

Bootstrap

Objectifs

- ✔ Améliorer l'intégration et le temps de conception d'un site web à l'aide de bootstrap.

01

Un framework CSS, qu'est-ce que c'est ?

Un framework est avant tout un cadre de travail permettant de faire profiter aux développeurs d'une arborescence et d'une architecture communes.

Le plus souvent cela se présente sous forme de dossier pour faciliter l'organisation du code et simplifier la maintenance dans le temps.

Un framework peut aussi apporter un socle de fonctionnalités par défaut.

Qu'est-ce qu'on obtient concrètement lorsqu'on télécharge bootstrap ? juste un dossier contenant quelques fichiers css et js.

02

Bootstrap, c'est quoi ?



Bootstrap est un framework CSS permettant de faciliter l'écriture de la mise en forme CSS d'un site web.

Avec Bootstrap, vous gagnerez du temps et n'aurez plus besoin de vous prendre la tête sur un décalage de quelques pixels sur votre page web. Tout le CSS a déjà été pensé pour vous.

Plus généralement, nous pouvons dire que c'est un rassemblement de différents morceaux de code utiles à la création d'une page web, aussi bien dans la mise en forme CSS que dans l'animation et l'interactivité du contenu JavaScript.

03

Naissance de Bootstrap

Bootstrap a été écrit par deux développeurs Twitter pour répondre à des problématiques communes et gagner du temps dans la mise en forme des pages web.

Mark Otto et Jacob Thornton ont écrit bootstrap aux alentours de l'année 2010 puisque le projet a été rendu OpenSource en 2011.

Cela permet notamment de ne pas devoir recommencer toute l'écriture du code de zéro à chaque nouvelle page web.

04

Avantages de Bootstrap

« Bootstrap, j'en entends parler partout, okay mais quel est son intérêt ? je suis très bien sans lui, pourquoi je devrai l'utiliser ? »

Selon moi, les principaux avantages de bootstrap sont :

1. **Responsive** : Vous obtiendrez un site web responsive de base, ce qui n'est pas négligeable, cela est même indispensable de nos jours avec les smartphones et tablette !
2. **Un site web jolie** : Il existe plusieurs thèmes prêts faits de haute qualité et une multitude de classes CSS que l'on peut utiliser !
3. **Un gain de temps** : Vous ne vous occupez quasiment plus que d'écrire le code HTML et d'appeler des classes CSS pré-existantes. Il devient très rapide de monter la mise en forme d'une page web.
4. **Compatibilité** : A priori, la présentation visuelle sera la même sur tous les navigateurs.
5. **Graphisme** : Lorsqu'on développe en programmation, cela nous permet d'avoir une apparence de site sans écrire une seule ligne de CSS (à l'origine, bootstrap a justement été créé pour simplifier le travail de présentation visuelle aux développeurs côté back). Cela est une aubaine pour les développeurs qui préfèrent la partie fonctionnelle à la partie graphique

Naturellement, cette liste n'est pas exhaustive, il y a d'autres avantages, comme notamment l'intégration de plugin jquery, ce qui permet de gagner également du temps sur la partie JavaScript.

05

Bootstrap : pour ou contre ?



➤ Question « attend, si bootstrap fait tout, à quoi ça sert de connaître le langage CSS ? »

C'est vrai que Bootstrap apporte beaucoup sur la partie CSS, mais il est néanmoins quasi indispensable de disposer d'un minimum de connaissances techniques dans les métiers de l'intégration HTML / CSS pour pouvoir s'en servir ou encore apporter des modifications supplémentaires.



➤ Question « Bootstrap, tu me dis que c'est un super outil avec des thèmes pré-conçus okay mais ni moi ni mes clients veulent le même site que tout le monde ! »

C'est vrai, selon les projets (ou disons le budget et l'exigence du client final) parfois il est pratique d'installer un thème prêt fait et d'autres fois il faut partir de zéro from scratch sur la création d'un thème personnalisé.

Quoi qu'il en soit, dans bootstrap, il reste possible d'ajouter son propre code CSS pour profiter d'une base thème graphique et de le personnaliser selon ses envies.



➤ Question « attend, tu me dis que ça va aller vite d'écrire le site mais où est-ce que je mets mon CSS dans bootstrap ? ».

Généralement, nous utilisons les classes de bootstrap existantes et ajoutons une feuille de style supplémentaire pour ajouter notre propre code CSS.



➤ Question « Malgré les modifications possibles, je n'aime pas qu'on m'impose une structure ou une vision du code qui n'est pas la mienne ».

Il est important de suivre les évolutions et au moins de les "tester", même si l'on retrouve cela de plus en plus sur les offres d'emploi, utiliser bootstrap n'est pas obligatoire !

Pour les puristes qui ne sont pas très joyeux à l'idée d'utiliser bootstrap en FRONT, ils pourront toujours l'utiliser en BackOffice, et oui en BackOffice, ce n'est pas un problème d'avoir un design "type". Ainsi l'administrateur pourra gérer son site via son smartphone (du fait que bootstrap apporte la partie responsive nativement).

Il n'y a pas de réel inconvénient à travailler avec bootstrap car il s'agit avant tout d'un choix.

Certains vous diront que bootstrap (framework, jquery (bibliothèque) ne sont pas des langages et viennent ralentir les temps d'exécution du site au niveau des performances.

Même s'il l'on minimise (coller le code sur 1 seule ligne) le code, ces personnes n'ont pas forcément tort mais par rapport à tout ce que cela apporte, il faut peser le pour et le contre, cela dépend du projet et du développeur.

Par exemple : est-on prêt à sacrifier un peu de performance contre un gain de temps lors de la création du code ?

L'apprentissage prend un peu de temps, mais ça, c'est comme tout le reste, il ne s'agit donc pas d'un inconvénient ! Apprendre c'est la vie du développeur !

Vous l'aurez compris, bootstrap apporte plusieurs choses.

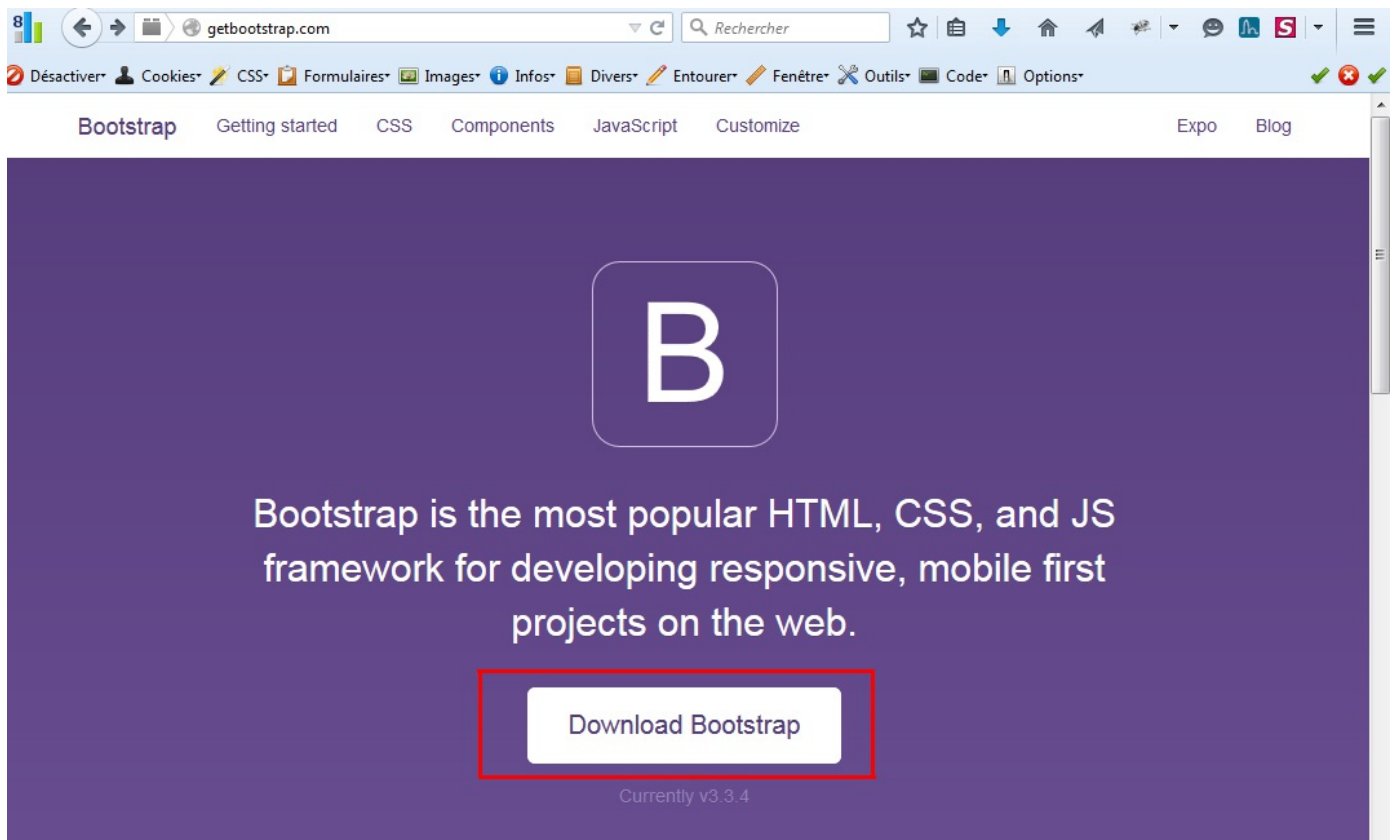
Avant tout : une feuille de style CSS contenant des propriétés et valeur de base pour tous les éléments HTML.

Cela permet de bénéficier d'un graphisme standardisé pour tous les types d'éléments habituels d'un site à un autre : icônes, liens, listes, tableaux, formulaires, etc.

Bootstrap est régulièrement mis à jour, vous pourrez donc suivre les dernières actualités sur [GitHub](#) ou sur le [blog officiel](#).

Pour se servir de bootstrap, il faut avant tout le télécharger.

Nous allons donc nous rendre sur le [site officiel de bootstrap pour le télécharger](#).



Ensuite, plusieurs choix s'offrent à nous :

Bootstrap

Compiled and minified CSS, JavaScript, and fonts. No docs or original source files are included.

[Download Bootstrap](#)

Source code

Source Less, JavaScript, and font files, along with our docs.

Requires a Less compiler and some setup.

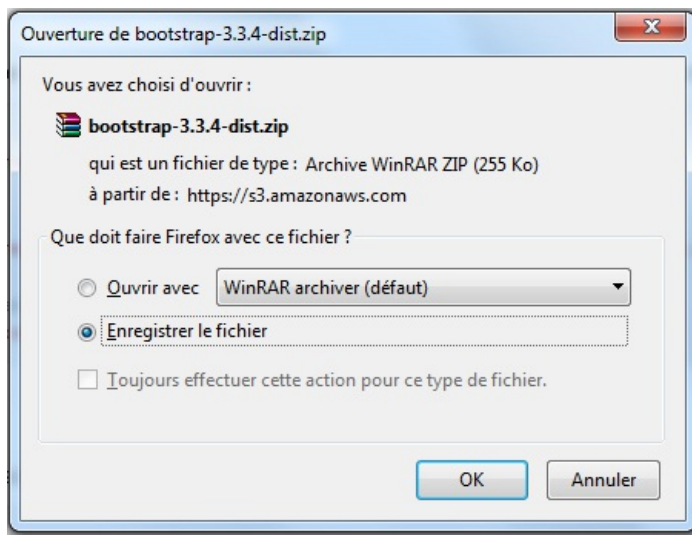
[Download source](#)

Sass

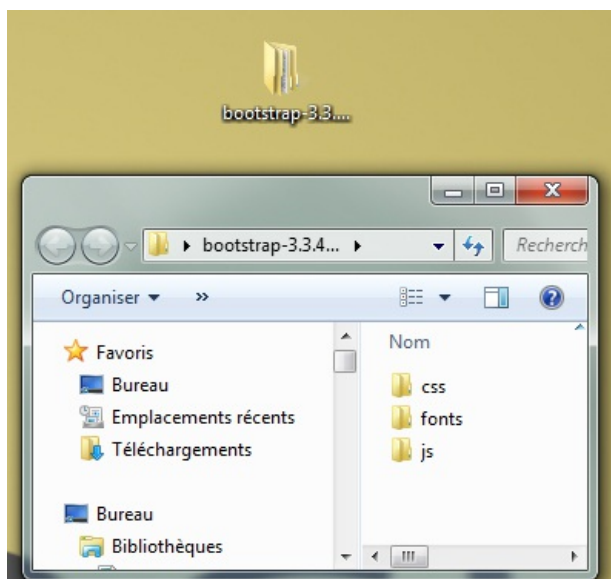
[Bootstrap ported from Less to Sass](#) for easy inclusion in Rails, Compass, or Sass-only projects.

[Download Sass](#)

- **Download Bootstrap** correspond à une première utilisation de base.
Les fichiers CSS, JS et les polices sont minifiés. Aucun document source n'est inclu.
- **Download source** permet d'avoir accès à toutes les sources de bootstrap.
Source Less, JavaScript et les fichiers police, avec les documents d'origine. LESS nécessite une configuration et un compilateur.
- **Download Sass** permet de ...
Version de Bootstrap porté du Less au Sass pour une inclusion facile dans Rails, Compass, ou Sass.



Lorsque vous télécharger Bootstrap, cela se présente sous cette forme :

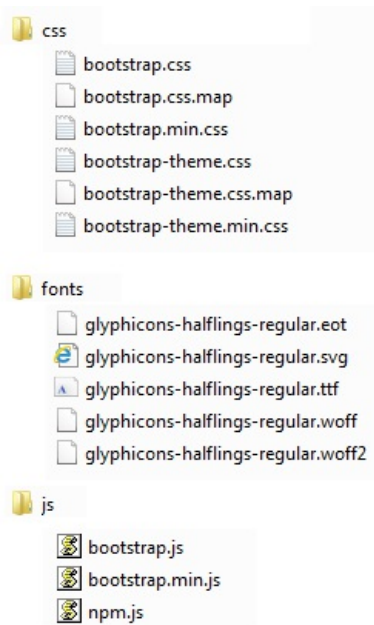


Un dossier **css** pour la mise en forme.

Un dossier **js** pour l'animation / interactivité.

Un dossier **fonts** pour les polices / icônes.

Le mot clé "dist" indique qu'il s'agit d'une distribution. Ce terme est utilisé régulièrement en Informatique.



- **bootstrap.css** représente le fichier principal de bootstrap avec la déclaration de toutes les classes CSS
- **bootstrap.css.map** ce fichier permet de retrouver la position d'une ligne de code dans la version normale à partir du code minifié
- **bootstrap.min.css** Même contenu que le fichier bootstrap.css de manière mimifié (tout le code sur une ligne pour gagner en performances).
- **bootstrap-theme.css** représente le fichier de bootstrap permettant de créer des thèmes spécifiques
- **bootstrap-theme.css.map** ce fichier permet de retrouver la position d'une ligne de code dans la version normale à partir du code minifié
- **bootstrap-theme.min.css** Même contenu que le fichier bootstrap-theme.css de manière mimifiée (tout le code sur une ligne pour gagner en performances).
- **glyphicons-halflings-regular.eot** icône au format EOT
- **glyphicons-halflings-regular.svg** icône au format SVG
- **glyphicons-halflings-regular.ttf** icône au format TTF
- **glyphicons-halflings-regular.woff** icône au format WOFF
- **glyphicons-halflings-regular.woff2** icône au format WOFF2
- **bootstrap.js** contient le code javascript de bootstrap
- **bootstrap.min.js** Même contenu que le fichier bootstrap.js de manière mimifié (tout le code sur une ligne pour gagner en performances).
- **npm.js** est un fichier généré automatiquement pouvant être utile dans l'environnement CommonJS

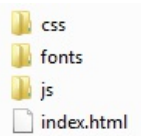


➤ **Attention**

Si des changements doivent se faire, nous créerons une feuille de style à part pour ne pas modifier la feuille de style CSS d'origine de bootstrap.

Pour créer votre page web avec bootstrap vous n'êtes pas obligé de garder le nom d'origine et pouvez renommer le dossier `/bootstrap-3.3.4-dist/` en `/monsie/`.

Ensuite, vous pouvez créer votre page web à la racine, par exemple : `index.html` (index correspond au nom de la page d'accueil).



Voici le code Html à prévoir pour notre page d'accueil :

```
index.html

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/style.css">
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body>
    ...
  </body>
</html>
```

Et nous voilà relié à bootstrap en seulement 1 ligne ! `css/bootstrap.css` (si cela ne fonctionne pas, pensez à bien revoir le chemin du fichier).

Nous ajoutons également un lien vers une feuille de style CSS `style.css` pour pouvoir agir à tout moment sur le code CSS.

Il sera donc important de créer le fichier correspondant dans le dossier `/css/` à côté des fichiers de bootstrap, tout comme nous l'aurions fait si nous n'étions pas accompagnés de bootstrap !

Concernant la ligne viewport, on demande que l'affichage occupe tout l'espace disponible avec une taille de 1, autrement dit sans zoom. L'attribut « viewport » permet de préciser au navigateur quelle taille il doit prendre pour afficher une page web.

`width=device-width` permet de régler la largeur de la page web sur la largeur de la fenêtre de l'appareil qui consulte actuellement la page

`initial-scale=1` permet de régler le niveau de zoom sur 100 % (par défaut).

Si vous souhaitez incorporer la partie JavaScript, vous aurez besoin de quelques lignes complémentaires :

```
index.html

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/style.css">
    <script src="http://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="js/bootstrap.js"></script>
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body>
    ...
  </body>
</html>
```

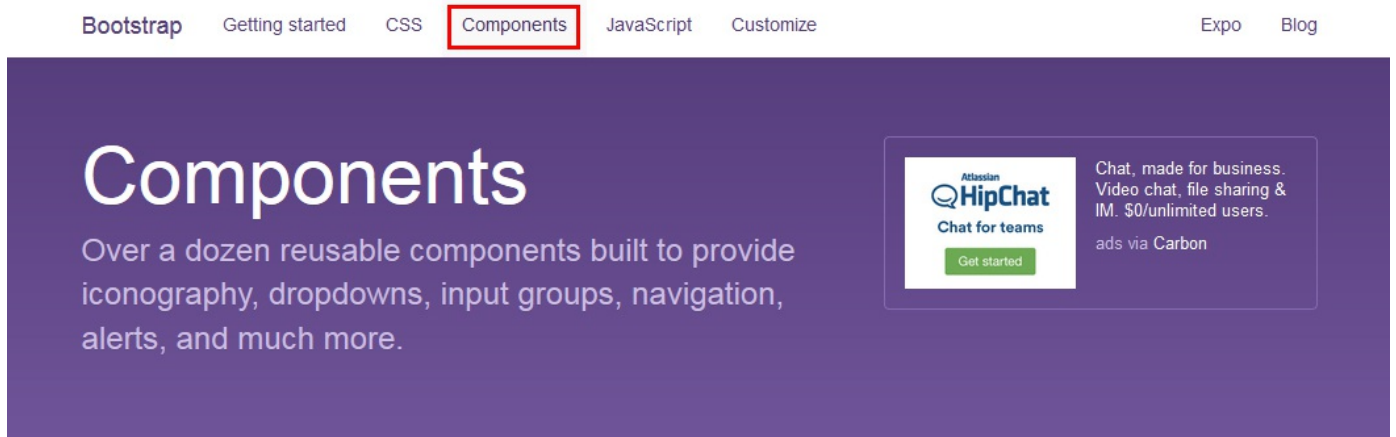
Vous devez donc forcément intégrer la librairie jquery avant le fichier `bootstrap.js` (car `bootstrap.js` se sert de JQUERY) sinon vous aurez une belle erreur exception dans votre console JavaScript.

Et maintenant ?

Nous pouvons déclarer des balises et éléments Html comme d'habitude. Mais maintenant que vous êtes relié à bootstrap, l'idéal serait d'appeler des classes CSS propres à Bootstrap.

Pour cela, il faut les connaître (quelques-unes, pas par cœur ;-)), et on a de la chance car la [documentation officielle de bootstrap](#) est très claire sur le sujet !

Je vous invite donc à me rejoindre sur cette [page](#) :



Glyphicons

Available glyphs

Includes over 250 glyphs in font format from the Glyphicon Halflings set. [Glyphicons](#) Halflings are normally not available for free, but their creator has made them available for Bootstrap free of cost. As a thank you, we only ask that you include a link back to [Glyphicons](#) whenever possible.

 glyphicon glyphicon- asterisk	 glyphicon glyphicon-p- lus	 glyphicon glyphicon-e uro	 glyphicon glyphicon-e ur	 glyphicon glyphicon- minus	 glyphicon glyphicon-cl oud	 glyphicon glyphicon- envelope	 glyphicon glyphicon- pencil
 glyphicon glyphicon-g lass	 glyphicon glyphicon- music	 glyphicon glyphicon- search	 glyphicon glyphicon-h eart	 glyphicon glyphicon-s tar	 glyphicon glyphicon- star-empty	 glyphicon glyphicon-u ser	 glyphicon glyphicon-fil m

Glyphicons
Dropdowns
Button groups
Button dropdowns
Input groups
Navs
Navbar
Breadcrumbs
Pagination
Labels
Badges
Jumbotron
Page header
Thumbnails
Alerts




Faisons un test rapide avec les icônes :

```
index.html

<!doctype html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/style.css">
    <script src="http://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="js/bootstrap.js"></script>
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body>
    <span class="glyphicon glyphicon-star"></span>
  </body>
</html>
```

Le nom de la classe `glyphicon glyphicon-star` ce n'est pas moi qui l'ai inventé, je suis comme vous, j'ai été le chercher dans la doc officielle de bootstrap. ;-).

Le Résultat

 Le résultat avec bootstrap nous donne une image en forme d'étoile sans l'utilisation de la balise ``

Ca fonctionne ? Si vous avez pu utiliser une classe CSS, vous pourrez utiliser toutes les autres !

Soyez curieux ! Essayer d'autres icônes avec les classes CSS `glyphicons`.

Vous n'avez presque plus besoin de moi ! Sauf qu'avant de vous lacher dans la nature, il faut absolument que je vous parle de la base de bootstrap : [le système de grille à 12 colonnes](#).

Système de grille

11

Apprendre Bootstrap et le système de grille 12 colonnes !

Bootstrap repose sur un système de grille à 12 colonnes. C'est l'un de ses fondements et on ne peut pas prétendre maîtriser Bootstrap sans connaître ce concept de base.

Nous allons avoir besoin de faire des tests pour l'expliquer. Je vous propose donc qu'à la racine de notre installation Bootstrap, nous allons créer un fichier nommé `grille.html`.

Voici la base à conserver tout le long de notre tuto :

grille.html

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/style.css">
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body>
    ...
  </body>
</html>
```

Nous préparons quelques bordures dans notre fichier `css/style.css` pour plus tard voir nos zones apparaître :

style.css

```
.container{ border: 1px solid; }
.bordureRouge{ border: 3px solid red; padding: 10px; }
.bordureVerte{ border: 3px solid green; padding: 10px; }
.bordureBleue{ border: 3px solid blue; padding: 10px; }
.bordureJaune{ border: 3px solid yellow; padding: 10px; }
.bordureOrange{ border: 3px solid orange; padding: 10px; }
```

C'est dans notre fichier personnel `style.css` que nous mettrons notre propre code, nous ne modifierons pas le fichier `bootstrap.css`, cela ne se fait pas dans les usages et d'autre part si l'on devait mettre à jour le fichier, cela ferait sauter nos modifications !

Bootstrap est donc avant tout un système de grille comportant 12 colonnes.

En général on centre son contenu sur la page web, cela se fait via la classe CSS Bootstrap `.container`. Une fois à l'intérieur de cette zone `.container` nous disposons de 12 emplacements.

Pour se l'imaginer, voici ce que cela donne en théorie :

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

Il faut donc voir cette grille comme des emplacements disponibles sur notre page web !
 C'est bien une grille, et nous pouvons prendre de la place à l'intérieur d'elle :

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

Pour prendre 1 place dans cette grille, nous pourrons utiliser la classe CSS (prédéfinie par bootstrap) `.col-md-1`
 Voici maintenant ce que cela donne en pratique avec le code :

grille.html

```

<!Doctype html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/style.css">
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body>
    <div class="container">
      <div class="row">
        <div class="col-md-1">
          ...
        </div>
      </div>
    </div>
  </body>
</html>
  
```

Si vous faites des tests, l'idéal serait d'ajouter une classe CSS supplémentaire pour personnaliser l'affichage, je vous invite à donner une couleur de fond ou mieux : une couleur de bordure. Exemple ci-dessous :

grille.html

```

<!Doctype html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
  
```

```
<link rel="stylesheet" href="css/style.css">
<meta name="viewport" content="width=device-width, initial-scale=1">
</head>
<body>
  <div class="container">
    <div class="row">
      <div class="col-md-1 bordureRouge">
        ...
      </div>
    </div>
  </div>
</body>
</html>
```

Décryptons ce code :

La classe container

La classe `.container` prédéfinie dans la feuille de style `bootstrap.css` (à laquelle nous sommes reliés) permet d'englober le reste de notre code et de le centrer sur la page web.

La classe row

La classe `.row` prédéfinie dans la feuille de style `bootstrap.css` (à laquelle nous sommes reliés) permet de prendre une ligne dans notre grille :

	col 1	col 2	col 3	col 4	col 5	col 6	col 7	col 8	col 9	col 10	col 11	col 12
row												
row												
row												

La classe col : col-md-1

col veut dire colonne.

Ce mot sera invariable dans bootstrap

md veut dire middle.

Les lettres correspondent au comportement des emplacements. nous le verrons un peu plus en détail plus bas dans ce tutorial.

Ce mot variable pourra prendre les valeurs suivantes : *xs, sm, md, lg*

1 veut dire 1 emplacement

Les chiffres expriment le nombre d'emplacements. nous le verrons également un peu plus en détail plus bas dans ce tutorial.

Cette partie variable pourra prendre les valeurs suivantes : 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

Si nous avions voulu prendre 2 emplacements, nous aurions pu le préciser de la sorte :

```
grille.html

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/style.css">
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body>
    <div class="container">
      <div class="row">
        <div class="col-md-2 bordureRouge">
          ...
        </div>
      </div>
    </div>
  </body>
</html>
```

`col-md-2` correspond donc à 2 emplacements

Résultat

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

Pour l'exemple, nous allons essayer de reproduire cette grille :

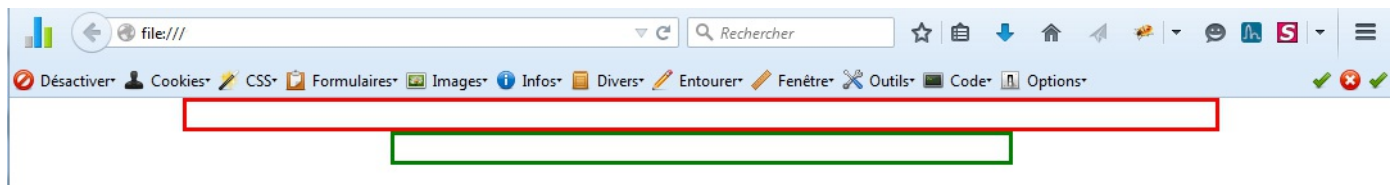
1	2	3	4	5	6	7	8	9	10	11	12

```

grille.html
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/style.css">
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body>
    <div class="container">
      <div class="row">
        <div class="col-md-10 col-md-offset-1 bordureRouge"> ... </div>
      </div>
      <div class="row">
        <div class="col-md-6 col-md-offset-3 bordureVerte">...</div>
      </div>
    </div>
  </body>
</html>

```

Résultat



Quelques explications :

1ère ligne

La classe `.row` permet de créer une nouvelle ligne.

La classe `.col-md-10` permet de créer un nouvel espace qui prend 10 places (sur les 12 possibles).

La classe `.col-md-offset-1` permet de créer une marge à gauche d'1 place.

Nous avons donc une zone s'étalant sur 10 emplacements centrés avec 1 emplacement à gauche et 1 à droite de marge (ce qui fait bien 12 au total).

Nous utilisons md pour dire middle, mais nous aurions également pu utiliser xs pour quelque chose de plus petit. Tout ça se trouve dans la doc!

La classe `.bordureRouge` permet de simplement de voir apparaître notre zone à l'écran entourée de rouge.

2ème ligne

Ensuite, nous ré-utilisons la classe `.row` pour créer une nouvelle ligne en dessous.

La classe `.col-md-6` permet de créer un nouvel espace qui prend 6 places (sur les 12 possibles).

La classe `.col-md-offset-3` permet de créer une marge à gauche de 3 places.

Nous avons donc une zone s'étalant sur 6 emplacements centrés avec 3 emplacements à gauche et 3 à droite de marge (ce qui fait bien 12 au total).

La classe `.bordureVerte` permet de simplement de voir apparaître notre zone à l'écran entourée de vert.

Allons plus loin avec ce schéma :

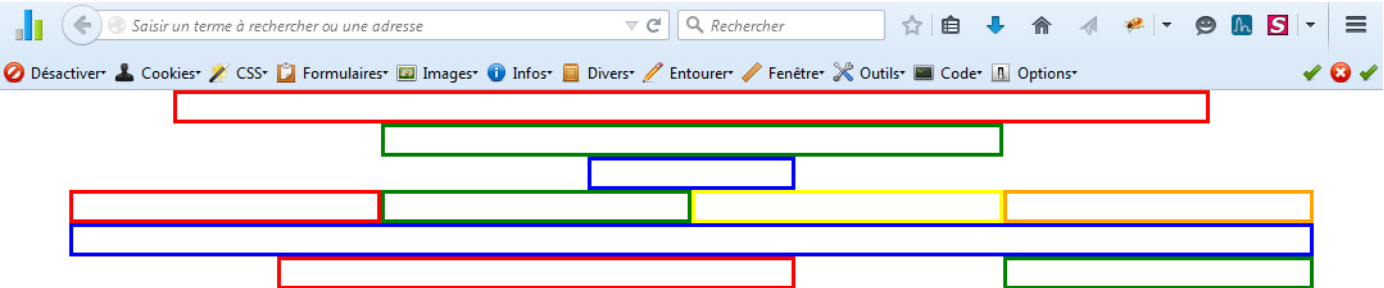
1	2	3	4	5	6	7	8	9	10	11	12

Si vous pensez avoir compris le principe et que souhaitez faire la suite en exercice pour vous entrainer, a vous de coder ! vous n'êtes pas obligé de regarder le code ci-dessous :

grille.html

```
<!Doctype html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/style.css">
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body>
    <div class="container">
      <div class="row"> <!-- exemple - ligne 1 -->
        <div class="col-md-10 col-md-offset-1 bordureRouge"></div>
      </div>
      <div class="row"> <!-- exemple - ligne 2 -->
        <div class="col-md-6 col-md-offset-3 bordureVerte"></div>
      </div>
      <div class="row"> <!-- exercice - ligne 3 -->
        <div class="col-md-2 col-md-offset-5 bordureBleue"></div>
      </div>
      <div class="row"> <!-- exercice - ligne 4 -->
        <div class="col-md-3 bordureRouge"></div>
        <div class="col-md-3 bordureVerte"></div>
        <div class="col-md-3 bordureJaune"></div>
        <div class="col-md-3 bordureOrange"></div>
      </div>
      <div class="row"> <!-- exercice - ligne 5 -->
        <div class="col-md-12 bordureBleue"></div>
      </div>
      <div class="row"> <!-- exercice - ligne 6 -->
        <div class="col-md-5 col-md-offset-2 bordureRouge"></div>
        <div class="col-md-3 col-md-offset-2 bordureVerte"></div>
      </div>
    </div>
  </body>
</html>
```

Le Résultat



Quelques explications :

3ème ligne

La classe `.row` permet de créer une nouvelle ligne.

La classe `.col-md-2` permet de créer un nouvel espace qui prend 2 places (sur les 12 possibles).

La classe `.col-md-offset-5` permet de créer une marge à gauche de 5 places.

Nous avons donc une zone s'étalant sur 2 emplacements centrés avec 5 emplacements à gauche et 5 à droite de marge (ce qui fait bien 12 au total).

La classe `.bordureBleue` permet de simplement de voir apparaitre notre zone à l'écran entourée de bleu.

4ème ligne

La classe `.row` permet de créer une nouvelle ligne.

La classe `.col-md-3` permet de créer un nouvel espace qui prend 3 places (sur les 12 possibles).

Nous avons donc 4 zones s'étalant sur 3 emplacements chacun et sans aucune marge entre eux (ce qui fait bien 12 au total).

Les bordures permettent simplement de voir apparaitre nos zones à l'écran entourées de couleur.

5ème ligne

La classe `.row` permet de créer une nouvelle ligne.

La classe `.col-md-12` permet de créer un nouvel espace qui prend 12 places (sur les 12 possibles).

Nous avons donc une seule zone s'étalant sur 12 emplacements centrés et sans aucune marge entre eux (ce qui fait bien 12 au total).

La classe `.bordureBleue` permet de simplement de voir apparaitre notre zone à l'écran entourée de bleu.

6ème ligne

La classe `.row` permet de créer une nouvelle ligne.

La classe `.col-md-5` permet de créer un nouvel espace qui prend 5 places (sur les 12 possibles).

La classe `.col-md-3` permet de créer un nouvel espace qui prend 3 places (sur les 12 possibles).

La classe `.col-md-offset-2` permet de créer une marge à gauche de 2 places.

Nous avons donc 2 zones s'étalant respectivement sur 5 et 3 emplacements (ce qui fait 8 places) et 2 emplacements de marge chacun (x2) (ce qui fait bien 12 au total).

Les bordures permettent simplement de voir apparaitre nos zones à l'écran entourées de couleur.

Bon, on pourrait continuer longtemps ainsi mais si vous lisez attentivement ce passage je pense que vous comprendrez aisément ce principe !

Par conséquent, plus besoin de `float: left;` pour mettre 2 zones côte à côte dans bootstrap ! C'est plus simple et aussi plus souple.



➤ Résumé

Pour obtenir plusieurs lignes, nous faisons appel plusieurs fois à la classe `<div class="row">`.

Pour "passer" des colonnes, nous utilisons la classe `.offset`.

Revenons tout de même sur un point : les tailles et comportements d'emplacements (à ne pas confondre avec le nombre d'emplacements !).

Si nous avons voulu prévoir un emplacement le plus petit possible, nous aurions pu le préciser de la sorte :

```
grille.html

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/style.css">
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body>
    <div class="container">
      <div class="row">
        <div class="col-xs-1 bordureRouge">
          ...
        </div>
      </div>
    </div>
  </body>
</html>
```

D'autres tailles d'emplacements sont possibles : `col-xs-*` ou `col-sm-*` ou `col-md-*` ou `col-lg-*`.

Question : « Je ne comprends pas, c'est le chiffre 1 ou les lettres md qui gère la taille de l'emplacement ? »

Très bonne question, le nombre d'emplacements est géré par le chiffre tandis que la taille d'un emplacement et son élasticité sont gérées par les lettres :

xs, sm, md ou lg.

1. xs = extra small screens (mobile - écran petit)
2. sm = small screens (tablette - écran réduit)
3. md = medium screens (ordinateur portable - écran moyen)
4. lg = large screens (ordinateur de bureau - grand écran)

Pour le comprendre, voici un tableau récapitulatif avec les 4 tailles majeures d'écrans et de résolutions :

Écran	Écran minimum	Écran réduit	Écran moyen	Grand Écran
Illustration				
Type	SmartPhone	Tablette	Ordinateur Portable	Ordinateur de bureau
Classe CSS Bootstrap	col-xs-*	col-sm-*	col-md-*	col-lg-*
Résolution d'écran	< 768 px	>= 768 px < 992 px	>= 992 px < 1200 px	>= 1200 px

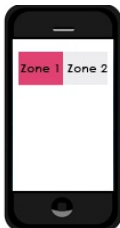

Vous pouvez retrouver le [tableau officiel de bootstrap en cliquant ici](#)

Comme vous le voyez, Bootstrap a prévu des classes différentes en fonction des tailles d'écrans.

Question : « Oui d'accord, j'y vois à peine plus clair mais moi qu'est-ce que je dois choisir entre col-xs-*, col-lg-* et les autres possibilités ? et comment je m'y retrouve ? »

Réponse : Tout dépend de ce que vous voulez faire, pour cela il nous faut un cas d'étude !

Exemple: vous avez 2 zones que vous souhaitez conserver côte à côte lors de la lecture sur smartphone. Il faudra donc choisir col-xs-*.

Zone col-xs-* sur SmartPhone	Zone col-lg-* sur SmartPhone
	

Si vous choisissez col-lg-*, il est certain que les zones auraient fini par s'empiler l'une en dessous de l'autre lors de l'affichage sur un petit écran.

Pour résumé :

col-xs-* : les 2 zones resteront côte à côte sur un petit écran

col-lg-* : les 2 zones s'empileront l'une en dessous de l'autre sur un petit écran.

Pour info et si vous souhaitez tester ce comportement, voici le code correspondant :

```

<!doctype html>
<html>
<head>
<meta charset="utf-8" >
<title>Mon Site avec Bootstrap</title>
<link rel="stylesheet" href="css/bootstrap.css">
<link rel="stylesheet" href="css/style.css">
<meta name="viewport" content="width=device-width, initial-scale=1">
</head>
<body>
<div class="container">
<div class="row">
<div class="col-xs-6 bordureRouge"> ... </div>
<div class="col-xs-6 bordureVerte"> ... </div>
</div>
<div class="row">
<div class="col-lg-6 bordureBleue"> ... </div>
<div class="col-lg-6 bordureOrange"> ... </div>
</div>
</div>
</body>
</html>

```

Il ne vous reste plus qu'à redimensionner et recadrer la fenêtre pour faire des tests.

Dans ce code, sur la 1ère ligne nous prenons 6 emplacements 2 fois en XS, ce qui fait bien un total de 12 places au total (la capacité maximum de notre grille). Idem concernant la seconde ligne en LG.

12

Différence entre col-xs-*, col-sm-*, col-md-*, col-lg-*

Les tailles largeur et comportement d'emplacement sont exprimées en lettre : xs, sm, md, lg.

Le nombre d'emplacements sur la grille est exprimé en chiffre : 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12.

Pour se donner un ordre d'idée, voici ce que cela donne s'il n'y a qu'un seul emplacement sur une seule ligne :

En résolution d'écran >= 1200px :

col-xs-1 : 8.33%

col-sm-1 : 8.33%

col-md-1 : 8.33%

col-lg-1 : 8.33%

	col 1	col 2	col 3	col 4	col 5	col 6	col 7	col 8	col 9	col 10	col 11	col 12
row	xs - 8,33%											
row	sm - 8,33%											
row	md - 8,33%											
row	lg - 8,33%											

En résolution d'écran >= 992 px et < 1200 px :

col-xs-1 : 8.33%

col-sm-1 : 8.33%

col-md-1 : 8.33%

col-lg-1 : 100%

	col 1	col 2	col 3	col 4	col 5	col 6	col 7	col 8	col 9	col 10	col 11	col 12
row	xs - 8,33%											
row	sm - 8,33%											
row	md - 8,33%											
row	lg - 100 %											

En résolution d'écran >= 768 px et < 992 px :

col-xs-1 : 8.33%
col-sm-1 : 8.33%
col-md-1 : 100%
col-lg-1 : 100%

	col 1	col 2	col 3	col 4	col 5	col 6	col 7	col 8	col 9	col 10	col 11	col 12
row	xs - 8,33%											
row	sm - 8,33%											
row	md - 100%											
row	lg - 100 %											

En résolution d'écran < 768 px :

col-xs-1 : 8.33%
col-sm-1 : 100%
col-md-1 : 100%
col-lg-1 : 100%

	col 1	col 2	col 3	col 4	col 5	col 6	col 7	col 8	col 9	col 10	col 11	col 12
row	xs - 8,33%											
row	sm - 100 %											
row	md - 100 %											
row	lg - 100 %											

Question : A quelle taille correspondent exactement les chiffres et le nombre d'emplacements ?

Tableau des pourcentage % par nombre d'emplacement.												
emplacement	1	2	3	4	5	6	7	8	9	10	11	12
taille en largeur (width)	8,33%	16,66%	25%	33,33 %	41,66 %	50 %	58,33 %	66,66 %	75 %	83,33 %	91,66 %	100 %

Si vous souhaitez faire des tests, voici le code correspodant :

grille.html

```
<!Doctype html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/style.css">
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body>
    <div class="container">
      <div class="row">
        <div class="col-xs-1 bordureRouge"> ... </div>
        <div class="col-sm-1 bordureVerte"> ... </div>
        <div class="col-md-1 bordureBleue"> ... </div>
        <div class="col-lg-1 bordureOrange"> ... </div>
      </div>
    </div>
  </body>
</html>
```

Redimensionner la fenêtre pour voir le comportement de chacune des zones

Et oui, si vous êtes un ancien intégrateur il va falloir vous y faire ! nous n'intégrons plus en pixels (px) mais en % le plus possible ! avec ou sans bootstrap.



➤ Bon à savoir

Noter que nous ne réglons pas forcément la hauteur (height) car cela dépend du contenu à l'intérieur de chaque zone.

La classe `.container` et la classe `.container-fluid`

La classe `.container` permet d'englober la page (ainsi que le reste du code) et de le centrer sur la page, dans une taille fixe qui s'adapte en fonction de la résolution d'écran.

La classe `.container-fluid` permet d'englober la page (ainsi que le reste du code) sur toute la largeur de la page.

Écran	Écran minimum	Écran réduit	Écran moyen	Grand Écran
Illustration				
Type	SmartPhone	Tablette	Ordinateur Portable	Ordinateur de bureau
Résolution d'écran	< 768 px	>= 768 px < 992 px	>= 992 px < 1200 px	>= 1200 px
Classe CSS <code>.container</code>	100 % de la taille restante	750 px	970 px	1170 px
Classe CSS <code>.container-fluid</code>	100 %	100 %	100 %	100 %

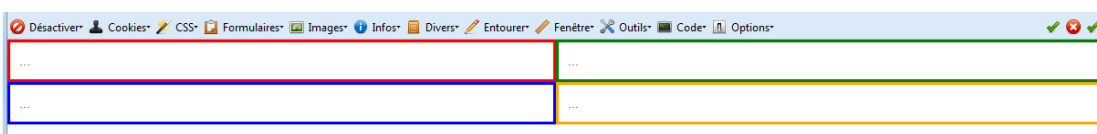
Voici un exemple avec la classe `.container-fluid` :

```

grille.html

<!Doctype html>
<html>
<head>
  <meta charset="utf-8" >
  <title>Mon Site avec Bootstrap</title>
  <link rel="stylesheet" href="css/bootstrap.css">
  <link rel="stylesheet" href="css/style.css">
  <meta name="viewport" content="width=device-width, initial-scale=1">
</head>
<body>
  <div class="container-fluid">
    <div class="row">
      <div class="col-xs-6 bordureRouge"> ... </div>
      <div class="col-xs-6 bordureVerte"> ... </div>
    </div>
    <div class="row">
      <div class="col-lg-6 bordureBleue"> ... </div>
      <div class="col-lg-6 bordureOrange"> ... </div>
    </div>
  </div>
</body>
</html>

```



résultat

Avec l'utilisation de la classe CSS `.container-fluid`, les emplacements prennent toute la largeur de la page web.

grille.html

```

<!Doctype html>
<html>
<head>
  <meta charset="utf-8" >
  <title>Mon Site avec Bootstrap</title>
  <link rel="stylesheet" href="css/bootstrap.css">
  <link rel="stylesheet" href="css/style.css">
  <meta name="viewport" content="width=device-width, initial-scale=1">
</head>
<body>
  <div class="container">
    <div class="row">
      <div class="col-xs-6 col-sm-4 bordureRouge">
        Colonne 1
      </div>
      <div class="col-xs-6 col-sm-8 bordureVerte">
        Colonne 2
      </div>
    </div>
  </div>
</body>
</html>

```

La colonne 1 prendra 1/2 de l'écran pour les écrans mobiles (la moitié soit 6 places), et 1/3 de l'écran pour les tablettes (un tiers soit 4 places).

La colonne 2 prendra 1/2 de l'écran pour les écrans mobiles (la moitié soit 6 places), et 2/3 de l'écran pour les tablettes (deux tiers soit 8 places).

- Sur un écran mobile (smartphone), taille inférieure à < 768px, ce sont les classes **xs** qui seront prises en compte.
- Sur un écran type tablette, taille supérieure à >= 768 px et inférieure à < 992 px, ce sont les classes **sm** qui seront prises en compte.
- Sur un écran type ordinateur portable, taille supérieure ou égal à >= 992 px et inférieure à < 1200 px, ce sont les classes **md** qui seront prises en compte.
- Sur un écran type ordinateur de bureau, taille supérieure ou égal à >= 1200 px, ce sont les classes **lg** qui seront prises en compte.

Avec bootstrap il est facilement possible de donner des règles d'affichage différentes en fonction de la résolution utilisée.

Exemple / Cas d'étude : Imaginons que sur ma page web, je souhaite afficher une zone d'informations complémentaires sur ordinateur et non pas sur mobile (car je dois garder uniquement le contenu principal et faire des choix d'affichage par manque de place).

Pour cela, nous allons pouvoir nous servir des classes `visible-* hidden-*`

Nous pouvons également demander un nombre d'emplacements selon la taille de l'écran et la résolution utilisée.

grille.html

```

<!Doctype html>
<html>
<head>
  <meta charset="utf-8" >

```

```

<title>Mon Site avec Bootstrap</title>
<link rel="stylesheet" href="css/bootstrap.css">
<link rel="stylesheet" href="css/style.css">
<meta name="viewport" content="width=device-width, initial-scale=1">
</head>
<body>
  <div class="container">
    <div class="row">
      <div class="col-xs-6 col-sm-5 col-md-5 col-lg-4 bordureRouge">
        Zone Gauche
      </div>
      <div class="col-xs-6 col-sm-5 col-md-5 col-lg-4 bordureVerte">
        Zone Milieu
      </div>
      <div class="hidden-xs col-sm-2 col-md-2 col-lg-4 bordureOrange">
        Zone Droite
      </div>
    </div>
  </div>
</body>
</html>

```

résultat En taille grand écran (maximum, ordinateur de bureau) - LG :



La zone de gauche (rouge), celle du milieu (verte) et celle de droite (orange) prennent toutes 4 places chacune.

En taille écran moyen (moyen, ordinateur portable) - MD :



La zone de gauche (rouge) prend 5 places (+1), la zone du milieu (verte) prend 5 places (+1), et la zone de droite (orange) prend 2 places (-2).

En taille écran réduit (réduit, tablette) - SM :



La zone de gauche (rouge) prend toujours 5 places (-), la zone du milieu (verte) prend toujours 5 places (-), et la zone de droite (orange) prend toujours 2 places (-).

En taille petit écran (petit, smartphone) - XS :



La zone de gauche (rouge) prend 6 places (+1), la zone du milieu (verte) prend 6 places (+1), et la zone de droite (orange) ne s'affiche plus du tout avec `hidden-xs` donc 0 place (-2).

Avec l'utilisation des float, nous pouvons démarrer un flux de la gauche ou de la droite.

Cela est un peu différent dans bootstrap, nous pouvons créer des décalages.

Pour s'écarter de la gauche, nous utiliserons la classe : `col-lg-push-*`

Pour s'écarter de la droite, nous utiliserons la classe : `col-lg-pull-*`

Exemple :

```
grille.html
<!Doctype html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/style.css">
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body>
    <div class="container">
      <div class="row">
        <div class="col-md-9 col-md-push-3 bordureRouge">.col-md-9 .col-md-push-3</div>
        <div class="col-md-3 col-md-pull-9 bordureVerte">.col-md-3 .col-md-pull-9</div>
      </div>
    </div>
  </body>
</html>
```

17

Bootstrap et les images !

Bootstrap inclut une classe `img-responsive` qui permet aux images de suivre la taille de leur conteneur parent.

Exemple :

```
grille.html
<!Doctype html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/style.css">
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body>
    <div class="container">
      <div class="row">
        <div class="col-md-5 bordureRouge">
          
        </div>
      </div>
    </div>
  </body>
</html>
```

Et voici le code prédéfini dans bootstrap correspondant :

```
bootstrap.css
.img-responsive,
.thumbnail > img,
.thumbnail a > img,
.carousel-inner > .item > img,
.carousel-inner > .item > a > img {
  display: block;
  max-width: 100%;
  height: auto;
}
```

En appliquant cette classe, cela permet donc à chaque image de se recadrer (redimensionner) automatiquement à une taille de 100% du conteneur dans lequel elles seront contenues.

résultat Pour une fenêtre ayant 300 pixels de largeur et 180 pixels de hauteur :

Pour une fenêtre ayant 200 pixels de largeur et 80 pixels de hauteur :

L'image se redimensionnera dans tous les cas aux bonnes dimensions avec l'utilisation de la classe `img-responsive`.

Voici le code à étudier :

exercice-grille.html

```
<!Doctype html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/style.css">
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body>
    <div class="container">
      <div class="row"> <!-- exercice - ligne 6 -->
        <div class="col-md-6 col-md-offset-2 bordureRouge"></div>
        <div class="col-md-3 col-md-offset-2 bordureVerte"></div>
      </div>
    </div>
  </body>
</html>
```

Prenons le cas suivant :



Nous souhaitons mettre les zones côte à côte comme ceci :



D'après-vous, pourquoi les zones ne sont pas côte à côte ? pourtant nous n'avons pas rappeler de `<div class="row">`, qu'est-ce qui ne fonctionne pas ?

Et bien nous avons mis un emplacement de 6 places, avec un emplacement de 3 places et 2 place de marge de chaque côté (soit 4 places).

```
<div class="row"> <!-- exercice - ligne 6 -->
  <div class="col-md-6 col-md-offset-2 bordureRouge"></div>
  <div class="col-md-3 col-md-offset-2 bordureVerte"></div>
</div>
```

Tout cela fait 13 emplacements au total, du coup le cadre vert passe à la ligne car il y est contraint !

➤ Informations

Attention à ne pas dépasser le nombre d'emplacements disponibles porté a 12 dans le système de bootstrap !

Maintenant que nous avons appris tout cela, l'objectif est quand même de pouvoir s'en resserrer à bonne utilisation !

Pour cela, je vous propose de construire le design d'une page web dans une grille bootstrap, suivons ce schéma standard :

	col 1	col 2	col 3	col 4	col 5	col 6	col 7	col 8	col 9	col 10	col 11	col 12
row	header - 1	header - 1		header - 2	header - 2	header - 2	header - 2	header - 2	header - 2		header - 3	header - 3
row	header - 1	header - 1		header - 2	header - 2	header - 2	header - 2	header - 2	header - 2		header - 3	header - 3
row	nav	nav	nav	nav	nav	nav	nav	nav	nav	nav	nav	nav
row	section - 1	section - 1		section - 2	section - 2	section - 2	section - 2	section - 2	section - 2		section - 3	section - 3
row	section - 1	section - 1		section - 2	section - 2	section - 2	section - 2	section - 2	section - 2		section - 3	section - 3
row	section - 1	section - 1		section - 2	section - 2	section - 2	section - 2	section - 2	section - 2		section - 3	section - 3
row	section - 1	section - 1		section - 2	section - 2	section - 2	section - 2	section - 2	section - 2		section - 3	section - 3
row	section - 1	section - 1		section - 2	section - 2	section - 2	section - 2	section - 2	section - 2		section - 3	section - 3
row	section - 1	section - 1		section - 2	section - 2	section - 2	section - 2	section - 2	section - 2		section - 3	section - 3
row	section - 1	section - 1		section - 2	section - 2	section - 2	section - 2	section - 2	section - 2		section - 3	section - 3
row	section - 1	section - 1		section - 2	section - 2	section - 2	section - 2	section - 2	section - 2		section - 3	section - 3
row	section - 1	section - 1		section - 2	section - 2	section - 2	section - 2	section - 2	section - 2		section - 3	section - 3
row	footer - 1	footer - 1		footer - 2	footer - 2	footer - 2	footer - 2	footer - 2	footer - 2		footer - 3	footer - 3
row	footer - 1	footer - 1		footer - 2	footer - 2	footer - 2	footer - 2	footer - 2	footer - 2		footer - 3	footer - 3

Comme vous pouvez le constater, nous pouvons apercevoir :

- 3 zones en haut de page (balise header).
- 1 zone de menu pour la navigation (balise nav).
- 3 zones de contenu (balise section).
- 3 zones de footer (balise footer).

Exercice :

A vous de jouer pour construire cela sur la grille !

Voici un indice : PS : Bien entendu, pour plus de personnalisation, nous pouvons créer une grille dans une grille !

Correction :

grille-page-web.html

```

<!Doctype html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/grille-page-web.css">
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body>
    <div class="container">
      <div class="row">
        <div class="col-md-12 bordureRouge">
          <div class="col-md-2 bordureRouge header-1"> header - 1 </div>
          <div class="col-md-6 col-md-offset-1 bordureRouge header-2"> header - 2 </div>
          <div class="col-md-2 col-md-offset-1 bordureRouge header-3"> header - 3 </div>
        </div>
      </div>
    </div>
  </body>
</html>

```

```

<div class="row">
  <nav class="col-md-12 bordureOrange nav">
    nav
  </nav>
</div>
<div class="row">
  <section class="col-md-12 bordureBleue">
    <div class="col-md-2 bordureBleue section-1"> section - 1 </div>
    <div class="col-md-6 col-md-offset-1 bordureBleue section-2"> section - 2 </div>
    <div class="col-md-2 col-md-offset-1 bordureBleue section-3"> section - 3 </div>
  </section>
</div>
<div class="row">
  <footer class="col-md-12 bordureVerte">
    <div class="col-md-2 bordureVerte footer-1"> footer - 1 </div>
    <div class="col-md-6 col-md-offset-1 bordureVerte footer-2"> footer - 2 </div>
    <div class="col-md-2 col-md-offset-1 bordureVerte footer-3"> footer - 3 </div>
  </footer>
</div>
</div>
</body>
</html>

```

La correction Css :

grille-page-web.css

```

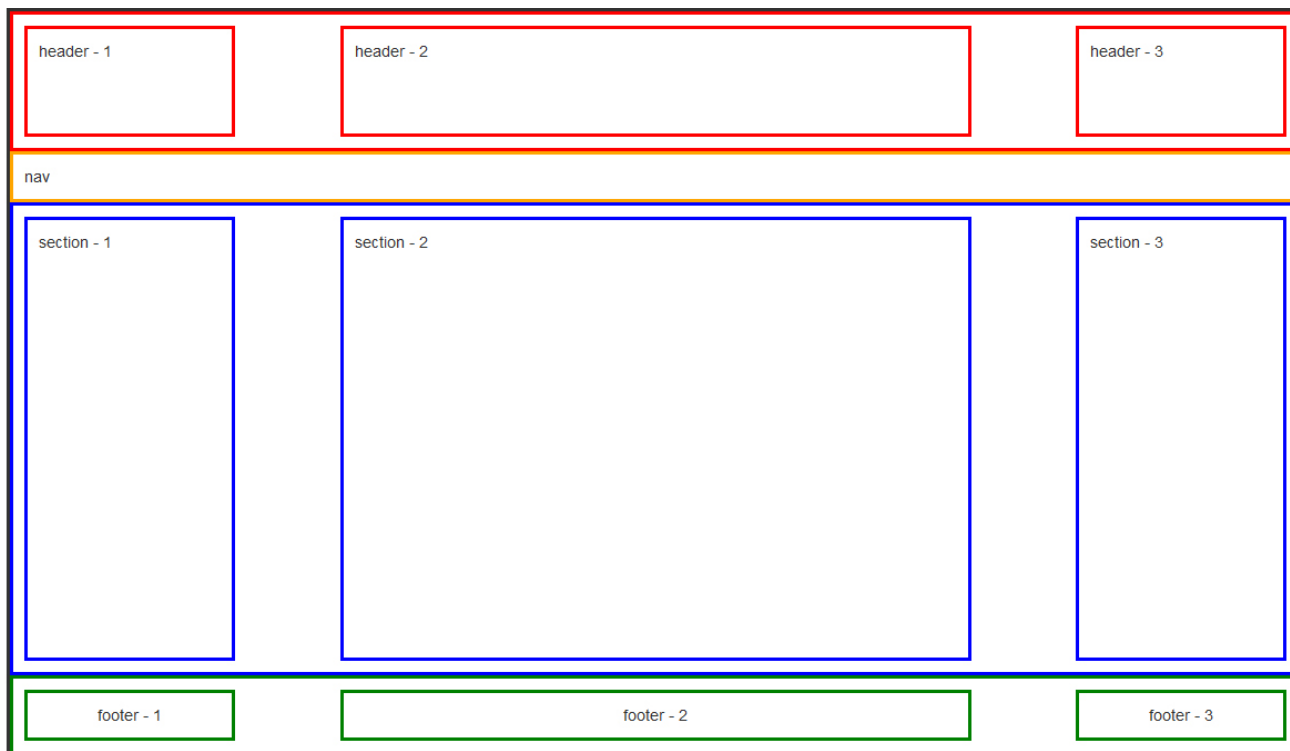
.container{ border: 3px solid; }

.bordureRouge{ border: 3px solid red; padding: 10px; }
.bordureVerte{ border: 3px solid green; padding: 10px; }
.bordureBleue{ border: 3px solid blue; padding: 10px; }
.bordureJaune{ border: 3px solid yellow; padding: 10px; }
.bordureOrange{ border: 3px solid orange; padding: 10px; }

.header-1, .header-2, .header-3{ min-height: 100px; }
.section-1, .section-2, .section-3{ min-height: 400px; }

```

Résultat



J'espère que tout cela a été clair pour vous !, si ce n'est pas le cas n'hésitez pas à relire autant que nécessaire le code ainsi que les explications de ce chapitre.

Pour créer un site web avec Bootstrap, nous aurons besoin de garder à disposition les classes css prédéfinies, nous allons donc ouvrir la [documentation officielle](#) dans un nouvel onglet (sauf si vous connaissez déjà tout par coeur :p)

index.html

```

<!Doctype html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/style.css">
    <script src="http://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="js/bootstrap.js"></script>
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body>
    <header>
      <div class="container"> <!-- Tout contenant se divise en 12 colonnes. #container doit etre present et il permet de centrer -->
        <div class="row"> <!-- .row : on utilise "row" dès lors que l'on souhaite faire une ligne. -->
          <div class="col-md-7 col-md-offset-1 bordureRouge"> <!-- col est colonne // md c'est middle (xs petit) // 7 est le nb de place prise sur
les 12 disponible, -- col-md-offset-1 correspond a la marge de gauche de l'espace d'une colonne. -->
            <h1>
              <span class="glyphicon glyphicon-star"></span>
              <a href="index.html">Monnom Monprenom</a>
            </h1>
            <small>Communication / Marketing / Digital</small>
          </div>
          <div class="col-md-3 bordureVerte"> <!-- col-md-3 je prends 3 places (sur les 12) -->
            <a href="fichier/cv.pdf">Telecharger mon CV</a><br>
            <a href="">Partagez</a>
          </div>
        </div>
        <div class="row"> <!-- row : une ligne. -->
          <div class="col-md-10 col-md-offset-1"> <!-- col-md-10 : je prends 10 places // col-md-offset-1 : marge de l'espace d'une colonne-->
            <nav class="navbar"> <!-- navbar marges en dessous le menu -->
              <ul class="nav nav-tabs nav-justified" role="tablist"> <!-- nav-justified permet de prendre l'espace totale en largeur, peut importe le nb
de liens, c'est recalculé (tester en ajoutant 1 lien en+)-->
                <li><a href="index.html">Accueil</a></li> <!-- nav-tabs permet de mettre un trait en desous et de donner un style "onglet". -->
                <li><a href="moncv.html">Mon CV</a></li><!-- nav permet de mettre un display block et faire en sorte qu'il prenne davantage de
place. -->
                <li><a href="mesprojets.html">Mes Projets</a></li>
                <li><a href="contact.php">Contact</a></li>
              </ul>
            </nav>
          </div>
        </div>
      </div>
    </header>
  </body>
</html>

```

Résultat



Telecharger mon CV
Partagez

Quelques Explications (sur les classes bootstrap utilisées ci-dessus pour construire le haut de site)

A l'intérieur de la balise <header>, la zone <div class="container"> permet de centrer le contenu sur la page web et de travailler à l'intérieur du système 12 colonnes.

Une fois à l'intérieur du container, nous créons une <div class="row"> qui permet de déclarer une nouvelle ligne.

A l'intérieur de cette zone `<div class="row">`, nous prévoyons un emplacement de 7 places `.col-md-7` avec 1 place de marge à gauche `.col-md-offset-1`

Le `` présent à l'intérieur contient la classe `.glyphicon glyphicon-star` permettant d'afficher une icône à côté du titre et slogan du site.

A côté de cette première zone, nous y associons un deuxième emplacement `.col-md-3` (3 places) contenant des liens.

L'utilisation à nouveau de `<div class="row">` nous permet de déclarer une nouvelle ligne pour prévoir le menu de navigation.

Pour cela, nous n'utilisons qu'un seul emplacement de 10 places `.col-md-10` avec 1 place de marge de chaque côté `.col-md-offset-1`.

La classe CSS `.navbar` permet d'appliquer une marge sous le menu.

La classe CSS `.nav` permet d'appliquer un `display block` sur le ``.

La classe CSS `.nav-tabs` permet de mettre un trait en dessous et de donner un style "onglet".

La classe CSS `.nav .nav-tabs .nav-justified` permet de prendre l'espace total en largeur, peu importe le nombre de liens, c'est recalculé.

```
index.html
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/style.css">
    <script src="http://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="js/bootstrap.js"></script>
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body>
    <header>
      <div class="container">
        <div class="row">
          <div class="col-md-7 col-md-offset-1 bordureRouge">
            <h1>
              <span class="glyphicon glyphicon-star"></span>
              <a href="index.html">Monnom Monprenom</a>
            <br>
            <small>Communication / Marketing / Digital</small>
            </h1>
          </div>
          <div class="col-md-3 bordureVerte">
            <a href="fichier/cv.pdf">Telecharger mon CV</a><br>
            <a href="">Partagez</a>
          </div>
        </div>
        <div class="row">
          <div class="col-md-10 col-md-offset-1">
            <nav class="navbar">
              <ul class="nav nav-tabs nav-justified" role="tablist">
                <li><a href="index.html">Accueil</a></li>
                <li><a href="moncv.html">Mon CV</a></li>
                <li><a href="mesprojets.html">Mes Projets</a></li>
                <li><a href="contact.php">Contact</a></li>
              </ul>
            </nav>
          </div>
        </div>
      </div>
    </header>
    <section>
      <div class="container">
        <div class="row">
          <div class="col-md-7 col-md-offset-1">
            <div class="panel panel-default">
              <div class="panel-heading"><h2>Presentation</h2></div>
              <div class="panel-body">
                <div class="col-md-2">
                  
                </div>
                <div class="col-md-10">
                  <p class="lead">
                    Vous etes a la recherche d'une personne competente et efficace pour gerer vos projets web ?
                  </p>
                </div>
                <div class="col-md-12">
                  <hr>
                  <p>Actuellement, je suis ..., je fais ... actuellement, je suis ..., je fais ... actuellement, je suis ..., je fais ... actuellement, je suis ..., je fais ...</p>
                  <ul class="list-unstyled">
                    <li>Redaction cahier des charges</li>
                    <li>elaboration de votre design & Integration precise</li>
                    <li>Administration sur CMS : Wordpress & Drupal</li>
                    <li>Reflexion strategique, statistiques, référencement</li>
                  </ul>
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
    </section>
  </body>
</html>
```

```

        
      </p>
    </div>
  </div>
</div>
<div class="panel panel-default">
  <div class="panel-heading"><h2>Actualites</h2></div>
  <div class="panel-body">
    <p>
      <em>
        17/06/2013 : Formation a l'IFOCOP @Paris.<br>
        15/05/2013 : Actualite 3 <br>
        12/03/2013 : Actualite 2<br>
        01/01/2013 : Actualite 1<br>
      </em>
    </p>
  </div>
</div>
<div class="col-md-3">
  <div class="panel panel-default">
    <div class="panel-heading"><h2>Qui suis-je ?</h2></div>
    <div class="panel-body">
      <p>
        Description en quelques lignes ... <br>
        Description en quelques lignes ... <br>
        Description en quelques lignes ... <br>
      </p>
    </div>
  </div>
  <div class="panel panel-default">
    <div class="panel-heading"><h2>Experience</h2></div>
    <div class="panel-body">
      <p>
        <ul>
          <li>Experience 1 ... </li>
          <li>Experience 2 ... </li>
          <li>Experience 3 ... </li>
        </ul>
      </p>
    </div>
  </div>
  <div class="panel panel-default">
    <div class="panel-heading"><h2>Contact</h2></div>
    <div class="panel-body">
      <p>
        Contact ... <br>
      </p>
    </div>
  </div>
</div>
</div>
</section>
</body>
</html>

```

Résultat

Presentation



Vous etes a la recherche d'une personne competente et efficace pour gerer vos projets web ?

Actuellement, je suis ..., je fais ... actuellement, je suis ..., je fais ... actuellement, je suis ..., je fais ...
... actuellement, je suis ..., je fais ...

Redaction cahier des charges
elaboration de votre design & Integration precise
Administration sur CMS : Wordpress & Drupal
Reflexion strategique, statistiques, référencement



Qui suis-je ?

Description en quelques lignes ...
Description en quelques lignes ...
Description en quelques lignes ...

Experience

- Experience 1 ...
- Experience 2 ...
- Experience 3 ...

Contact

Contact ...

Actualites

17/06/2013 : Formation a l'IFOCOP @Paris.
15/05/2013 : Actualite 3
12/03/2013 : Actualite 2
01/01/2013 : Actualite 1

Quelques Explications (sur les classes bootstrap utilisées ci-dessus pour construire le milieu de site)

A l'intérieur de la balise `<section>`, la zone `<div class="container">` permet de centrer le contenu sur la page web et de travailler à l'intérieur du système 12 colonnes.

Une fois à l'intérieur du container, nous créons une `<div class="row">` qui permet de déclarer une nouvelle ligne.

A l'intérieur de cette zone `<div class="row">`, nous prévoyons un emplacement de 7 places `.col-md-7` avec 1 place de marge à gauche `.col-md-offset-1`

Les classes présentes à l'intérieur sont `.panel`, `.panel-default`, `.panel-heading`, `.panel-body`

La classe CSS `.panel` permet d'appliquer de créer un cadre avec un effet "box".

La classe CSS `.panel-default` permet d'ajouter de l'ombre.

La classe CSS `.panel-heading` permet de donner une entete à notre cadre panel.

La classe CSS `.panel-body` permet d'appliquer du padding (marges intérieures) à l'intérieur du cadre.

Dans cette même zone de 7 places, nous recréons d'autres zones (une grille dans une grille).

Nous prévoyons un autre emplacement de 2 places `.col-md-2` avec une image à l'intérieur

Et à côté, un emplacement de 10 places `.col-md-10` avec un texte à l'intérieur

Cela prend 12 places à l'intérieur de la zone dans laquelle ils sont contenus, soit 12 places à l'intérieur des 7 places de leur parent.

La classe CSS `.lead` permet d'avoir un effet gras.

Un autre emplacement de 12 places (à l'intérieur des 7 places du parent) est déclaré, du coup il passe forcément à la ligne même sans l'utilisation de `<div class="row">` car il n'a plus la place de se mettre à côté.

Une liste est déclarée avec la classe CSS `list-unstyled` qui permet d'éviter les puces (point noir) de la liste.

D'autres cadres utilisant les classes CSS ci-dessus sont déclarés pour avoir un effet 2 colonnes.

Voici les liens pour récupérer les deux images utilisées dans le cadre de ce site web. Cliquez ici : [image1](#) & [image2](#)

index.html

```
<!Doctype html>
<html>
  <head>
    <meta charset="utf-8" >
    <title>Mon Site avec Bootstrap</title>
    <link rel="stylesheet" href="css/bootstrap.css">
    <link rel="stylesheet" href="css/style.css">
    <script src="http://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="js/bootstrap.js"></script>
  </head>
  <body>
    <header>
      <div class="container">
        <div class="row">
          <div class="col-md-7 col-md-offset-1 bordureRouge">
            <h1>
              <span class="glyphicon glyphicon-star"></span>
              <a href="index.html">Monnom Monprenom</a>
            <br>
            <small>Communication / Marketing / Digital</small>
            </h1>
          </div>
          <div class="col-md-3 bordureVerte">
            <a href="fichier/cv.pdf">Telecharger mon CV</a><br>
            <a href="">Partagez</a>
          </div>
        </div>
        <div class="row">
          <div class="col-md-10 col-md-offset-1">
            <nav class="navbar">
              <ul class="nav nav-tabs nav-justified" role="tablist">
                <li><a href="index.html">Accueil</a></li>
                <li><a href="moncv.html">Mon CV</a></li>
                <li><a href="mesprojets.html">Mes Projets</a></li>
                <li><a href="contact.php">Contact</a></li>
              </ul>
            </nav>
          </div>
        </div>
      </div>
    </header>
    <section>
      <div class="container">
        <div class="row">
          <div class="col-md-7 col-md-offset-1">
            <div class="panel panel-default">
              <div class="panel-heading"><h2>Presentation</h2></div>
              <div class="panel-body">
                <div class="col-md-2">
                  
                </div>
                <div class="col-md-10">
                  <p class="lead">
                    Vous etes a la recherche d'une personne competente et efficace pour gerer vos projets web ?
                  </p>
                </div>
                <div class="col-md-12">
                  <hr>
                  <p>Actuellement, je suis ..., je fais ... actuellement, je suis ..., je fais ... actuellement, je suis ..., je fais ... actuellement, je suis ..., je fais ...</p>
                  <ul class="list-unstyled">
                    <li>Redaction cahier des charges</li>
                    <li>elaboration de votre design & Integration precise</li>
                    <li>Administration sur CMS : Wordpress & Drupal</li>
                    <li>Reflexion strategique, statistiques, référencement</li>
                  </ul>
                  
                </div>
              </div>
            </div>
          </div>
          <div class="panel panel-default">
            <div class="panel-heading"><h2>Actualites</h2></div>
            <div class="panel-body">
```


Presentation



Vous etes a la recherche d'une personne
competente et efficace pour gerer vos projets web ?

Actuellement, je suis ..., je fais ... actuellement, je suis ..., je fais ... actuellement, je
suis ..., je fais ... actuellement, je suis ..., je fais ...

Redaction cahier des charges
elaboration de votre design & Integration precise
Administration sur CMS : Wordpress & Drupal
Reflexion strategique, statistiques, référencement



Qui suis-je ?

Description en quelques lignes ...
Description en quelques lignes ...
Description en quelques lignes ...

Experience

- Experience 1 ...
- Experience 2 ...
- Experience 3 ...

Contact

Contact ...

Actualites

17/06/2013 : Formation a l'IFOCOP @Paris.
15/05/2013 : Actualite 3
12/03/2013 : Actualite 2
01/01/2013 : Actualite 1

Presentation



Vous etes a la recherche d'une personne
competente et efficace pour gerer vos
projets web ?

Actuellement, je suis ..., je fais ...
actuellement, je suis ..., je fais ...
actuellement, je suis ..., je fais ...
actuellement, je suis ..., je fais ...

Redaction cahier des charges
elaboration de votre design & Integration
precise
Administration sur CMS : Wordpress & Drupal
Reflexion strategique, statistiques,
referencement



Quelques Explications (sur les classes bootstrap utilisées ci-dessus pour construire le bas de site)

A l'intérieur de la balise `<footer>`, la zone `<div class="container">` permet de centrer le contenu sur la page web et de travailler à l'intérieur du système 12 colonnes.

Une fois à l'intérieur du container, nous créons une `<div class="row">` qui permet de déclarer une nouvelle ligne.

A l'intérieur de cette zone `<div class="row">`, nous prévoyons un emplacement de 10 places `.col-md-10` avec 1 place de marge à gauche `.col-md-offset-1`

Nous y mettons simplement quelques liens et une phrase de texte.

Tous les éléments type `<a>`, `<hr>`, `<p>`, etc. bénéficient d'une légère retouche graphique prévue par bootstrap.

Voilà pour la création d'une première page web avec l'aide de bootstrap ! comme vous le voyez, cela va plus vite puisqu'on écrit pas le code CSS, on ne fait qu'appeler des classes CSS contenant du code déjà prédéfini.

D'autre part, vous pouvez tenter de redimensionner votre fenêtre, vous verrez que votre page web est déjà responsive !

D'une manière générale, si vous avez un doute sur le comportement d'une classe ou sur son apport, vous pouvez :

- Consulter la documentation officielle
- Allez voir le code css du fichier `bootstrap.css`
- L'observer avec firebug

- La retirer du code pour voir ce que cela change à l'affichage, avant de la rétablir dans le code.

The screenshot shows a web browser displaying a portfolio website titled "Monnom Monprenom" with the subtitle "Communication / Marketing / Digital". The website has a navigation bar with links: Accueil, Mon CV, Mes Projets, and Contact. The main content area is divided into two columns. The left column has a "Presentation" section with a heading "Vous etes a la recherche d'une personne competente et efficace pour gerer vos projets web ?", a paragraph of placeholder text, a list of services (Redaction cahier des charges, elaboration de votre design & Integration precise, Administration sur CMS : Wordpress & Drupal, Reflexion strategique, statistiques, referencement), and a 3D pie chart icon. The right column has sections for "Qui suis-je ?", "Experience" (listing three experiences), and "Contact". Below the website, the browser's developer tools are open, showing the HTML structure of the "div.panel-heading" element and the corresponding CSS styles from bootstrap.min.css.

HTML Structure:

```

<div class="row">
  </div>
</header>
<section>
  <div class="container">
    <div class="row">
      <div class="col-md-7 col-md-offset-1">
        <div class="panel panel-default">
          <div class="panel-heading">
            <div class="panel-body">
              </div>
            <div class="panel panel-default">
              </div>
            <div class="col-md-3">
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </section>

```

CSS Styles:

```

*::before, *::after {
  box-sizing: border-box;
}
*::before, *::after {
  box-sizing: border-box;
}
.panel-default > .panel-heading {
  background-color: #f5f5f5;
  border-color: #ddd;
  color: #333;
}
.panel-heading {
  border-bottom: 1px solid transparent;
  border-top-left-radius: 3px;
  border-top-right-radius: 3px;
}

```

Theme et template prefait bootstrap.

Plusieurs themes prefaits existent, vous pouvez les recuperer et les observer à l'adresse officielle de bootstrap dans la partie [Getting Started](#), voici quelques liens d'exemple :

- [Starter Template](#)
- [Bootstrap Theme example](#)
- [Grids](#)
- [Jumbotron](#)
- [Narrow jumbotron](#)
- [Carousel](#)
- [Blog](#)
- [Dashboard \(theme pour backOffice\)](#)
- [Justified nav](#)
- [Off-canvas](#)

