



BOOTSTRAP 4 & 5

HTML, CSS & JS framework for RWD & mobile first projects

Sacha RESTOUEIX | 2021

sacha8milo@gmail.com

PRÉAMBULE

Bootstrap est le plus populaire de tous les **frameworks HTML, CSS & JS**. Ce document vous présente la version 4 et évoque les modifications de la **version 5** (<https://getbootstrap.com/docs/5.0/getting-started/introduction/>)

En programmation informatique, un **framework** Web est un ensemble cohérent de **composants structurels**, qui sert à créer les fondations ainsi que les grandes lignes de tout ou d'une partie d'un site Web.

L'objectif de Bootstrap est simple : gagner en productivité et faciliter le passage à l'ère du mobile (**Responsive design & mobile first**). Depuis la version 4 du Framework son orientation est clairement **Mobile First**.

Bootstrap utilise les dernières fonctionnalités de mise en page, avec l'utilisation des modules CSS3 **Flexbox** et **Grid** proposés par le W3C.

Bootstrap met donc à notre disposition des tas de choses préconçues en HTML, CSS et JS pour nous faciliter la vie :

- Le principe de mise en page sous forme de **grille** comprenant X colonnes,
- des formulaires pré-stylisés,
- des boutons pré-stylisés,
- des outils de navigation,
- des extensions JS,
- etc.

Le tout est compatible avec tous les navigateurs récents et avec Internet Explorer à partir de sa version 10 et plus. Avec la **version 5**, Internet Explorer n'est plus pris en compte à partir de sa version 11

De plus, dans sa **version 5**, la bibliothèque **JQuery** a été mise de côté comme prérequis à l'utilisation des composants JS de Bootstrap. Ce qui ne veut pas dire que vous ne pouvez pas utiliser **JQuery** dans vos développements conjointement à Bootstrap !!!



TABLE DES MATIÈRES

PRÉAMBULE	2
INSTALLATION	4
• Installation avec npm	4
• Installation par téléchargement classique.....	4
LE CSS POUR LE TEXTE	5
• Le corps de page	5
• Les paragraphes	5
• Les titres.....	5
• Les mises en évidence.....	6
• L'alignement du texte	7
• Le texte coloré.....	7
• Les adresses	8
• Les listes à puces.....	8
• L'affichage du code	8
LE CSS POUR LES FORMULAIRES	9
• La structure de base	9
• Les formulaires en ligne.....	10
• D'autres exemples.....	10
• Les boutons	11
LE CSS POUR LES IMAGES	12
• Par défaut	12
• Arrondie	12
• Inscrite dans un cercle / oval.....	12
• Adaptative	13
LE CSS POUR LA MISE EN PAGE	14
• Le flottement des blocs	14
LA GRILLE ADAPTATIVE	15
• Les paliers et le conteneur principal.....	16
• La largeur des boîtes avec Bootstrap	16
• L'utilisation des lignes et de la class .row.....	17
• La largeur des colonnes.....	17
• Les blocs décalés.....	18
• Afficher et masquer des éléments	18
COMPOSANTS PRÉ-FABRIQUÉS	19
PLUGINS JAVASCRIPT	20
• Tooltips	20
• Fenêtres modales.....	20
• Scrollspy	21
• Les accordéons.....	21
• Les carrousels d'images	22
PERSONNALISEZ VOTRE EXPÉRIENCE	23

INSTALLATION

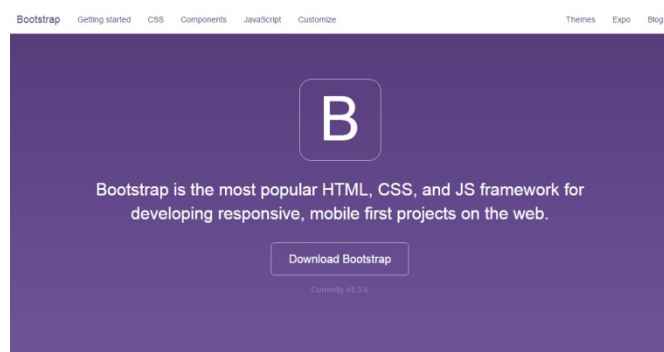
● Installation avec npm

Pour installer Bootstrap comme dépendance de votre projet, rien de plus simple, vous indiquez dans votre console :

```
npm install bootstrap
```

● Installation par téléchargement classique

Vous vous rendez sur le site officiel de bootstrap (<http://getbootstrap.com/>) et vous cliquez sur le gros bouton de téléchargement qui va vous rediriger vers la page de téléchargement où vous sélectionnerez le pack dont vous avez besoin (en sass ou en less).



Designed for everyone, everywhere.

Bootstrap makes front-end web development faster and easier. It's made for folks of all skill levels, devices of all shapes, and projects of all sizes.

Si vous optez pour cette solution, voici ce que vous retrouverez dans votre fichier zippé :



Dans le dossier **CSS**, vous retrouverez les versions **minifiées** et **non-minifiées** de toutes les règles CSS de Bootstrap et des thèmes de Bootstrap qui permettent de modifier le style de base de Bootstrap.

Dans le dossier **js**, vous trouverez tout ce qui est déjà prévu par Bootstrap niveau JS (slider, burger, etc.).

Vous glissez donc l'ensemble de ces fichiers dans votre projet et c'est parti pour l'aventure **Bootstrap** !

LE CSS POUR LE TEXTE

Avec **Bootstrap**, toute une série de **règles sont prédéfinies** pour vous. Il est bon de savoir lesquelles et comment les contourner/changer. Suivant l'**évolution des versions** du Framework, ces **règles** sont amenées à **changer**, il est important d'en avoir conscience et c'est pourquoi nous allons parcourir ensemble certaines de ces règles de base pour la version **4.5.0** du Framework.

● Le corps de page

À partir du moment où vous faites la liaison entre votre **fichier HTML** et **Bootstrap**, votre **body** prend les propriétés CSS suivantes :

```
body {  
  margin: 0;  
  font-family: -apple-system, BlinkMacSystemFont, "Segoe UI", Roboto, "Helvetica Neue", Arial,  
    "Noto Sans", sans-serif, "Apple Color Emoji", "Segoe UI Emoji", "Segoe UI Symbol", "Noto Color Emoji";  
  font-size: 1rem;  
  font-weight: 400;  
  line-height: 1.5;  
  color: #212529;  
  text-align: left;  
  background-color: #fff;  
}
```

● Les paragraphes

Les paragraphes ont, par défaut, le style suivant (juste une marge en bas) :

```
p {  
  margin-top: 0;  
  margin-bottom: 1rem;  
}
```

● Les titres

Les titres ont, par défaut, tous les styles suivants :

```
h1, h2, h3, h4, h5, h6,  
{  
  font-family: inherit;  
  font-weight: 500;  
  line-height: 1.1;  
  color: inherit;  
}
```

Et chaque niveau de titre a ces valeurs par défaut :

```
h1{
  font-size: 2.5rem;
}
h2{
  font-size: 2rem;
}
h3{
  font-size: 1.75rem;
}
h4{
  font-size: 1.5rem;
}
h5{
  font-size: 1.25rem;
}
h6{
  font-size: 1rem;
}
```

Les marges des titres sont gérées indépendamment de h1 à h6 :

```
h1,h2,h3,h4,h5,h6{
  margin-top: 0;
  margin-bottom: 0.5rem;
}
```

Il faut aussi savoir que **Bootstrap** propose des noms de classes `.h1`, `.h2`, `.h3`, `.h4`, `.h5` et `.h6` qui ont exactement les mêmes propriétés CSS que les titres ci-dessus.

● Les mises en évidence

Contrairement au reset que l'on avait pour habitude d'utiliser, **Bootstrap** n'enlève rien au style par défaut des balises **strong**, **b**, **em** et **i**. Pour l'élément **small**, la règle stipule une taille de caractères de 80% :

```
small{
  font-size: 80%;
}
```

Bootstrap propose une classe `.small` qui fait exactement la même chose que le sélecteur ci-dessus.

● L'alignement du texte

Pour aligner du texte, **Bootstrap** a ses propres classes :

```
.text-left {
  text-align: left !important;
}
.text-right {
  text-align: right !important;
}
.text-center {
  text-align: center !important;
}
.text-justify {
  text-align: justify !important;
}
```

● Le texte coloré

Bootstrap a prévu un certain formatage pour le texte au niveau de sa couleur pour faire passer certaines informations du genre : pas important (**muted**), important (**primary**), succès (**success**), information (**info**), attention (**warning**) ou danger (**danger**) et les effets de hover qui vont avec.

```
.text-muted {
  color: #6c757d !important;
}
.text-primary {
  color: #007bff !important;
}
a.text-primary:hover, a.text-primary:focus {
  color: #0056b3 !important;
}
.text-success {
  color: #28a745 !important;
}
a.text-success:hover, a.text-success:focus {
  color: #19692c !important;
}
.text-info {
  color: #17a2b8 !important;
}
a.text-info:hover, a.text-info:focus {
  color: #0f6674 !important;
}
.text-warning {
  color: #ffc107 !important;
}
a.text-warning:hover, a.text-warning:focus {
  color: #ba8b00 !important;
}
.text-danger {
  color: #dc3545 !important;
}
a.text-danger:hover, a.text-danger:focus {
  color: #a71d2a !important;
}
```

Muted, primary, success, info, warning, danger

● Les adresses

Vous pouvez mettre en forme un bloc d'adresse avec les styles par défaut de **Bootstrap** suivants :

```
address {  
  margin-bottom: 1rem;  
  font-style: normal;  
  line-height: inherit;  
}
```

● Les listes à puces

Bootstrap ne modifie pas grand-chose au comportement par défaut des **listes à puces**, il uniformise juste les marges.

```
ul, ol {  
  margin-top: 0;  
  margin-bottom: 1rem;  
}
```

Pour avoir une liste sans puces (et sans retrait) :

```
.list-unstyled {  
  padding-left: 0;  
  list-style: none;  
}
```

Pour avoir une liste au format horizontal, la classe **.list-inline** existe :

```
.list-inline {  
  padding-left: 0;  
  list-style: none;  
}  
.list-inline-item {  
  display: inline-block;  
}  
.list-inline-item {  
  margin-right: 0.5rem;  
}
```

● L'affichage du code

Bootstrap a prévu des styles pour la balise **code** afin de mettre du code HTML en page :

```
code {  
  
  font-size: 87.5%;  
  color: #e83e8c;  
  word-wrap: break-word;  
}
```


LE CSS POUR LES FORMULAIRES

Bootstrap a fait un travail considérable sur la mise en forme des **formulaires**. Voici les règles principales qui pourraient vous être utiles.

• La structure de base

Pour profiter des styles de Bootstrap, vous devez englober votre label et son input/textarea dans un div qui aura la classe **form-group** alors que l'input de type textuel aura la classe **form-control**.

Voici les styles appliqués à ces classes :

```
.form-control {
  display: block;
  width: 100%;
  height: calc(1.5em + 0.75rem + 2px);
  padding: 0.375rem 0.75rem;
  font-size: 1rem;
  font-weight: 400;
  line-height: 1.5;
  color: #495057;
  background-color: #fff;
  background-clip: padding-box;
  border: 1px solid #ced4da;
  border-radius: 0.25rem;
  transition: border-color 0.15s ease-in-out, box-shadow 0.15s ease-in-out;
}

.form-group {
  margin-bottom: 1rem;
}
```

Avec une structure simple comme celle-ci :

```
<form>
  <div class="form-group">
    <label for="name">Nom</label>
    <input type="text" class="form-control" id="name">
  </div>
  <div class="checkbox">
    <label>
      <input type="checkbox"> Accepter les conditions
    </label>
  </div>
  <button type="submit" class="btn btn-default">Envoyer</button>
</form>
```

On obtient un résultat pré-formaté comme ceci :

Nom

☐ Accepter les conditions

Envoyer

• Les formulaires en ligne

Si vous mettez la classe **.form-inline** sur la balise form, vous obtiendrez ce résultat-ci :

Nom ☐ Accepter les conditions

Plutôt pratique !

• D'autres exemples

Les champs actifs dans bootstrap sont stylisés de cette façon, sans rien ajouter :

Nom

La classe **.border-success** sur l'input du formulaire change le rendu comme ceci :

Nom

Pour la classe **.border-warning** :

Nom

Pour la classe **.border-error** :

Nom

• Les boutons

Dans Bootstrap, une attention toute particulière a été accordée aux styles des boutons.

```
.btn {
  display: inline-block;
  font-weight: 400;
  color: #212529;
  text-align: center;
  vertical-align: middle;
  -webkit-user-select: none;
  -moz-user-select: none;
  -ms-user-select: none;
  user-select: none;
  background-color: transparent;
  border: 1px solid transparent;
  padding: 0.375rem 0.75rem;
  font-size: 1rem;
  line-height: 1.5;
  border-radius: 0.25rem;
  transition: color 0.15s ease-in-out, background-color 0.15s ease-in-out, border-color 0.15s ease-in-out, box-shadow 0.15s ease-in-out;
}
```

Les boutons sont donc bien stylisés dans Bootstrap et on peut leur appliquer différentes variantes grâce à la panoplie suivante et leurs classes (**.btn-default**, **.btn-primary**, **.btn-success**, etc.).



On retrouve donc les mêmes noms de classe que celles déjà présentes pour mettre en page le texte, logique !

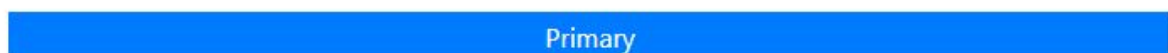
Chaque bouton a son effet de survol et de focus.

On peut également modifier la taille des boutons en ajoutant une classe **.btn-lg** (long), **.btn-sm** (small).

Ici 3 boutons, avec en premier le bouton par défaut et ensuite du plus grand au plus petit :



On peut aussi transformer un bouton en balise bloc avec la classe **.btn-block** :



LE CSS POUR LES IMAGES

Bootstrap a quelques petits trucs pour nos images également.

● Par défaut

Sur nos balises IMG, par défaut, Bootstrap applique les styles suivants :

```
img {  
  border: 0;  
}  
img {  
  max-width: 100% !important;  
}  
img {  
  vertical-align: middle;  
}
```

Alignement vertical automatique, pas de bordure et les images sont prêtes pour le responsive design, cool !

● Arrondie

Une simple classe **.rounded** suffira pour arrondir les angles de nos images :



● Inscrite dans un cercle / oval

Une simple classe **.rounded-circle** permet d'inscrire votre image dans un oval (si votre image est rectangulaire) ou un rond si elle est carrée ☺.



● Adaptative

Avec la classe **.img-fluid**, on dit à l'image de se comporter comme un élément de type block, d'adapter sa hauteur en fonction d'une largeur qu'on met en max-width : 100%.

```
.img-fluid
{
  display: block;
  max-width: 100%;
  height: auto;
}
```

LE CSS POUR LA MISE EN PAGE

Tout a été également prévu pour jouer avec les **float** dans Bootstrap, mais pas que.

● Le flottement des blocs

Les classes **.float-right** et **.float-left** servent à flotter à droite ou à gauche des éléments de type **block**.

```
.float-right {  
  float: right !important;  
}  
.float-left {  
  float: left !important;  
}
```

Pour contrer les problèmes de float, Bootstrap utilise la classe **.clearfix** (avec injection d'une balise dans le code html).

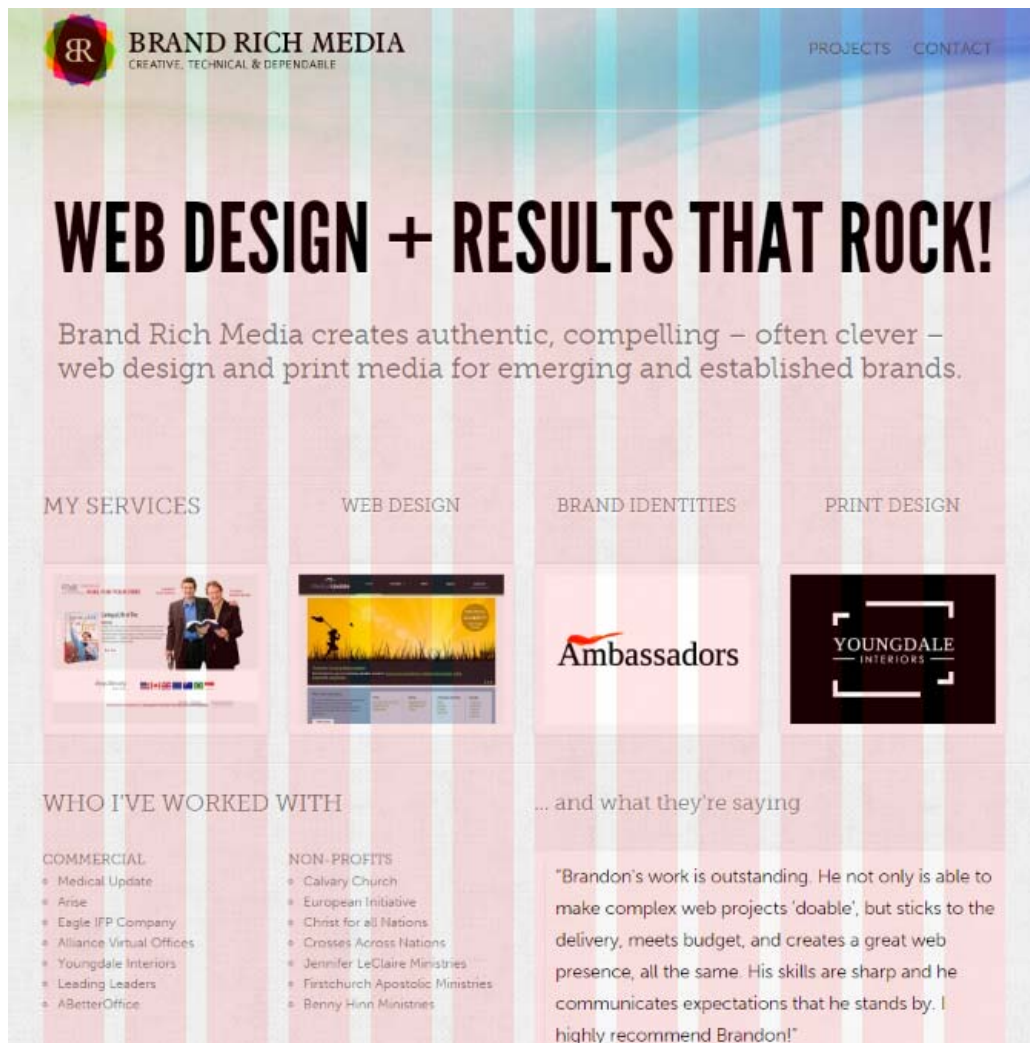
Historiquement, les éléments flottants ont été utilisé pour faire de la mise en page web. Pourtant, ils étaient destinés à permettre le flottement d'une image à droite ou à gauche d'un texte.

A partir de la **version 4** et donc aussi pour la **version 5** de **Bootstrap**, ce sont les **Flexbox** qui sont utilisés par le Framework pour obtenir une mise en page **Responsive**.

LA GRILLE ADAPTATIVE

C'est le cœur de Bootstrap. C'est ici que Bootstrap prend encore plus de sens puisque si votre design a suivi les canvas d'une **grille de mise en page**, Bootstrap va vous aider à facilement rendre votre site Responsive.

De manière générale, les grilles de mise en page pour le Web ont une largeur maximale de 960 pixels et sont composées de **12** à 16 colonnes.



Un exemple de site basé sur une grille en 12 colonnes.

• Les paliers et le conteneur principal

Au niveau responsive design, Bootstrap travaille sur **plusieurs paliers**, ce sont les **breakpoints** :

- **sm** pour **small** est pour les petits écrans dont la **largeur** est **supérieure ou égale** à **576** pixels.
- **md** pour **médium** est pour les écrans moyens dont la **largeur** est **supérieure ou égale** à **768** pixels.
- **lg** pour **large** est pour les grands écrans dont la **largeur** est **supérieure ou égale** à **992** pixels
- **xl** pour **extra large** est pour les écrans dont la **largeur** est **supérieure ou égale** à **1200** pixels.
- **xxl** pour **extra extra large** est pour les écrans dont la **largeur** est **supérieure ou égale** à **1400** pixels.

Pour les très petits écrans, inférieurs à **576** pixels, il n'est pas nécessaire d'indiquer une abréviation de taille, car c'est l'option par défaut en vertu de l'orientation spécifique à **Bootstrap 4** et **5: Mobile First**. On trouve cependant encore l'abréviation **xs** pour ce **breakpoint**.

Ces règles se trouvent dans le fichier **bootstrap.css**. Généralement, vous allez avoir un conteneur principal et pour Bootstrap celui-ci doit avoir la classe **.container**

```
.container {  
  width: 100%;  
  padding-right: 15px;  
  padding-left: 15px;  
  margin-right: auto;  
  margin-left: auto;  
}  
  
@media (min-width: 576px) {  
  .container {  
    max-width: 540px;  
  }  
}  
  
@media (min-width: 768px) {  
  .container {  
    max-width: 720px;  
  }  
}  
  
@media (min-width: 992px) {  
  .container {  
    max-width: 960px;  
  }  
}  
  
@media (min-width: 1200px) {  
  .container {  
    max-width: 1140px;  
  }  
}
```

Cet élément qui aura la classe **.container** s'adapte déjà parfaitement à toutes les tailles d'écran et en plus contient déjà du **padding** et des **margin**.

• La largeur des boîtes avec Bootstrap

Il est extrêmement important de savoir que Bootstrap utilise la propriété **box-sizing :border-box** dans son calcul de largeur des boîtes.

Ça veut donc dire qu'à la largeur (**width**) vont venir s'ajouter la valeur des **padding** de **gauche** et de **droite** mais aussi l'épaisseur des **bordures**.

● L'utilisation des lignes et de la class .row

Pour mieux organiser votre mise en page en ligne, **Bootstrap** préconise d'utiliser une boîte utilisant la classe **.row**.

Il faut donc penser votre mise en page sous forme de plusieurs lignes. Chaque ligne va comporter les boîtes disposées sur les colonnes de la grille.

Voici ce que fait la classe **.row** :

```
.row {  
  display: -ms-flexbox;  
  display: flex;  
  -ms-flex-wrap: wrap;  
  flex-wrap: wrap;  
  margin-right: -15px;  
  margin-left: -15px;  
}
```

Les marges négatives sont simplement là pour que les padding qui seront ajoutés dans les colonnes ne « doublent » pas la **gouttière** de **15 pixels**.

● La largeur des colonnes

Pour gérer la largeur des colonnes dans la grille adaptative, Bootstrap utilise un système de classes dédiées défini dans le fichier **bootstrap.css**.

Chaque règle **@media** utilise son propre préfixe pour les classes CSS à utiliser :

Taille écran	@media	Préfixe des CSS
Smartphone (Small devices)	@media (min-width :576px)	.col-sm-*
Tablette (Medium devices)	@media (min-width :768px)	.col-md-*
Large (Large devices)	@media (min-width :992px)	.col-lg-*
Extra Large(Extra Large devices)	@media (min-width :1200px) min-width :1400px	.col-xl-* .col-xxl-*

Le suffixe ***** correspond au nombre de colonnes sur lequel doit être placée la classe.

Par exemple, si nous voulons qu'un div prenne la place de 4 colonnes en taille moyenne (md), on lui attribuera la classe : **.col-md-4**.

Le tout en image, pour une grille en 12 colonnes en taille moyenne :

.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1
.col-md-8								.col-md-4			
.col-md-4				.col-md-4				.col-md-4			
.col-md-6						.col-md-6					

blog.webnet.fr

Toutes les largeurs des colonnes sont définies en % de leur parent (souvent l'élément qui a la classe **.row**). Ces colonnes sont **flexibles** et sont en position relative par défaut. Elles ont également un padding à gauche et à droite de 15px pour récupérer une gouttière de 30px.

• Les blocs décalés

Bootstrap vous permet de décaler des blocs ou créer de l'espace de la taille de **X colonnes**. Et ceci pour les différentes résolutions d'écran, vous aurez de façon générique la classe : **offset-{sm,md,lg,xl,xxl}-X**

Si vous voulez par exemple faire commencer votre bloc qui mesure **4 colonnes** après l'équivalent en largeur de **3 colonnes**, vous écrirez pour une large résolution d'écran (**lg**) :

```
<div class="container">
  <div class="row">
    <div class="col-lg-4 offset-lg-3">...</div>
  </div>
</div>
```

Dans ce cas, vous devez évidemment prendre en compte l'offset que vous avez indiqué dans le « compte » de vos 12 colonnes. Ce bloc vaut en réalité **7 colonnes, offset y compris**.

• Afficher et masquer des éléments

En responsive design, il arrive souvent qu'on affiche certains blocs dans les résolutions plus larges mais qu'on s'en passe dans les résolutions plus étroites.

Bootstrap dans sa version **4** a pensé à ce genre de choses et des classes sont à votre disposition pour répondre à votre demande.

Pour **masquer** les colonnes en fonction de la largeur de l'écran, utilisez la **d-none** classe ou l'une des **d-{sm,md,lg,xl,xxl}-none** classes.

Pour **afficher** des colonnes sur certaines tailles d'écran, combinez les classes mentionnées ci-dessus avec des classes **d-block** ou **d-{sm,md,lg,xl,xxl}-block**.

COMPOSANTS PRÉ-FABRIQUÉS

Bootstrap regorge de composants (<http://getbootstrap.com/components/>) prêts à l'emploi pour :

- Des dropdowns (<http://getbootstrap.com/components/#dropdowns>)
- Des boutons (<http://getbootstrap.com/components/#btn-groups>)
- Des boutons avec dropdown (<http://getbootstrap.com/components/#btn-dropdowns>)
- Des champs input (<http://getbootstrap.com/components/#input-groups>)
- Des systèmes de navigation (<http://getbootstrap.com/components/#nav>)
- Des barres de navigation (<http://getbootstrap.com/components/#navbar>)
- Des systèmes de breadcrumbs ou Fil d'Ariane (<http://getbootstrap.com/components/#breadcrumbs>)
- Des systèmes de pagination (<http://getbootstrap.com/components/#pagination>)
- Des « badges » (<http://getbootstrap.com/components/#badges>)
- Des blocs d'introduction (<http://getbootstrap.com/components/#jumbotron>) n'existe plus en **version 5**
- Un bloc de header (<http://getbootstrap.com/components/#page-header>)
- Des vignettes (<http://getbootstrap.com/components/#thumbnails>)
- Des alertes (<http://getbootstrap.com/components/#alerts>)
- Des barres de progression (<http://getbootstrap.com/components/#progress>)
- Des objets média (<http://getbootstrap.com/components/#media>)
- Des groupes de listes (<http://getbootstrap.com/components/#list-group>)
- Des embed responsive (<http://getbootstrap.com/components/#responsive-embed>)
- Des Cards (<https://getbootstrap.com/docs/4.5/components/card/>)
- ...

Il serait un peu inutile de les détailler ici tellement ils sont bien représentés sur la page des composants de **Bootstrap**.

PLUGINS JAVASCRIPT

Plein de plugins sont aussi inclus à l'intérieur de Bootstrap. Pour les rendre actifs avec les **versions de Bootstrap inférieure à 5**, n'oubliez pas d'incorporer la **bibliothèque JQuery** avant d'appeler le fichier **bootstrap.js** qui est le fichier js qui regroupe tous les plugins de Bootstrap.

A partir de la **version 5** du Framework ce n'est plus nécessaire, vous pouvez directement utiliser le fichier : **bootstrap.bundle.min.js**

Nous allons voir quels genres de plugins sont disponibles avec **Bootstrap** mais je vous renvoie d'ores et déjà vers leur site officiel pour voir comment les mettre en œuvre (<http://getbootstrap.com/javascript/>).

• Tooltips

Des « **infobulles** » sont utilisables dans Bootstrap. On les active en passant par-dessus un élément et il est possible de les faire apparaître à gauche, à droite, en bas ou en haut d'un élément. Voici à quoi ces tooltips ressemblent :

EXAMPLE

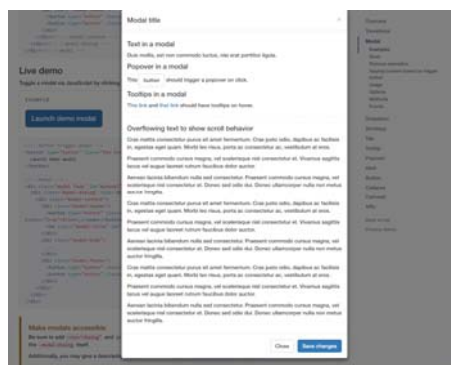


Pour les voir en action et comprendre quelle structure vous devez mettre en place pour les utiliser, rendez-vous ici : <http://getbootstrap.com/javascript/#tooltips>

La bibliothèque **popper.js** offre par ailleurs un grand choix d'**infobulles** et autre **popover**, elle est désormais intégrée au Framework **Bootstrap** dans le script **bootstrap.bundle.js** (<https://popper.js.org/>)

• Fenêtres modales

Il est possible de faire apparaître des **fenêtres modales** au clic avec Bootstrap. Elles ressemblent à ceci :



Elles apparaissent et disparaissent avec une transition et vous pouvez la fermer en cliquant sur la croix. Il y a pas mal de paramètres sur lesquels vous pouvez jouer pour personnaliser cette fenêtre et ses transitions. Toute la doc est ici : <http://getbootstrap.com/javascript/#modals>

• Scrollspy

Vous pouvez facilement mettre en place un effet de **scrollspy** qui suit votre navigation au fur et à mesure que vous parcourez une page. L'utilisation du scrollspy demande d'utiliser le composant « nav » de Bootstrap pour bien fonctionner.

Toutes les infos sont ici : <http://getbootstrap.com/javascript/#scrollspy>

• Les accordéons

Vous avez toute une série d'accordéons pré-écrits pour vous avec Bootstrap. Certains qui se déroulent suite à un clic sur un bouton :

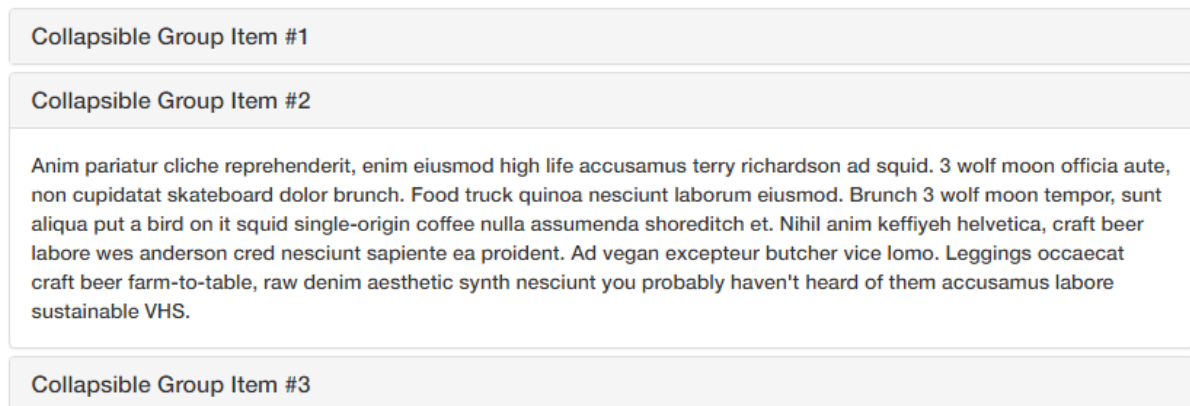
EXAMPLE



Anim pariatur cliche reprehenderit, enim eiusmod high life accusamus terry richardson ad squid. Nihil anim keffiyeh helvetica, craft beer labore wes anderson cred nesciunt sapiente ea proident.

d'autres qui s'ouvrent dans un bloc d'accordéon classique :

EXAMPLE



Pour en savoir plus : <http://getbootstrap.com/javascript/#collapse>

● Les carrousels d'images

Il en existe un grand nombre et une grande variété sur Internet et Bootstrap a cru bon d'en intégrer un aussi dans ses plugins par défaut. Il est paramétrable et personnalisable sur plusieurs points et voici à quoi il ressemble :

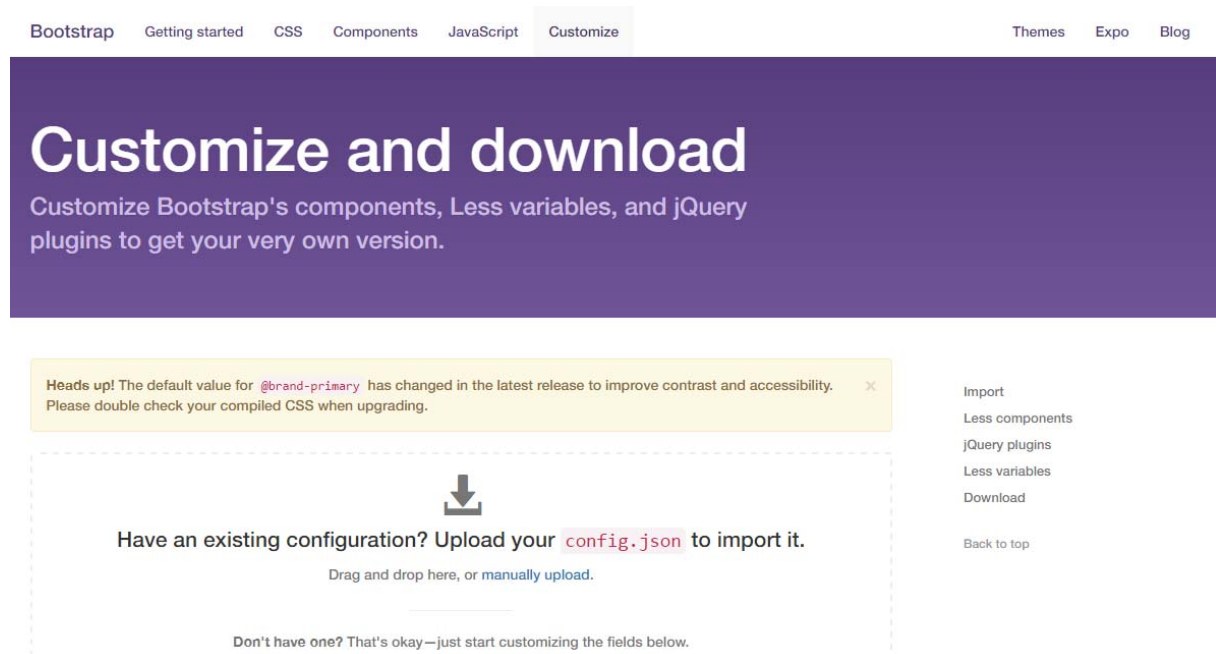


Pour un savoir plus : <http://getbootstrap.com/javascript/#carousel>

PERSONNALISEZ VOTRE EXPÉRIENCE

Chez **Bootstrap**, ils ont pensé à tout : vous avez le loisir de n'embarquer avec vous, dans votre projet, que ce dont vous avez strictement besoin à la réalisation complète de votre site.

Pour ce faire, ils ont créé un petit onglet « customize » dans leur menu.



Cet outil vous permet de choisir uniquement les « paquets » dont vous avez besoin et ça, c'est cool !

Common CSS

- ☒ Print media styles
- ☒ Typography
- ☒ Code
- ☒ Grid system
- ☒ Tables
- ☒ Forms
- ☒ Buttons
- ☒ Responsive utilities

Components

- ☒ Glyphicons
- ☒ Button groups
- ☒ Input groups
- ☒ Navs
- ☒ Navbar
- ☒ Breadcrumbs
- ☒ Pagination
- ☒ Pager
- ☒ Labels
- ☒ Badges
- ☒ Jumbotron
- ☒ Thumbnails
- ☒ Alerts
- ☒ Progress bars
- ☒ Media items
- ☒ List groups
- ☒ Panels
- ☒ Responsive embed
- ☒ Wells
- ☒ Close icon

JavaScript components

- ☒ Component animations (for JS) (includes Collapse)
- ☒ Dropdown
- ☒ Tooltip
- ☒ Popover
- ☒ Modal
- ☒ Carousel