'ajustent automatiquement pour s'adapter à la taille de l'écran.

```

```



Bootstrap –Formulaires responsives

```
<div class="container py-3">
  <form action="/action_page.php" class="row g-3">
    <div class="col-md-6">
      <label for="nom" class="form-label">Nom:</label>
      <input type="text" class="form-control" id="nom" name="nom">
    </div>
    <div class="col-md-6">
      <label for="email" class="form-label">Email:</label>
      <input type="email" class="form-control" id="email" name="email">
    </div>
    <div class="col-md-12">
      <label for="Adresse" class="form-label">Adresse:</label>
      <input type="text" class="form-control" id="adresse" name="adresse">
    </div>
    <div class="col-md-4">
      <label for="cp" class="form-label">Code postal:</label>
      <input type="text" class="form-control" id="cp" name="cp">
    </div>
    <div class="col-md-8">
      <label for="ville" class="form-label">Ville:</label>
      <input type="text" class="form-control" id="ville" name="ville">
    </div>
    <div class="col-md-12">
      <button type="submit" class="btn btn-primary">Submit</button>
    </div>
  </form>
</div></div>
```

py-3	padding y 3
g-3	Gutter 3
md-6	Breakpoint Medium (6 colonnes)

Nom:

Email:

Adresse:

Code postal:

Ville:

Submit

Bootstrap – Les formulaires - Les boutons

Styles de boutons

Bootstrap 5 propose différents styles de boutons :

Basic

Primary

Secondary

Success

Info

Warning

Danger

Dark

Light

[Link](#)

```
<button type="button" class="btn">Basic</button>
<button type="button" class="btn btn-primary">Primary</button>
<button type="button" class="btn btn-secondary">Secondary</button>
<button type="button" class="btn btn-success">Success</button>
<button type="button" class="btn btn-info">Info</button>
<button type="button" class="btn btn-warning">Warning</button>
<button type="button" class="btn btn-danger">Danger</button>
<button type="button" class="btn btn-dark">Dark</button>
<button type="button" class="btn btn-light">Light</button>
<button type="button" class="btn btn-link">Link</button>
```

Les classes de boutons peuvent être utilisées sur les éléments `<a>`, `<button>`, ou `<input>`:

Bootstrap – Les boutons

Contour du bouton

Bootstrap 5 fournit également huit boutons contours/bordés.



```
<button type="button" class="btn btn-outline-primary">Primary</button>
<button type="button" class="btn btn-outline-secondary">Secondary</button>
<button type="button" class="btn btn-outline-success">Success</button>
<button type="button" class="btn btn-outline-info">Info</button>
<button type="button" class="btn btn-outline-warning">Warning</button>
<button type="button" class="btn btn-outline-danger">Danger</button>
<button type="button" class="btn btn-outline-dark">Dark</button>
<button type="button" class="btn btn-outline-light text-dark">Light</button>
```

Bootstrap – Les boutons

```
<div class="btn-group-vertical" role="group" aria-label="Vertical button group">  
<button onclick="window.location.href='https://w3docs.com';" type="button" class="btn btn-primary">Button 1</button>  
<button type="button" class="btn btn-primary">Button 2</button>  
<button type="button" class="btn btn-primary">Button 3</button>  
<button type="button" class="btn btn-primary">Button 4</button>  
</div>
```

Javascript (onclick) pour ajouter un hyperlien

Button 1

Button 2

Button 3

Button 4

Bootstrap – Barre de navigation de base

Une barre de navigation standard est créée avec la classe `.navbar`, suivie d'une classe de `.navbar-expand-xxl|xl|lg|md|sm` repli réactive : (empile la barre de navigation verticalement sur des écrans xlarge, extra large, large, moyen ou petit).

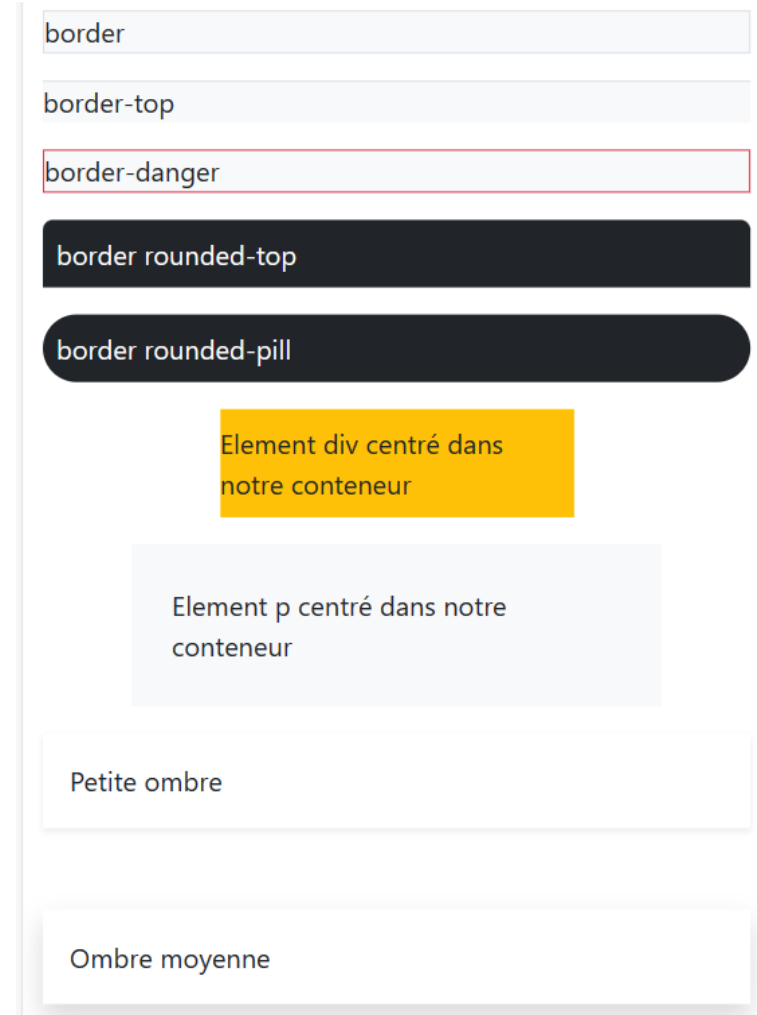
Pour ajouter des liens dans la barre de navigation, utilisez soit un ``élément (ou un `<div>`) avec `class="navbar-nav"`. Ajoutez ensuite des éléments `` avec une classe `.nav-item` suivis d'un élément `<a>` avec une classe `.nav-link` :

```
<div class="container py-3">
  <nav class="nav navbar navbar-expand-sm bg-primary navbar-dark">
    <div class="container-fluid">
      <ul class="navbar-nav">
        <li class="nav-item"><a class="nav-link" href="#">Link 1</a></li>
        <li class="nav-item"><a class="nav-link" href="#">Link 2</a></li>
        <li class="nav-item"><a class="nav-link" href="#">Link 3</a></li>
        <li class="nav-item"><a class="nav-link disabled" href="#">Disabled</a></li>
      </ul>
    </div>
  </nav>
</div>
```

Link 1 Link 2 Link 3 Disabled

Bootstrap: padding(p), margin (m)- border-shadow

```
<p class="bg-light border mt-2">border</p>
<p class="bg-light border-top">border-top</p>
<p class="bg-light border border-danger">border-
danger</p>
<p class="text-white bg-dark p-2 rounded-
top">border rounded-top</p>
<p class="text-white bg-dark p-2 rounded-
pill">border rounded-pill</p>
<div class="mx-auto bg-warning w-50">
<p class="py-2">Element div centré dans notre
conteneur</p>
</div>
<p class="mx-auto p-4 bg-light w-75">Element p
centré dans notre conteneur</p>
<div class="shadow-sm p-3 mb-5 bg-white">Petite
ombre</div>
<div class="shadow p-3 mb-5 bg-white">Ombre
moyenne</div>
```



Bootstrap – Alertes

Les alertes sont créées avec la classe `.alert`, suivie de l'une des classes contextuelles `.alert-success`, `.alert-info`, `.alert-warning`, `.alert-danger`, `.alert-primary`, `.alert-secondary`, `.alert-light` ou `.alert-dark`:

```
<div class="alert alert-success">  
    <strong> Succès!</strong>Indique une action réussie ou positive.  
</div>
```

Succès! Cette boîte d'alerte peut indiquer une action réussie ou positive.

Info! Cette boîte d'alerte pourrait indiquer un changement ou une action informatif neutre.

Avertissement! Cette boîte d'alerte peut indiquer un avertissement qui peut nécessiter votre attention.

Danger! Cette boîte d'alerte peut indiquer un danger ou une action qui nécessite votre attention.

Bootstrap – Liens d'alerte

Ajoutez la classe `.alert-link` à tous les liens à l'intérieur de la boîte d'alerte pour créer des « liens de couleur assortie » :

Succès! Vous devriez [lire ce message](#) .

Info! Vous devriez [lire ce message](#) .

Avertissement! Vous devriez [lire ce message](#) .

```
<div class="alert alert-success">
  <strong> Succès!</strong> Vous deviez<a href="#" class="alert-link">
lire ce message</a>.
</div>
```


Bootstrap – Alertes de fermeture

Pour fermer le message d'alerte, ajoutez une classe `.alert-dismissible` au conteneur d'alerte. Ajoutez ensuite `class="btn-close"` et `data-bs-dismiss="alert"` à un lien ou à un élément de bouton (lorsque vous cliquez dessus, la boîte d'alerte disparaîtra).

Succès! Cette boîte d'alerte peut indiquer une action réussie ou positive.



```
<div class="alert alert-success alert-dismissible">
  <button type="button" class="btn-close" data-bs-dismiss="alert"></button>
  <strong>Succès!</strong> Cette boîte d'alerte peut indiquer une action
  réussie ou positive.
</div>
```

Alertes animées

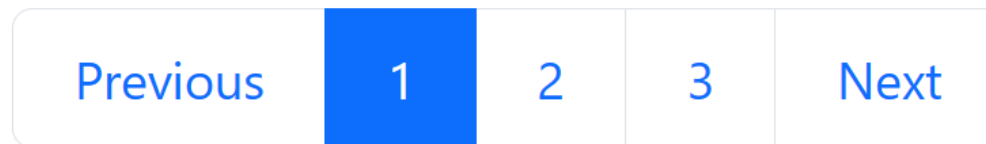
Les classes `.fade` et `.show` ajoutent un effet de décoloration lors de la fermeture du message d'alerte :

```
<div class="alert alert-danger alert-dismissible fade show">
```

Bootstrap – Dimensionnement de la pagination - État actif

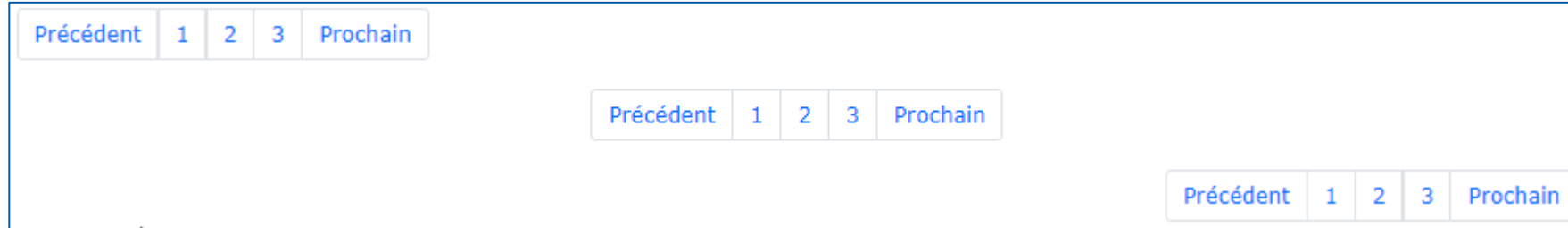
Les blocs de pagination peuvent également être dimensionnés à une taille plus grande ou plus petite :

```
<ul class="pagination pagination-lg">
  <li class="page-item"><a class="page-link" href="#">Previous</a></li>
  <li class="page-item active"><a class="page-link" href="#">1</a></li>
  <li class="page-item"><a class="page-link" href="#">2</a></li>
  <li class="page-item"><a class="page-link" href="#">3</a></li>
  <li class="page-item"><a class="page-link" href="#">Next</a></li>
</ul>
<ul class="pagination pagination-sm">
  <li class="page-item"><a class="page-link" href="#">Previous</a></li>
  <li class="page-item"><a class="page-link" href="#">1</a></li>
  <li class="page-item"><a class="page-link" href="#">2</a></li>
  <li class="page-item active"><a class="page-link" href="#">3</a></li>
  <li class="page-item"><a class="page-link" href="#">Next</a></li>
</ul>
```



Bootstrap – Alignement de la pagination

Utilisez des classes utilitaires pour modifier l'alignement de la pagination :



```
<!-- Default (left-aligned) -->
<ul class="pagination" style="margin:20px 0">
  <li class="page-item">...</li>
</ul>

<!-- Center-aligned -->
<ul class="pagination justify-content-center" style="margin:20px 0">
<li class="page-item">...</li></ul>

<!-- Right-aligned -->
<ul class="pagination justify-content-end" style="margin:20px 0">
<li class="page-item">...</li>
</ul>
```

Bootstrap – Pagination - Breadcrumbs

Une autre forme de pagination est le fil d'Ariane :

Les classes .breadcrumb et .breadcrumb-item indiquent l'emplacement de la page actuelle dans une hiérarchie de navigation :

```
<ul class="breadcrumb">
  <li class="breadcrumb-item"><a href="#">Photos</a></li>
  <li class="breadcrumb-item"><a href="#">Summer 2023</a></li>
  <li class="breadcrumb-item"><a href="#">Morocco</a></li><li class="breadcrumb-item
active">Agadir</li>
</ul>
```

[Photos](#) / [Summer 2023](#) / [Morocco](#) / Agadir

Bootstrap – Les Badges

Utilisez la classe `.badge` avec une classe contextuelle (comme `.bg-secondary`) dans les éléments `` pour créer des badges rectangulaires. Notez que les badges s'adaptent à la taille de l'élément parent (le cas échéant) :

Badges contextuels

Utilisez l'une des classes contextuelles (`.bg-*`) pour changer la couleur d'un badge :

```
<h1>Exemple<span class="badge bg-secondary">Nouveau</span></h1>

<span class="badge bg-primary">Primary</span>
<span class="badge bg-secondary">Secondary</span>
<span class="badge bg-success">Success</span>
<span class="badge bg-danger">Danger</span>
<span class="badge bg-warning">Warning</span>
<span class="badge bg-info">Info</span>
<span class="badge bg-light">Light</span>
<span class="badge bg-dark">Dark</span>
```

Exemple **Nouveau**

Primary

Secondary

Success

Danger

Warning

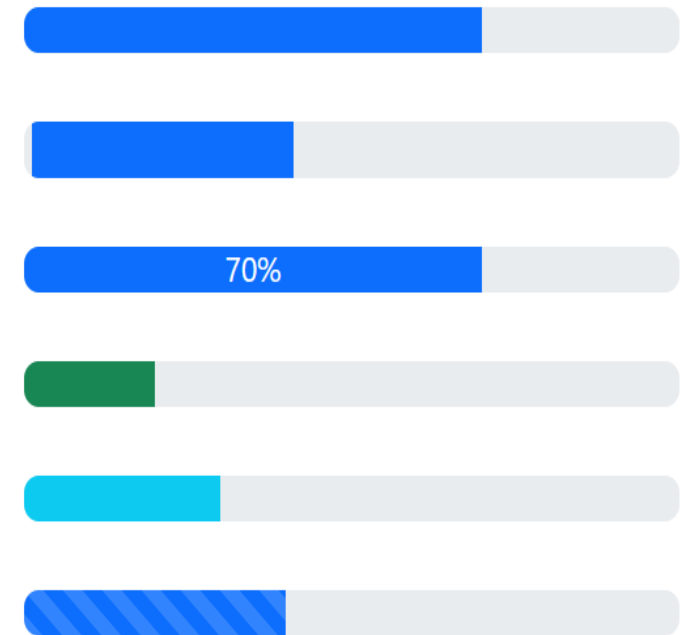
Info

Light

Dark

Bootstrap – Barre de progression

```
<div class="progress"><div class="progress-bar" style="width:70%">
</div></div><br />
<div class="progress" style="height:20px">
<div class="progress-bar" style="width:40%;height:20px">
  </div> </div><br />
<div class="progress"><div class="progress-bar"
  style="width:70%">70%</div> </div><br />
<!-- Green -->
<div class="progress"><div class="progress-bar bg-success"
style="width:20%"></div></div><br />
<!-- Turquoise -->
<div class="progress"><div class="progress-bar bg-info"
style="width:30%"></div></div> <br />
<div class="progress"><div class="progress-bar progress-bar-striped"
style="width:40%"></div> </div>
```



Bootstrap – Cartes

Une carte dans Bootstrap 5 est une boîte bordée avec un peu de rembourrage autour de son contenu. Il comprend des options pour les en-têtes, les pieds de page, le contenu, les couleurs, etc.

La classe `.card-header` ajoute un en-tête à la carte et la classe `.card-footer` ajoute un pied de page à la carte :

Entête
Teneur
Bas de page

```
<div class="card">  
  <div class="card-header">Entête</div>  
  <div class="card-body">Teneur</div>  
  <div class="card-footer">Base de page</div>  
</div>
```

Pour ajouter une couleur d'arrière-plan à la carte, utilisez des classes contextuelles (`.bg-primary`, `.bg-success`, `.bg-info`, `.bg-warning`, `.bg-danger`, `.bg-secondary`, `.bg-dark`, `.bg-light`).

Bootstrap – Cartes - Images de la carte

Ajoutez `.card-img-top` ou `.card-img-bottom` à un `` pour placer l'image en haut ou en bas à l'intérieur de la carte. Notez que nous avons ajouté l'image à l'extérieur de `.card-body` pour couvrir toute la largeur :

```
<div class="card" style="width:400px">

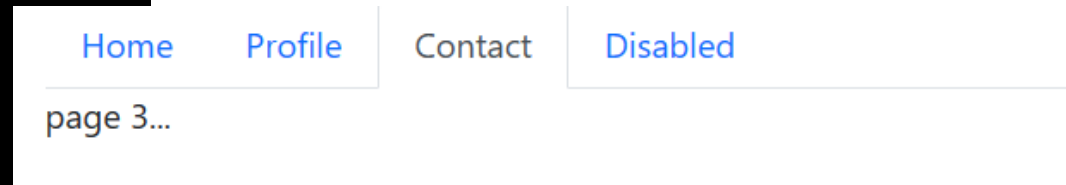
<div class="card-body">
<h4 class="card-title">Mohamed Ali</h4>
<p class="card-text">Quelques exemples de textes.</p>
<a href="#" class="btn btn-primary">Voir Profil</a>
</div>
</div>
```

Si on change `<div class="card-body">`
par
`<div class="card-img-overlay">`
L'image devient en arrière plan



Bootstrap – onglet de navigation

```
<nav>
<div class="nav nav-tabs" id="nav-tab" role="tablist">
<button class="nav-link active" id="nav-home-tab" data-bs-toggle="tab" data-
bs-target="#nav-home" type="button" role="tab" aria-controls="nav-home"
aria-selected="true">Home</button>
<button class="nav-link" id="nav-profile-tab" data-bs-toggle="tab" data-bs-
target="#nav-profile" type="button" role="tab" aria-controls="nav-profile"
aria-selected="false">Profile</button>
<button class="nav-link" id="nav-contact-tab" data-bs-toggle="tab" data-bs-
target="#nav-contact" type="button" role="tab" aria-controls="nav-contact"
aria-selected="false">Contact</button>
<button class="nav-link" id="nav-disabled-tab" data-bs-toggle="tab" data-bs-
target="#nav-disabled" type="button" role="tab" aria-controls="nav-disabled"
aria-selected="false" disabled>Disabled</button>
</div>
</nav>
<div class="tab-content" id="nav-tabContent">
<div class="tab-pane fade show active" id="nav-home" role="tabpanel" aria-
labelledby="nav-home-tab" tabindex="0">page 1...</div>
<div class="tab-pane fade" id="nav-profile" role="tabpanel" aria-
labelledby="nav-profile-tab" tabindex="0">page 2...</div>
<div class="tab-pane fade" id="nav-contact" role="tabpanel" aria-
labelledby="nav-contact-tab" tabindex="0">page 3. de contact</div>
<div class="tab-pane fade" id="nav-disabled" role="tabpanel" aria-
labelledby="nav-disabled-tab" tabindex="0">4...</div>
</div>
```



Connection to JSON file

- Introduction to JSON
- JSON vs XML
- Display JSON data in HTML table using JavaScript



- JSON stands for **J**ava**S**cript **O**bject **N**otation
- JSON is a **text format** for storing and transporting data
- JSON is "self-describing" and easy to understand

JSON vs XML

JSON Example

```
{  
  "employees": [  
    { "firstName": "John", "lastName": "Doe" },  
    { "firstName": "Anna", "lastName": "Smith" },  
    { "firstName": "Peter", "lastName": "Jones" }  
  ]  
}
```

JSON is Like XML Because

- Both JSON and XML are "self describing"
- Both are hierarchical (values within values)
- Both can be parsed and used by lots of programming languages
- Both can be fetched with an XMLHttpRequest

JSON is Unlike XML Because

- JSON doesn't use end tag
- JSON is shorter
- JSON is quicker to read and write
- JSON can use arrays

XML Example

```
<employees>  
  <employee>  
    <firstName>John</firstName> <lastName>Doe</lastName>  
  </employee>  
  <employee>  
    <firstName>Anna</firstName> <lastName>Smith</lastName>  
  </employee>  
  <employee>  
    <firstName>Peter</firstName> <lastName>Jones</lastName>  
  </employee>  
</employees>
```

XML is much more difficult to parse than JSON.
JSON is parsed into a ready-to-use JavaScript object.

Parse : Analyser

JSON – Exemple

N	LastName	FirstName	Email	Photo
1	Ajawin	Sunday Otwel Akat	otwelsunday94@gmail.com	photos/Ajawin-Sunday-Otwel-Akat.jpg
2	Allal El Bakhti	Mouad	allalelbakhti.mouad@gmail.com	photos/Allal-El-Bakhti-Mouad.jpg
3	Azemat	Zainab	zainab.azemat@etu.uae.ac.ma	photos/Azemat-Zainab.jpg
4	Barhoun	Mohamed	mohamedbarhun2020@gmail.com	photos/Barhoun-Mohamed.jpg
5	Benzeyan	Safouan	safouanbenziyane@gmail.com	photos/Benzeyan-Safouan.jpg
6	Bouzarhoun	Ismail	Ismaïlbouzarhoune09@gmail.com	photos/Bouzarhoun-Ismail.jpg
7	El Gazi	Youssef	ghyyoussef@gmail.com	photos/El-Gazi-Youssef.jpg
8	El yamlaoui	Oussama	oussamaelyamlaoui10@gmail.com	photos/El-yamlaoui-Oussama.jpg
9	Ghbalou	Mohammed diyae	meddiya.ghbalou@gmail.com	photos/Ghbalou-Mohammed-Diyae.jpg
10	Haidouch	Yassir	haidouchy5@gmail.com	photos/Haidouch-Yassir.jpg
11	Jmili	Mouad	mouad.jmili@etu.uae.ac.ma	photos/Jmili-Mouad.jpg
12	Lemnaouar	Doha	duha.lmnr@hotmail.com	photos/Lemnaouar-Doha.jpg
13	Mejdoubi	Mehdi	mehdimejdoubi134@gmail.com	photos/Mejdoubi-Mehdi.jpg
14	Zian	Al Amine	alaminezian@gmail.com	photos/Zian-Al-Amine.jpg
15	Zian	Hicham	zhicham63@gmail.com	photos/Zian-Hicham.jpg
16	Zoumhan	hajar	hajar.zoumhan20@gmail.com	photos/Zoumhan-Hajar.jpg

N	Last Name	First Name	Email	Photo
1	Ajawin	Sunday Otwel Akat	otwelsunday94@gmail.com	
2	Allal El Bakhti	Mouad	allalelbakhti.mouad@gmail.com	
3	Azemat	Zainab	zainab.azemat@etu.uae.ac.ma	
4	Barhoun	Mohamed	mohamedbarhun2020@gmail.com	

JSON file

students.json

```
[
  {
    "N": "1",
    "LastName": "Ajawin",
    "FirstName": "Sunday Otwel Akat",
    "Email": "otwelsunday94@gmail.com",
    "Photo": "photos/Ajawin-Sunday-Otwel-Akat.jpg"
  },
  {
    "N": "2",
    "LastName": "Allal El Bakhti",
    "FirstName": "Mouad",
    "Email": "allalelbakhti.mouad@gmail.com",
    "Photo": "photos/Allal-El-Bakhti-Mouad.jpg"
  },
  {
    "N": "3",
    "LastName": "Azemat",
    "FirstName": "Zainab",
    "Email": "zainab.azemat@etu.uae.ac.ma",
    "Photo": "photos/Azemat-Zainab.jpg"
  },
]
```

Convert table to JSON file

<https://tableconvert.com/excel-to-json>

JSON – HTML File – Table with rows from json file

```
<body>
<table>
  <thead>
    <tr>
      <th>N</th>
      <th>Last Name</th>
      <th>First Name</th>
      <th>Email</th>
      <th>Photo</th>
    </tr>
  </thead>
  <tbody id="data-output">
    <!-- Students from javascript file in here. -->
  </tbody>
</table>
  <script src="script.js"></script>
</body>
```

index.html

JSON –Javascript file

In line 1 we use the fetch() method to get the data from the students.json file.

```
1 fetch("stutents.json")
```

The fetch() method returns a Promise object. So in line 2 we use the .then() method to catch the Response object and to resolve it to a javascript object with the .json() method, in line 3.

```
2 .then(function(response){  
3   return response.json();  
4 })
```

The .json() method returns also a promise, so we have to use another .then() method to catch our data (in our case the products). That is what we do in line 5.

The products argument inside the function is holding a javascript array of products.

```
5 .then(function(students){
```

In line 6 we are targeting the table-body in the html file and we storing it in the placeholder variable.

```
6 let placeholder = document.querySelector("#data-output");
```


JSON –Javascript file

- In line 7 we initialize a variable named **out** and we are setting its value to an empty string, so we can use the **out** variable later in the script.

```
7 let out = "";
```

- Next we are going to loop through the students array in line 8 so we can access every student.

```
8 for(let student of students){
```

Inside the for loop, we use the out variable to append a table-row template which holds the student values. We are using a template literal (the back ticks ``) to write the html code.

```
9 out += `  
10 <tr>  
11     <td>${student.LastName}</td>  
12     <td>${student.FirstName}</td>  
13     <td>${student.Email}</td>  
14     <td>${student.Email}</td>  
15     <td> <img src='${student.photo}'> </td>  
16 </tr>  
17 `;  
18 }
```

JSON –Javascript file

We can insert in the html code, javascript variables using this structure `${product.image}`.

And the last thing we have to do is to target the **tbody** element and add the data that the `out` variable holds.

```
19 placeholder.innerHTML = out;  
20 });
```

JSON –Javascript file



```
fetch("students.json")
.then(function(response){
    return response.json();
})
.then(function(students){
    let placeholder = document.querySelector("#data-output");
    let out = "";
    for(let student of students){
        out += `
            <tr>
                <td>${student.N}</td>
                <td>${student.LastName}</td>
                <td>${student.FirstName}</td>
                <td>${student.Email}</td>
                <td> <img src='${student.Photo}'> </td>
            </tr>
        `;
    }
    placeholder.innerHTML = out;
});
```

script.js

JSON – CSS File

```
/* I always reset all the html elements */
*{margin: 0;padding: 0; box-sizing: border-box;}
body{font-family: sans-serif; min-height: 100vh; color: #555;}
table{width: 1000px; margin: 30px auto;}
table th{padding: 10px 0; background-color: #f4f4f4; border: thin solid #d4d4d4;}
table td{padding: 10px; border: thin solid #d4d4d4; width: 18%; text-align: center;
background-color: #fff;}
table img{width: 70%;}
table td{font-weight: bold; font-size: 22px;}
table a{text-decoration: none; color: darkred;background-color: #ffecef; padding: 10px
15px;}
table a:active{background-color: #ccf4d6; color: green;}
table select{padding: 10px; width: 200px; font-size: 16px;
border: thin solid #d4d4d4; background-color: transparent;}
table select:focus{outline: none;}
table img{ width: 60%;}
```

JSON - Result

N	Last Name	First Name	Email	Photo
1	Ajawin	Sunday Otwel Akat	otwelsunday94@gmail.com	
2	Allal El Bakhti	Mouad	allalelbakhti.mouad@gmail.com	
3	Azemat	Zainab	zainab.azemat@etu.uae.ac.ma	
4	Barhoun	Mohamed	mohamedbarhun2020@gmail.com	

Fournisseurs d'Hébergement Web mutualisé







Hébergement web - Maroc

Hébergement web - International

Hébergement web – Critères de choix

- **Espace disque offert >10GO**
- **Nombre de bases de données autorisé >10**
- **Nombre de sous-domaines autorisé >10**
- **Nombre de compte email >10**
- **Contrôle Cpanel**
- **CMS proposes par la plateforme (Wordpress, Drupal, Joomla, Woocommerce, PrestaShop, Magento...)**
- **Gestion des sauvegardes (gratuite ou payante)**
- **Prix d'hébergement**
- **Prix du nom de domaine (.org, .com, .ma, ...)**
- **Prix total en 1ère année, et prix après la 1ère année**

Hébergement web - Maroc

Site hébergeur	Logo	L'hébergement le moins cher (HT)	Taille min	.ma (HT)
https://www.heberjahiz.com		300	50 GO 5 sous-domaines	119
https://www.heberfacile.com		199	10 GO	139
https://www.adk-media.com		422	100 GO	
https://nindohost.ma		300	100 GO	1 ^{ère} année gratuit
https://www.ovhcloud.com/fr-ma		127,44	1 GO	Pas .ma .ovh 22,1DH
https://www.capconnect.com		120	10 GO 5 sous-domaines	118

Prix – Nov. 2023 - Voir s'il y a les promo. Black Friday

Hébergement web - International

Recommandé pour les sites e-commerce

<https://www.hostinger.com>



<https://www.godaddy.com>



<https://www.hostgator.com>



<https://www.bluehost.com>



<https://www.ionos.com>



Souvent les promo. Black Friday en novembre

Cours en ligne avec Certifications

OPENCLASSROOMS

coursera

udemy



sololearn

<https://www.classcentral.com/>

Cours avec Certifications

Nous encourageons constamment nos étudiants et diplômés à poursuivre des certifications dans des produits très recherchés sur le marché du travail.

We consistently encourage our students and graduate pursue certifications in products that are highly sought after in the job market



Cours avec Certifications

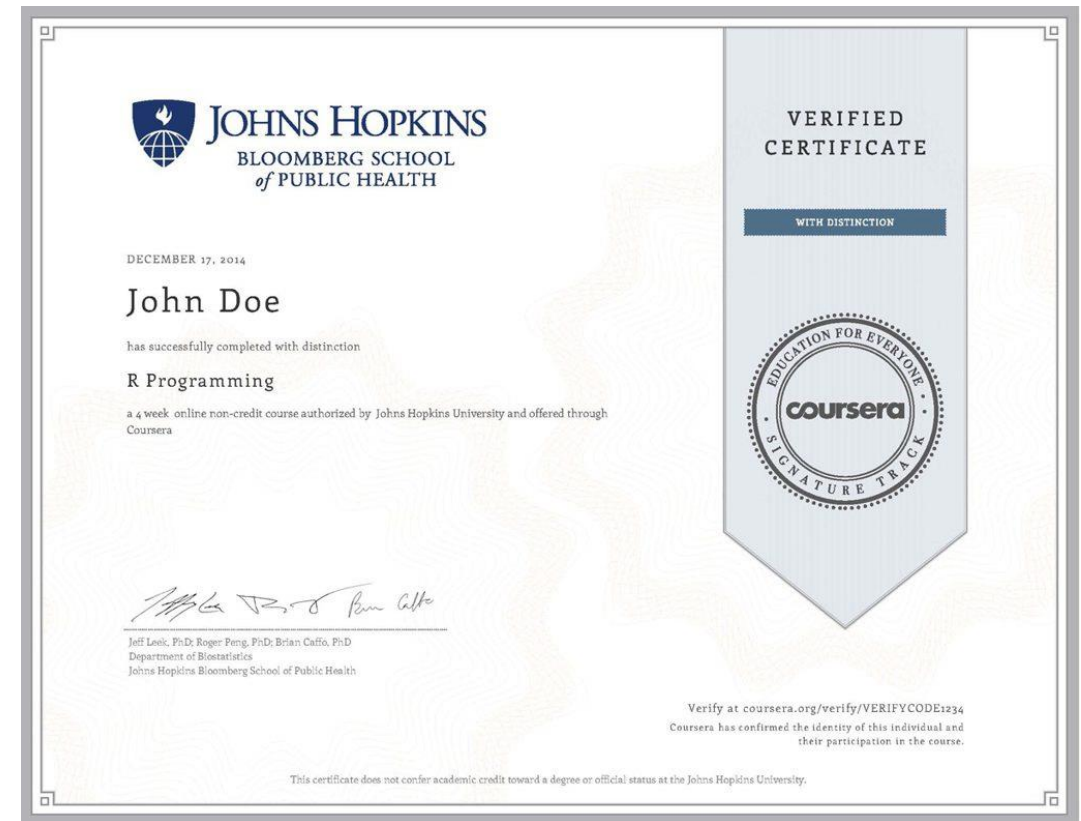
<https://openclassrooms.com>

OPENCLASSROOMS



<https://www.coursera.org>

coursera



Cours avec Certifications

<https://www.udemy.com>



<https://www.sololearn.com/learn>

