

BOOTSTRAP

I. Introduction

- A. Définition
- B. Chargement des fichiers
- C. Reboot

II. Les premières classes

- A. Typographie et texte
- B. Les couleurs
- C. Tableaux et images

III. La mise en page

- A. Box Model
- B. Display
- C. Position et Flexbox

IV. La grille Bootstrap

- A. Les conteneurs
- B. Colonnes et rangées
- C. Les breakpoints

INTRODUCTION

Définition

Bootstrap est un framework **CSS** Open Source développé en **2011** par deux ingénieurs de Twitter : Marc Otto et Jacob Thornton

L'objectif de cette énorme boîte à outils est de faciliter la création du **design** de pages web et de pouvoir rapidement leur assurer un comportement **responsive** tout en étant compatible sur l'ensemble des navigateurs

Le framework **Bootstrap** est constitué de **classes** CSS contenant chacune une série de déclarations de styles, mais également de **composants** prêts à l'emploi tels que des sliders ou des navbars

L'intégration de **Bootstrap** dans un projet peut se faire en **téléchargeant** l'ensemble des fichiers **CSS** nécessaires ou en les chargeant en ligne via un lien **CDN**

Le fonctionnement de certains **composants** dynamiques peut également nécessiter l'utilisation de **JavaScript**. Il faudra alors également charger les **fichiers JS**

La **version 5** de Bootstrap en place depuis **juin 2020** comporte des changements majeurs par rapport à la version précédente : l'abandon du support sur **Internet Explorer** mais également de l'utilisation de **jQuery**

Une des forces de Bootstrap est d'être parfaitement **responsive**, c'est à dire qu'il permet d'**adapter** la disposition des éléments aux **tailles d'écrans** grâce à l'utilisation de sa **grille**

Un ensemble de **préfixes** à ajouter aux classes est ainsi disponible, chaque préfixe correspondant à un changement de largeur d'écran appelé **breakpoint**

Bootstrap étant écrit selon le modèle **mobile first**, il ne propose pas de préfixe spécifique aux **petites tailles** d'écrans puisque ce sont justement ces tailles qui sont ciblées par défaut

Chargement des fichiers

📌 <https://getbootstrap.com/docs/5.2/getting-started/download>

Compiled CSS and JS

Download ready-to-use compiled code for **Bootstrap v5.2.2** to easily drop into your project, which includes:

- Compiled and minified CSS bundles (see [CSS files comparison](#))
- Compiled and minified JavaScript plugins (see [JS files comparison](#))

This doesn't include documentation, source files, or any optional JavaScript dependencies like Popper.

Download

Le dossier téléchargé ici contient l'ensemble des fichiers **CSS** et **JavaScript**

📌 <https://getbootstrap.com/docs/5.2/getting-started/download>

Source files

Compile Bootstrap with your own asset pipeline by downloading our source Sass, JavaScript, and documentation files. This option requires some additional tooling:

- [Sass compiler](#) for compiling Sass source files into CSS files
- [Autoprefixer](#) for CSS vendor prefixing

Should you require our full set of [build tools](#), they are included for developing Bootstrap and its docs, but they're likely unsuitable for your own purposes.

Download source

Le dossier téléchargé ici contient la totalité des fichiers : **CSS**, **JavaScript** mais aussi **Sass**, **json** ...

📌 <https://getbootstrap.com/docs/5.2/getting-started/download>

CDN via jsDelivr

Skip the download with [jsDelivr](#) to deliver cached version of Bootstrap's compiled CSS and JS to your project.

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-VK96RwkvWFO1wpeNBV4NqZv8XQyJKV4ngV1VtZQwVO58Uyrh72j43RwDvmuyi" crossorigin="anonymous">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-OERFAA2qR480srbV6/tORAZ04v3SM/IRBP746278Ktd937747SqnsX44Q7MKZFx4" crossorigin="anonymous">
```

La dernière possibilité consiste à utiliser les liens **CDN** pour charger les éléments nécessaires **en ligne**

Quelque soit l'alternative choisi, les fichiers **CSS** seront à intégrer via une balise `<link>` au sein de la balise `<head>` du fichier **HTML**

Quant aux fichiers **JavaScript**, ils seront à intégrer via une balise `<script>` placée juste avant la fermeture de la balise `<body>` du fichier **HTML**

Comme tout fichier **JavaScript**, il est également possible de placer cette balise `<script>` dans la balise `<head>` mais il faudra alors y ajouter l'attribut **defer** afin de retarder son exécution

Il est toujours possible d'ajouter ses **propres** déclarations **CSS** dans un fichier à part. Il faudra alors ajouter le lien vers celui-ci juste **après** les liens des fichiers **Bootstrap**

```
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title> Bootstrap Basics </title>
  <link href="css/bootstrap.min.css" rel="stylesheet">
  <link href="css/personal-style.css" rel="stylesheet">
</head>
<body>
  <script src="js/bootstrap.bundle.min.js"></script>
</body>
```

Reboot

La première chose que fait **Bootstrap** est de mettre en place un système de **reboot** basé sur le fichier **Normalize.css** afin de supprimer certains styles présents par défaut dans le HTML comme les marges ou les couleurs

Il peut alors définir ses propres règles CSS par défaut. Ainsi, avant même d'utiliser les classes de Bootstrap, on bénéficie déjà du style Bootstrap

Parmi ces règles de base on trouve l'application de `{ box-sizing: border-box }` sur l'ensemble des éléments, l'utilisation standardisée des **rem** et **em**, ou bien encore une palette de **couleurs** spécifiques définies dans des variables

```
body {  
    margin: 0;  
    font-family: var(--bs-body-font-family);  
    font-size: var(--bs-body-font-size);  
    font-weight: var(--bs-body-font-weight);  
    line-height: var(--bs-body-line-height);  
    color: var(--bs-body-color);  
    text-align: var(--bs-body-text-align);  
    background-color: var(--bs-body-bg);  
    -webkit-text-size-adjust: 100%;  
    -webkit-tap-highlight-color: transparent;  
}  
  
*, ::after, ::before {  
    box-sizing: border-box;  
}
```

reboot.scss:50

reboot.scss:19

LES PREMIÈRES CLASSES

Typographie et texte

✦ <https://getbootstrap.com/docs/5.2/content/typography>

Bootstrap propose différentes classes pour les **titres**, comme les classes **.h1** jusqu'à **.h6**, ou encore **.display-1** jusqu'à **.display-6**

La classe **.lead** permet de mettre en évidence un paragraphe, à la manière du **chapô** d'un article pour attirer l'attention du lecteur sur celui-ci

L'utilisation conjointe de quelques classes telles que **.blockquote** permet par exemple de rapidement optimiser le rendu de citations

```
<h1 class="display-4"> This title has class display-4 </h1>
```

```
<p>Lorem, ipsum dolor sit amet consectetur adipisicing elit. Ducimus magni corrupti doloribus quam reiciendis sequi sunt earum ad quis eaque quisquam expedita veritatis aspernatur quos sed, dolores nulla nostrum rerum quaerat amet velit facilis natus sapiente eos. Quibusdam dolor atque nostrum nemo. Illo ipsam placeat voluptas dolor ab officiis inventore, sapiente fugiat vel quibusdam eius nesciunt odit.</p>
```



This title has class display-4

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Ducimus magni corrupti doloribus quam reiciendis sequi sunt earum ad quis eaque quisquam expedita veritatis aspernatur quos sed, dolores nulla nostrum rerum quaerat amet velit facilis natus sapiente eos. Quibusdam dolor atque nostrum nemo. Illo ipsam placeat doloremque voluptas dolor ab officiis inventore, sapiente fugiat vel quibusdam eius nesciunt odit.

```
<article>
  <p class="lead">
    Here are the last best VSCode Extensions of 2022 for developers
  </p>
  <p> Lorem ipsum dolor sit amet, consectetur adipisicing elit. Nemo,
    accusantium, rem modi quaerat accusamus voluptates assumenda vero
    architecto aliquid quod. Nisi similique quo, facere vel dolore adipisci qui.
  </p>
</article>
```



Here are the last best VSCode Extensions of 2022 for developers

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Nemo, accusantium, rem modi quaerat accusamus voluptates assumenda vero in architecto aliquid quas quod. Nisi similique quo, facere vel dolore adipisci qui.

```
<figure class="text-center">
  <blockquote class="blockquote">
    <p>A well-known quote, contained in a blockquote element.</p>
  </blockquote>
  <figcaption class="blockquote-footer">
    Someone famous in <cite title="Source Title">Source Title</cite>
  </figcaption>
</figure>
```



A well-known quote, contained in a blockquote element.

— Someone famous in *Source Title*

✦ <https://getbootstrap.com/docs/5.2/utilities/text>

Bootstrap propose les classes `.text-start`, `.text-center` et `.text-end` permettant de gérer l'alignement des textes. Il n'y a cependant pas de classe permettant de justifier le texte ...

La casse d'un texte se fait avec l'ajout des classes `.text-lowercase`, `.text-uppercase` et `.text-capitalize`

Pour ce qui est de la taille des polices, il faut utiliser les classes `.fs1` pour les plus grandes à `.fs6` pour les plus petites

✦ <https://getbootstrap.com/docs/5.2/utilities/text>

L'épaisseur des polices est gérée à l'aide d'une série de classes allant de `.fw-lighter` à `.fw-bold`

D'autres classes sont également disponibles pour modifier d'autres propriétés CSS comme `line-height` ou bien `font-style` ou encore `text-decoration`

Les éléments HTML peuvent posséder un nombre illimité de classes, il suffit simplement de mettre un espace entre chaque nom de classe

<p> Lorem ipsum dolor sit amet consectetur adipisicing elit. Necessitatibus provident veritatis. Adipisci ratione qui vel, exercitationem consequuntur, at saepe vero quaerat corporis perferendis dolore commodi? </p>

<p class="fs-5 fw-lighter text-decoration-underline"> Lorem ipsum dolor sit amet consectetur adipisicing elit. Atque necessitatibus provident veritatis. Adipisci ratione qui vel, exercitationem consequuntur, at saepe vero quaerat corporis perferendis dolore commodi? </p>



Lorem ipsum dolor sit amet consectetur adipisicing elit. Atque necessitatibus provident veritatis. Adipisci ratione qui vel, exercitationem consequuntur, at saepe vero quaerat corporis perferendis dolore praesentium magnam ipsam eos? Odio fuga laudantium aut quo voluptatibus perspiciatis excepturi consectetur magni commodi?

Lorem ipsum dolor sit amet consectetur adipisicing elit. Atque necessitatibus provident veritatis. Adipisci ratione qui vel, exercitationem consequuntur, at saepe vero quaerat corporis perferendis dolore praesentium magnam ipsam eos? Odio fuga laudantium aut quo voluptatibus perspiciatis excepturi consectetur magni commodi?

Les couleurs

✦ <https://getbootstrap.com/docs/5.2/utilities/colors>

Bootstrap propose une palette de quelques couleurs de bases utilisables pour les **polices** et/ou les **backgrounds**. Ces couleurs sont très aisément identifiables et rapidement associées à Bootstrap

✦ <https://getbootstrap.com/docs/5.2/customize/css-variables>

S'ajoutent à celles-ci quelques couleurs supplémentaires déclarées à la racine **:root** et ainsi disponibles sous formes de **variables**

Il est aussi possible d'optimiser la **personnalisation** des couleurs via l'utilisation de **Sass** comme pour la plupart des autres propriétés CSS gérées par Bootstrap

```
<span class="text-primary"> .text-primary </span>  
<span class="text-secondary"> .text-secondary </span>  
<span class="text-success"> .text-success </span>  
<span class="text-info"> .text-info </span>  
<span class="text-danger"> .text-danger </span>  
<span class="text-warning"> .text-warning </span>  
<span class="text-black-50"> .text-black-50 </span>  
<span> var(--bs-pink) </span>
```



`.text-primary` `.text-secondary` `.text-success` `.text-info` `.text-danger` `.text-warning` `.text-black-50` `var(--bs-pink)`

📌 <https://getbootstrap.com/docs/5.2/utilities/background>

Pour associer des couleurs de **background** aux couleurs de **texte**, il suffit simplement d'associer le préfixe **bg-** aux mêmes classes vues précédemment

Certaines classes restent spécifiques aux textes comme **.text-black-50**, et d'autres spécifiques aux backgrounds comme **.bg-transparent**

📌 <https://getbootstrap.com/docs/5.2/helpers/colored-links>

Des classes sont également disponibles pour les **liens** avec un léger changement de couleur lors du **hover** sur l'élément

```
<div class="bg-primary text-white"> .bg-primary </div>  
<div class="bg-secondary text-white"> .bg-secondary </div>  
<div class="bg-success text-white"> .bg-success </div>  
<div class="bg-danger text-white"> .bg-danger </div>  
<div class="bg-warning text-dark"> .bg-warning </div>  
<div class="bg-info text-dark"> .bg-info </div>  
<div class="bg-light text-dark"> .bg-light </div>  
<div class="bg-dark text-white"> .bg-dark </div>
```



.bg-primary

.bg-secondary

.bg-success

.bg-danger

.bg-warning

.bg-info

.bg-light

.bg-dark

Tableaux et images

✦ <https://getbootstrap.com/docs/5.2/content/tables>

On trouve un ensemble de règles définissant le style par défaut des **tableaux** dans les fichiers **reboot.scss** et **tables.scss**

La classe principale **.table** est également disponible, à laquelle peuvent être ajoutées plusieurs autres classes comme **.table-{color}** pour gérer les backgrounds du tableau

La classe **.table-striped** permet de réaliser une alternance de dégradés entre les différentes rangées d'un tableau et la classe **.table-hover** permet d'ajouter un effet au hover


```
<table class="table table-striped">
  <thead class="table-warning">
    <tr>...</tr>
  </thead>
  <tbody class="table-success">
    <tr>...</tr>
    <tr>...</tr>
  </tbody>
</table>
```



#	First	Last	Handle
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat

✦ <https://getbootstrap.com/docs/5.2/content/images>

Bootstrap propose la classe `.img-fluid` permettant de rendre **responsive** une image par rapport à son conteneur.

Il est possible de rajouter un cadre discret à une image avec la classe `.thumbnail`. D'autres classes sont évoquées dans cette rubrique comme `.rounded` pour arrondir les angles ou encore `.mx-auto` `.d-block` pour centrer une image

Ces déclarations ne sont cependant pas spécifiques aux images mais aux bordures et aux marges et seront donc abordées dans les sections suivantes

```

```



LA MISE EN PAGE

Box model

✦ <https://getbootstrap.com/docs/5.2/utilities/sizing>

Différentes classes permettent de gérer les **tailles** des éléments. Les classes **.w-** et **.h-** suivies d'une des valeurs proposées [25 - 50 - 75 - 100 - auto] servent à indiquer la **largeur** ou la **hauteur** de l'élément en **pourcentage**

Il est possible de préciser une largeur ou une hauteur **maximale** de **100%** avec les classes **.mv-100** et **.mh-100**

Les largeurs et hauteurs précisées par rapport au **viewport** sont déclarées avec les classes **.min-vw-100** ou **.vw-100** et **.min-vh-100** ou **.vh-100**

✦ <https://getbootstrap.com/docs/5.2/utilities/spacing>

Pour déclarer les différentes **marges** d'un élément, la syntaxe à suivre est la suivante :

`{property} {sides} - {breakpoint} - {size}`

L'écriture reprend les syntaxes **CSS** habituelles, à l'exception des valeurs **left** et **right** remplacées par **start** et **end**. Les **tailles** sont indiquées à l'aide d'un indice entre **0** et **5**

Un **breakpoint** peut aussi être ajouté permettant de choisir de la **largeur** d'écran à partir de laquelle seront appliquées les déclarations

Where *property* is one of:

- **m** - for classes that set **margin**
- **p** - for classes that set **padding**

Where *sides* is one of:

- **t** - for classes that set **margin-top** or **padding-top**
- **b** - for classes that set **margin-bottom** or **padding-bottom**
- **s** - (start) for classes that set **margin-left** or **padding-left** in LTR, **margin-right** or **padding-right** in RTL
- **e** - (end) for classes that set **margin-right** or **padding-right** in LTR, **margin-left** or **padding-left** in RTL
- **x** - for classes that set both ***-left** and ***-right**
- **y** - for classes that set both ***-top** and ***-bottom**
- blank - for classes that set a **margin** or **padding** on all 4 sides of the element

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

.ps-0



padding-left = 1.5rem

.pt-3



padding-top = 0

.mx-auto



margin = 0 auto

.m-lg-5



margin = 3rem [>= 992px]

★ <https://getbootstrap.com/docs/5.2/utilities/borders>

La classe générale `.border` permet de gérer les **bordures** des éléments. Il faut ensuite ajouter des classes supplémentaires comme `.border-start` pour les côtés ou comme `.border-warning` pour préciser la couleur

Les tailles de bordure seront déclarées avec des classes allant de `.border-0` à `.border-5`. Le **border-radius** sera géré avec des classes comme `.rounded-top` et les classes `.rounded-0` jusqu'à `.rounded-5`

D'autres degrés de radius sont également disponibles avec les classes `.rounded-circle` ou `.rounded-pill`

`.border .border-warning .border-3 .rounded`



Lorem ipsum dolor, sit amet consectetur adipisicing elit. Sed sit necessitatibus iure nihil. In ad rem eos aperiam odit ipsa ut, ab hic quisquam culpa sit tempore atque sapiente sequi aut aspernatur laboriosam eius quibusdam impedit iusto nesciunt dolor id odio. Tenetur aliquid et error ea omnis dolorum quas rem?

Display

📌 <https://getbootstrap.com/docs/5.2/utilities/display>

Pour décider de l'affichage ou non d'un élément, la syntaxe à suivre est la suivante :

-d - {breakpoint} - {value}

Il faut tout simplement préciser à l'aide des différentes valeurs ainsi que des différents breakpoints à partir de quelle largeur d'écran afficher l'élément mais également à partir de quelle largeur d'écran ne plus afficher l'élément, et inversement

Visible on all	<code>.d-block</code>
Visible only on xs	<code>.d-block .d-sm-none</code>
Visible only on sm	<code>.d-none .d-sm-block .d-md-none</code>
Visible only on md	<code>.d-none .d-md-block .d-lg-none</code>
Visible only on lg	<code>.d-none .d-lg-block .d-xl-none</code>
Visible only on xl	<code>.d-none .d-xl-block .d-xxl-none</code>
Visible only on xxl	<code>.d-none .d-xxl-block</code>

Hidden on all	<code>.d-none</code>
Hidden only on xs	<code>.d-none .d-sm-block</code>
Hidden only on sm	<code>.d-sm-none .d-md-block</code>
Hidden only on md	<code>.d-md-none .d-lg-block</code>
Hidden only on lg	<code>.d-lg-none .d-xl-block</code>
Hidden only on xl	<code>.d-xl-none</code>
Hidden only on xxl	<code>.d-xxl-none .d-xxl-block</code>

Position et FlexBox

★ <https://getbootstrap.com/docs/5.2/utilities/position>

Les différents types de **positionnement** des éléments sont accessibles avec des classes comme **.position-absolute**. Les quatre **directions** pour les déplacements sont gérées avec les classes correspondantes comme **.top** associées à des pourcentages parmi **[0 - 50 - 100]**

Un élément positionné se déplaçant à partir de son coin haut-gauche, on rectifie habituellement ce décalage avec la déclaration **transform: translate(-50%, -50%)**

Ce rapide recentrage est également disponible chez Bootstrap avec l'ajout de la classe **.translate-middle**

✦ <https://getbootstrap.com/docs/5.2/utilities/flex>

La totalité du système **flexbox** peut être gérée avec toute une série de classes reprenant l'ensemble des **propriétés** flex ainsi que leurs **valeurs**

La classe principale est **.d-flex**. Les classes suivantes sont toutes aisément identifiables. Il est aussi possible d'ajouter un **breakpoint** à chacune de ces classes, rendant ainsi complètement **responsive** la gestion des containers et de leur contenu :

.flex-column-reverse | **.align-items-md-center...**

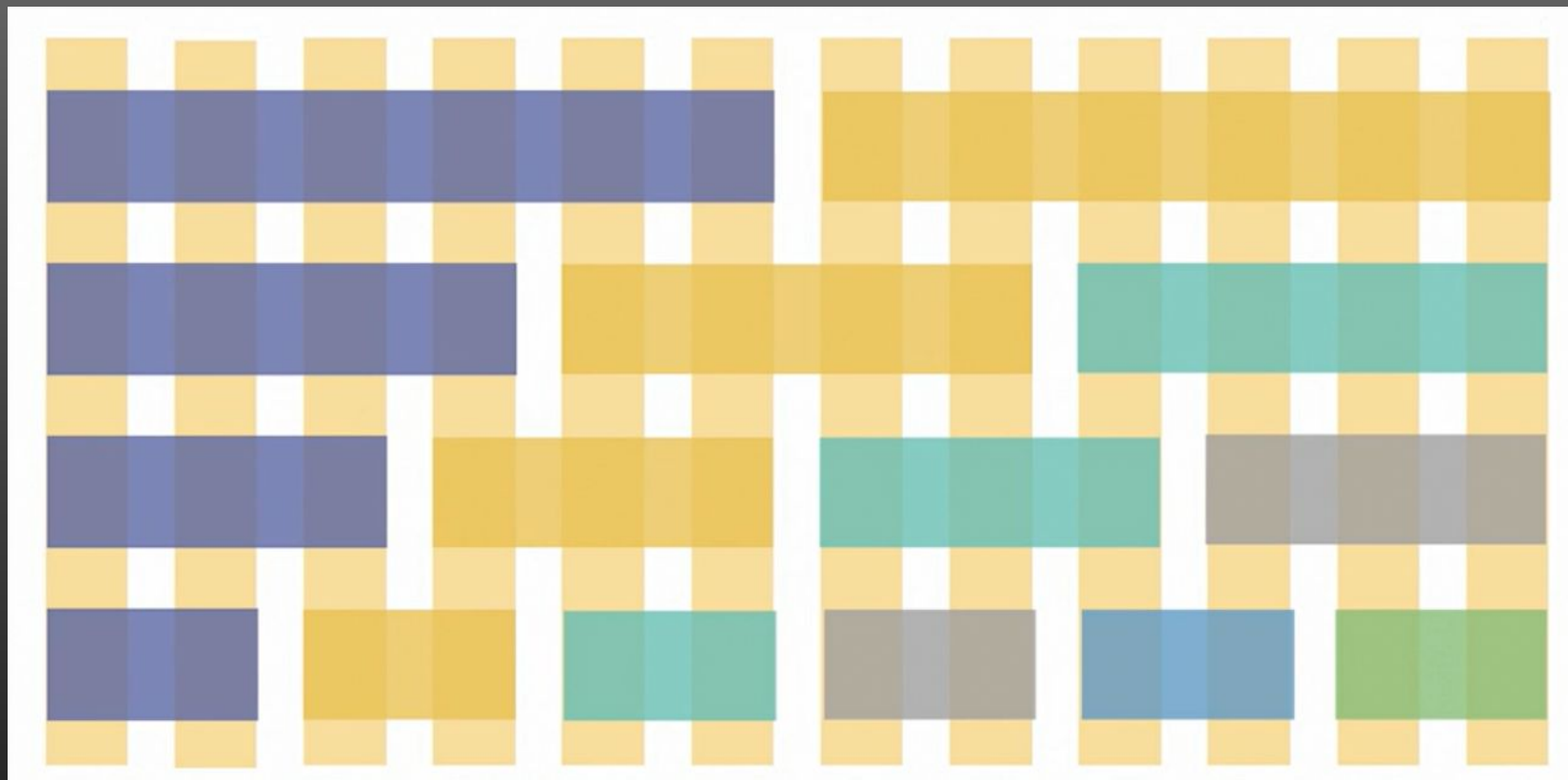
LA GRILLE BOOTSTRAP

★ <https://getbootstrap.com/docs/5.2/layout/grid>

Un des atouts de Bootstrap est sa **grille responsive** permettant une gestion très rapide de la mise en page des éléments. Cette grille est contenue dans un **container** principal fait de **rangées** et de **colonnes**

Chaque **rangée** au sein de cette grille dispose de **12 colonnes**. Il suffit alors simplement de préciser pour chaque élément de la page le nombre de colonnes à occuper

Viennent s'ajouter à cela les **breakpoints** qu'il est possible d'utiliser à toutes les niveaux de construction de la grille



Les conteneurs

✳ <https://getbootstrap.com/docs/5.2/layout/containers>

Plusieurs classes de **container** sont disponibles et ce qui va différencier toutes ces classes sera la **largeur** effective du container

A chaque classe correspond un **breakpoint** à partir duquel une **marge** apparaît sur les côtés. Avant ces breakpoints, tous les containers occupent toute la largeur disponible

Seule la classe **.container-fluid** propose un container qui occupera toujours la **totalité** de la largeur disponible quelque soit la largeur de l'écran

	Extra small <576px	Small ≥576px	Medium ≥768px	Large ≥992px	X-Large ≥1200px	XX-Large ≥1400px
<code>.container</code>	100%	540px	720px	960px	1140px	1320px
<code>.container-sm</code>	100%	540px	720px	960px	1140px	1320px
<code>.container-md</code>	100%	100%	720px	960px	1140px	1320px
<code>.container-lg</code>	100%	100%	100%	960px	1140px	1320px
<code>.container-xl</code>	100%	100%	100%	100%	1140px	1320px
<code>.container-xxl</code>	100%	100%	100%	100%	100%	1320px
<code>.container-fluid</code>	100%	100%	100%	100%	100%	100%

Colonnes et rangées

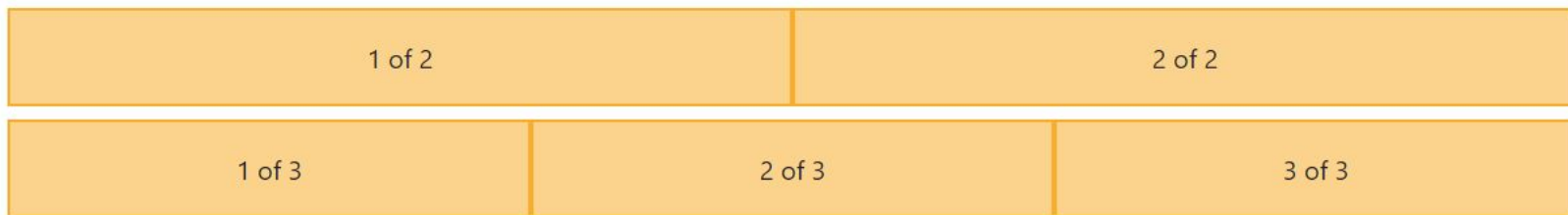
✦ <https://getbootstrap.com/docs/5.2/layout/columns>

Pour créer une **rangée** à l'intérieur d'un container, il faut utiliser la classe **.row**. Chaque rangée ainsi créée possède **12 colonnes** à pouvoir répartir entre les éléments

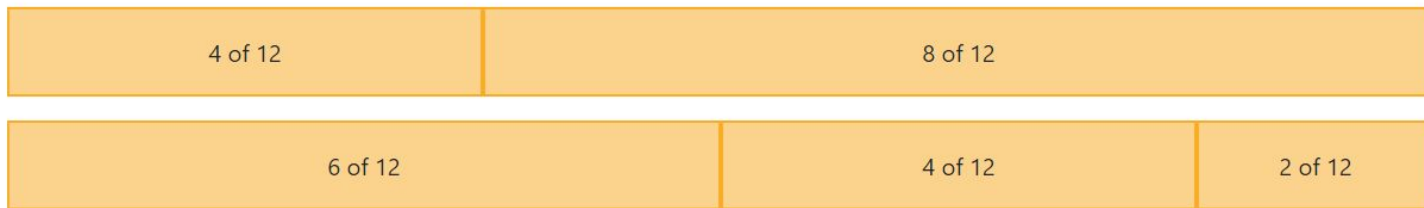
Pour créer une colonne, il faut utiliser la classe **.col**. Les 12 colonnes seront alors équitablement **réparties** entre tous les éléments de la rangée possédant cette classe

Il est possible de préciser le **nombre** de colonnes à occuper en ajoutant ce nombre à la classe **.col**. Les colonnes restantes seront alors réparties entre les autres éléments

```
<div class="container">  
  <div class="row">  
    <div class="col"> 1 of 2 </div>  
    <div class="col"> 2 of 2 </div>  
  </div>  
  <div class="row">  
    <div class="col"> 1 of 3 </div>  
    <div class="col"> 2 of 3 </div>  
    <div class="col"> 3 of 3 </div>  
  </div>  
</div>
```



```
<div class="container">
  <div class="row">
    <div class="col-4"> 4 of 12 </div>
    <div class="col-8"> 8 of 12 </div>
  </div>
  <div class="row">
    <div class="col-6"> 6 of 12 </div>
    <div class="col-4"> 4 of 12 </div>
    <div class="col"> 2 of 12 </div>
  </div>
</div>
```



S'il n'y a **pas assez** de colonnes disponibles pour un élément, celui-ci ira alors se placer en-dessous dans une **nouvelle rangée**

La grille Bootstrap est bâtie sur le modèle **flexbox**. Chacune de ses rangées est donc un **container flex** et tous les éléments s'y trouvant sont donc des **flex-items**

L'ensemble des propriétés **flexbox** sont donc disponibles pour décider de l'**alignement** ou de l'**ordre** des différents éléments. Les classes sont celles abordées dans la section flex

```
▼ .row {  
  --bs-gutter-x: 1.5rem;  
  --bs-gutter-y: 0;  
  display: flex;  
  flex-wrap: wrap;  
  margin-top: calc(-1 * var(--bs-gutter-y));  
  margin-right: calc(-0.5 * var(--bs-gutter-x));  
  margin-left: calc(-0.5 * var(--bs-gutter-x));  
}
```

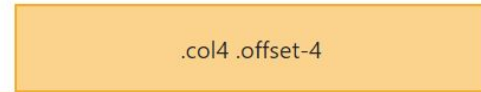
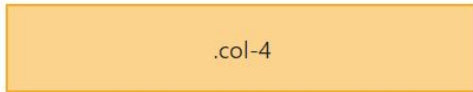
✦ <https://getbootstrap.com/docs/5.2/layout/columns/#offsetting-columns>

Il est possible de laisser des **colonnes vides** au sein d'une rangée avec la classe **.offset** suivie du nombre de colonnes. Cet espace vide sera placé **avant** l'élément possédant cette classe

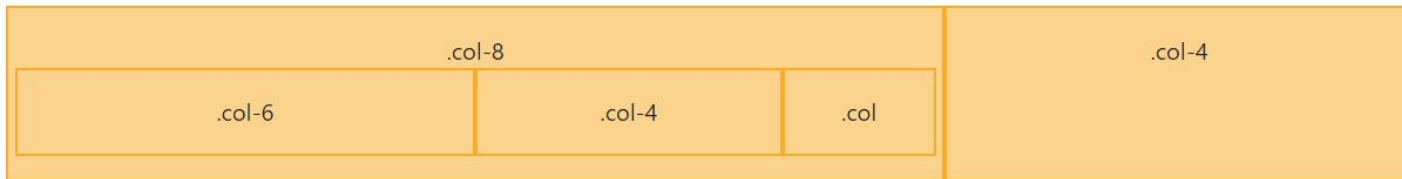
✦ <https://getbootstrap.com/docs/5.2/layout/grid/#nesting>

Une nouvelle **grille** peut être créée à l'intérieur d'une **colonne** en utilisant simplement la classe **.row** qui mettra alors 12 nouvelles colonnes à disposition


```
<div class="container">  
  <div class="row">  
    <div class="col-4"> 4 of 12 </div>  
    <div class="col-4 offset-4"> 4 of 12 </div>  
  </div>  
  <div class="row">  
    <div class="col-4 offset-4"> 4 of 12 </div>  
  </div>  
</div>
```



```
<div class="row">
  <div class="col-8"> .col-8 </div>
  <div class="row">
    <div class="col-6"> .col-6 </div>
    <div class="col-4"> .col-4 </div>
    <div class="col"> .col </div>
  </div>
</div>
<div class="col-4"> .col-4 </div>
</div>
```



Les breakpoints

★ <https://getbootstrap.com/docs/5.2/layout/breakpoints>

Pour chaque **colonne** présente à l'intérieur de la grille Bootstrap , il est possible de préciser un **breakpoint** pour lui assurer un comportement **responsive** :

`.col-{breakpoint}-{columns}`

La classe ainsi écrite permet d'associer un **nombre** de colonnes à une **largeur** d'écran à partir de laquelle appliquer cette mise en forme

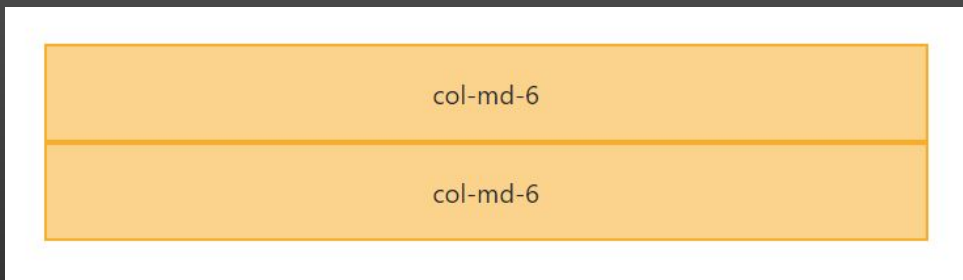
Bootstrap adoptant l'approche **Mobile First**, il n'y a pas besoin de préciser de breakpoint pour les **très petits écrans** : les éléments prendront par défaut **toute la largeur** disponible

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

```
<div class="container">  
  <div class="row">  
    <div class="col-md-6"> .col-md-6 </div>  
    <div class="col-md-6"> .col-md-6 </div>  
  </div>  
</div>
```



< 768px



>= 768px