
CSS et Bootstrap

I. Historique de Bootstrap

1. Bootstrap 1.0

Date de sortie : Août 2011

Principales fonctionnalités : Grille de mise en page, typographie, formulaires, boutons, navigation, etc.

2. Bootstrap 2.0

Date de sortie : Janvier 2012

Principales fonctionnalités : Intégration de la grille fluide, amélioration des composants existants, ajout de nouveaux composants.

Version 2.0.0 (31 janvier 2012) : Logs de mise à jour

Version 2.3.2 (31 août 2013) : Logs de mise à jour

3. Bootstrap 3.0

Date de sortie : Août 2013

Principales fonctionnalités : Passage à une conception mobile d'abord (responsive), mise à jour de la grille pour le support des appareils mobiles, introduction de glyphicons, nouvelles classes d'utilitaires, etc.

Version 3.0.0 (19 août 2013) : Logs de mise à jour

Version 3.3.7 (13 juillet 2016) : Logs de mise à jour

4. Bootstrap 4.0

Date de sortie : Janvier 2018

Principales fonctionnalités : Refonte complète du framework, introduction de Flexbox pour la mise en page, ajout de nouvelles classes d'utilitaires, mise à jour des composants existants, support de la conception responsive, etc.

Version 4.0.0 (18 janvier 2018) : Logs de mise à jour

Version 4.5.0 (27 mai 2020) : Logs de mise à jour

5. Bootstrap 5.0

Date de sortie : Mai 2020

Principales fonctionnalités : Suppression de jQuery comme dépendance, introduction de Vanilla JavaScript, mise à jour de la grille, simplification de la syntaxe des classes, ajout de nouveaux composants, améliorations de l'accessibilité, etc.

Version 5.0.0 (5 mai 2021) : Logs de mise à jour

II. Structuration et composants Bootstrap

Éléments de structuration

1. Container Bootstrap : En Bootstrap, vous pouvez utiliser la classe “container” pour envelopper votre contenu dans une division avec une mise en page par défaut.

Exemple en Bootstrap :

```
<div class="container">
  <!-- Contenu de la division -->
</div>
```

2. Flexbox CSS : Flexbox est un modèle de disposition flexible qui permet de créer des mises en page fluides et réactives. Il utilise des conteneurs flexibles (`display: flex;`) pour organiser les éléments à l'intérieur. Les éléments peuvent être facilement alignés, espacés et réorganisés en utilisant des propriétés telles que `justify-content`, `align-items`, `flex-direction`, etc.

Exemple en CSS :

```
.container {
  display: flex;
  justify-content: center;
  align-items: center;
}
```

Bootstrap : En Bootstrap, vous pouvez utiliser le système de grille flexbox pour créer des mises en page responsives. Utilisez les classes `d-flex` pour définir un conteneur flex, et les classes `justify-content-*`, `align-items-*`, `flex-direction-*`, etc., pour ajuster la disposition et l'alignement des éléments.

Exemple en Bootstrap :

```
<div class="d-flex justify-content-center align-items-center">
  <!-- Contenu avec flexbox -->
</div>
```

3. Row (ligne) Les row sont des éléments importants pour la structuration des mises en page dans CSS et Bootstrap.

CSS : une row peut être créée en utilisant les propriétés CSS telles que `display: flex` ou `display: grid`. La row permet de regrouper horizontalement plusieurs éléments et de les aligner selon un axe commun.

Exemple en CSS :

```
.row {
  display: flex;
  flex-wrap: wrap;
}
```

Bootstrap : les row font partie intégrante du système de grille. Elles servent à contenir et aligner les colonnes (`col`) qui structurent le contenu de la page. Les row dans Bootstrap sont généralement utilisées avec les classes `container` ou `container-fluid` pour créer des mises en page réactives.

Exemple en Bootstrap :

```
<div class="container">
  <div class="row">
    <div class="col-md-6">Contenu de la colonne 1</div>
    <div class="col-md-6">Contenu de la colonne 2</div>
  </div>
</div>
```

Dans cet exemple, la row est contenue dans un `container`, et deux colonnes (`col-md-6`) sont placées à l'intérieur de la row. Cela permet d'organiser le contenu en deux colonnes égales.

Les row sont donc des éléments essentiels pour structurer le contenu dans les mises en page CSS et Bootstrap, en permettant d'aligner les éléments horizontalement et de les répartir de manière adaptative en fonction de la taille de l'écran.

4. Colonnes (columns) **CSS** : Les colonnes sont utilisées pour diviser l'espace disponible à l'intérieur d'un conteneur en plusieurs colonnes égales ou de tailles différentes. Vous pouvez utiliser la propriété CSS `width` pour définir la largeur des colonnes.

Exemple en CSS :

```
.column {  
  width: 33.33%;  
  float: left;  
}
```

Bootstrap : En Bootstrap, vous pouvez utiliser les classes `col-*` pour spécifier les colonnes dans une grille. Les classes `col-md-*` sont utilisées pour des mises en page réactives sur les appareils de taille moyenne.

Exemple en Bootstrap :

```
<div class="row">  
  <div class="col-md-4">Colonne 1</div>  
  <div class="col-md-4">Colonne 2</div>  
  
  <div class="col-md-4">Colonne 3</div>  
</div>
```

5. Grid (grille) **CSS** : Le système de grille CSS permet de créer des mises en page en divisant l'espace disponible en colonnes et en lignes. Vous pouvez utiliser les propriétés CSS `grid-template-columns` et `grid-template-rows` pour définir la structure de la grille et positionner les éléments en fonction de leurs coordonnées.

Exemple en CSS :

```
.container {  
  display: grid;  
  grid-template-columns: 1fr 1fr 1fr;  
  grid-template-rows: auto;  
}
```

Bootstrap : En Bootstrap, vous pouvez utiliser le système de grille basé sur Flexbox pour créer des mises en page réactives. Utilisez les classes `container` pour définir un conteneur de grille, et les classes `col-*` pour spécifier les colonnes.

Exemple en Bootstrap :

```
<div class="container">
  <div class="row">
    <div class="col-md-4">Colonne 1</div>
    <div class="col-md-4">Colonne 2</div>
    <div class="col-md-4">Colonne 3</div>
  </div>
</div>
```

6. Grilles complexes CSS et Bootstrap offrent des fonctionnalités avancées pour la création de grilles plus complexes avec des colonnes imbriquées, des marges personnalisées, des offsets, etc. Ces fonctionnalités permettent de créer des mises en page plus flexibles et personnalisées.

Exemple en Bootstrap :

```
<div class="row">
  <div class="col-md-6">Colonne 1</div>
  <div class="col-md-6">
    <div class="row">
      <div class="col-md-6">Colonne 2.1</div>
      <div class="col-md-6">Colonne 2.2</div>
    </div>
  </div>
</div>
```

7. Conteneurs réactifs Les conteneurs réactifs permettent de créer des mises en page qui s'adaptent automatiquement à la taille de l'écran. Ils sont utiles pour créer des sections de contenu qui s'ajustent en largeur en fonction de la résolution de l'appareil.

Exemple en Bootstrap :

```
<div class="container-fluid">
  <!-- Contenu réactif -->
</div>
```

Composants

Rappel : Composant de navigation de base fournis par HTML5

1. En-tête (header) **CSS** : L'en-tête représente la partie supérieure d'un site web qui peut contenir le logo, le titre du site et éventuellement des liens de navigation. Vous pouvez le définir en utilisant la balise <header> en CSS.

Exemple en CSS :

```
header {  
    /* Styles pour l'en-tête */  
}
```

Bootstrap : En Bootstrap, vous pouvez utiliser la classe "header" pour styliser votre en-tête. Vous pouvez envelopper votre en-tête avec une balise <header> et ajouter la classe correspondante.

Exemple en Bootstrap :

```
<header class="header">  
    <!-- Contenu de l'en-tête -->  
</header>
```

2. Contenu principal (main) **CSS** : Le contenu principal représente la section principale du site web qui contient le contenu principal de chaque page. Utilisez la balise <main> en CSS pour définir cette zone.

Exemple en CSS :

```
main {  
    /* Styles pour le contenu principal */  
}
```

Bootstrap : En Bootstrap, vous pouvez envelopper votre contenu principal avec une balise <main> et ajouter la classe "container" pour une mise en page par défaut.

Exemple en Bootstrap :

```
<main class="container">  
    <!-- Contenu principal -->  
</main>
```

3. Pied de page (footer) CSS : Le pied de page est la section inférieure du site web qui contient généralement des informations de copyright, des liens de navigation supplémentaires, etc. Vous pouvez utiliser la balise <footer> en CSS pour définir le pied de page.

Exemple en CSS :

```
footer {  
  /* Styles pour le pied de page */  
}
```

Bootstrap : En Bootstrap, vous pouvez utiliser la classe “footer” pour styliser votre pied de page.

Exemple en Bootstrap :

```
<footer class="footer">  
  <!-- Contenu du pied de page -->  
</footer>
```

Composants fournis par Bootstrap Bootstrap offre une variété d'autres outils de structuration tels que les barres de navigation (navbar), les menus déroulants (dropdowns), les onglets (tabs), les accordéons, etc. Ces outils permettent de créer des composants interactifs et de navigation pour votre site web.

Bien sûr ! Voici un cours sur les différents composants de base de Bootstrap :

Badges Les badges sont de petits éléments utilisés pour afficher des informations supplémentaires ou des notifications. Ils peuvent être utilisés pour indiquer des numéros, des états, des tags, etc.

Exemple de code HTML :

```
<span class="badge badge-primary">1</span>  
<span class="badge badge-danger">Alert</span>
```

Button Les boutons sont des éléments interactifs utilisés pour déclencher des actions lorsqu'ils sont cliqués.

Exemple de code HTML :

```
<button class="btn btn-primary">Primary Button</button>
<button class="btn btn-danger">Danger Button</button>
```

Button Group Les groupes de boutons permettent de regrouper des boutons connexes.

Exemple de code HTML :

```
<div class="btn-group">
  <button class="btn btn-primary">Button 1</button>
  <button class="btn btn-primary">Button 2</button>
  <button class="btn btn-primary">Button 3</button>
</div>
```

Card Les cartes sont des composants polyvalents utilisés pour afficher des informations. Elles peuvent contenir des images, du texte, des boutons, etc.

Exemple de code HTML :

```
<div class="card">
  
  <div class="card-body">
    <h5 class="card-title">Card Title</h5>
    <p class="card-text">Some text goes here.</p>
    <a href="#" class="btn btn-primary">Read More</a>
  </div>
</div>
```

Dropdown Les menus déroulants sont des composants qui affichent une liste d'options lorsque l'utilisateur clique sur un bouton ou un lien.

Exemple de code HTML :

```
<div class="dropdown">
  <button
    class="btn btn-primary dropdown-toggle"
    type="button"
    data-toggle="dropdown"
  >
    Dropdown Button
```

```
</button>
<div class="dropdown-menu">
  <a class="dropdown-item" href="#">Option 1</a>
  <a class="dropdown-item" href="#">Option 2</a>
  <a class="dropdown-item" href="#">Option 3</a>
</div>
</div>
```

Nav Les composants de navigation sont utilisés pour créer des barres de navigation horizontales ou verticales.

Exemple de code HTML :

```
<ul class="nav">
  <li class="nav-item">
    <a class="nav-link active" href="#">Home</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#">About</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#">Contact</a>
  </li>
</ul>
```

Tabs Les onglets sont utilisés pour organiser le contenu en différentes sections et permettre à l'utilisateur de basculer entre elles.

Exemple de code HTML :

```
<ul class="nav nav-tabs">
  <li class="nav-item">
    <a class="nav-link active" href="#section1">Section 1</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#section2">Section 2</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#section3">Section 3</a>
  </li>
</ul>
```

```
</li>
</ul>
```

Navbar La barre de navigation est une composante couramment utilisée pour créer une barre de menu horizontale ou verticale.

Exemple de code HTML :

```
<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
  <a class="navbar-brand" href="#">Logo</a>
  <button
    class="navbar-toggler"
    type="button"
    data-toggle="collapse"
    data-target="#navbarNav"
    aria-controls="navbarNav"
    aria-expanded="false"
    aria-label="Toggle navigation"
  >
    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="collapse navbar-collapse" id="navbarNav">
    <ul class="navbar-nav">
      <li class="nav-item active">
        <a class="nav-link" href="#">Home</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">About</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Contact</a>
      </li>
    </ul>
  </div>
</nav>
```

Modal Les modales sont des fenêtres contextuelles qui s'affichent au-dessus du contenu principal pour afficher des informations supplémentaires ou demander une action à l'utilisateur.

Exemple de code HTML :

```

<button
  type="button"
  class="btn btn-primary"
  data-toggle="modal"
  data-target="#myModal"
>
  Open Modal
</button>

<div class="modal fade" id="myModal">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title">Modal Title</h5>
        <button type="button" class="close" data-dismiss="modal">
          &times;
        </button>
      </div>
      <div class="modal-body">
        <p>Modal content goes here.</p>
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-secondary"
          ↪ data-dismiss="modal">
          Close
        </button>
        <button type="button" class="btn btn-primary">Save</button>
      </div>
    </div>
  </div>
</div>

```

List Group Les listes de groupe sont des composants utilisés pour afficher des listes d'éléments avec des styles spécifiques.

Exemple de code HTML :

```

<ul class="list-group">
  <li class="list-group-item">Item 1</li>
  <li class="list-group-item">Item 2</li>
  <li class="list-group-item">Item 3</li>

```

```
</ul>
```

Toast Les toasts sont des messages temporaires qui apparaissent en haut ou en bas de l'écran pour afficher des notifications à l'utilisateur.

Exemple de code HTML :

```
<div class="toast" role="alert" aria-live="assertive" aria-atomic="true">
  <div class="toast-header">
    <strong class="mr-auto">Notification</strong>
    <small>Just now</small>
    <button
      type="button"
      class="ml-2 mb-1 close"
      data-dismiss="toast"
      aria-label="Close"
    >
      <span aria-hidden="true">&times;</span>
    </button>
  </div>
  <div class="toast-body">This is a toast message.</div>
</div>
```

III. Présentation de la dernière version majeure de Bootstrap (Bootstrap 5.0)

Bootstrap 5.0 est la version majeure la plus récente de Bootstrap. Elle présente plusieurs améliorations et fonctionnalités nouvelles par rapport à la version précédente. Voici quelques points saillants de Bootstrap 5.0 :

- Suppression de jQuery : Bootstrap 5.0 utilise désormais Vanilla JavaScript pour les fonctionnalités interactives, ce qui réduit les dépendances externes.
- Grille améliorée : La grille de mise en page de Bootstrap a été améliorée avec l'introduction de Flexbox, offrant ainsi plus de flexibilité pour la mise en page des éléments.
- Simplification des classes : Les classes de Bootstrap ont été simplifiées, ce qui facilite l'utilisation et la personnalisation du framework.
- Nouveaux composants : Bootstrap 5.0 propose de nouveaux composants tels que les modales, les cartes, les carrousels, les onglets, etc.
- Améliorations de l'accessibilité : Bootstrap 5.0 met l'accent sur l'accessibilité en ajoutant des attributs aria-label et en respectant les meilleures pratiques d'accessibilité.

V. Autres frameworks

Voici cinq frameworks CSS populaires, avec des liens vers leurs sites officiels, leur documentation et des tutoriels :

Tailwind CSS Tailwind CSS est un framework CSS basé sur les classes utilitaires. Il offre une approche totalement personnalisable pour la création d'interfaces utilisateur, en permettant de construire rapidement et efficacement des designs modernes.

Première version : 2017

Version la plus récente : 2021

Site officiel : <https://tailwindcss.com/>

Documentation : <https://tailwindcss.com/docs>

Tutoriel : <https://tailwindcss.com/course>

Bulma Bulma est un framework CSS léger et moderne qui propose une grille flexible et des composants prêts à l'emploi. Il met l'accent sur la simplicité et la modularité, offrant ainsi une grande flexibilité pour la conception d'interfaces réactives.

Première version : 2016

Version la plus récente : 2021

Site officiel : <https://bulma.io/>

Documentation : <https://bulma.io/documentation/>

Tutoriel : <https://bulma.io/resources/learning/>

Foundation Foundation est un framework CSS avancé qui propose une gamme complète d'outils et de composants pour la création de sites web réactifs et modernes. Il met l'accent sur la personnalisation et l'accessibilité.

Première version : 2011

Version la plus récente : 2021

Site officiel : <https://foundation.zurb.com/>

Documentation : <https://foundation.zurb.com/sites/docs/>

Tutoriel : <https://foundation.zurb.com/learn/tutorials.html>

Semantic UI Semantic UI est un framework CSS moderne qui utilise une approche sémantique pour la création d'interfaces utilisateur. Il offre une vaste collection de composants et de styles prédéfinis pour faciliter le développement web.

Première version : 2013

Version la plus récente : 2021

Site officiel : <https://semantic-ui.com/>

Documentation : <https://semantic-ui.com/introduction/getting-started.html>

Tutoriel : <https://semantic-ui.com/introduction/getting-started.html#learn-more>

Materialize CSS Materialize CSS est un framework CSS basé sur les principes du Material Design de Google. Il offre une collection de composants prêts à l'emploi, des animations et des effets visuels pour créer des interfaces modernes et attrayantes.

Première version : 2014 Version la plus récente : 2021

Site officiel : <https://materializecss.com/>

Documentation : <https://materializecss.com/getting-started.html>

Tutoriel : <https://materializecss.com/crash-course.html>

Vous pouvez explorer les sites officiels, consulter la documentation et suivre les tutoriels pour en apprendre davantage sur chaque framework et découvrir comment les utiliser efficacement dans vos projets.

VI. Sources

1. Sites de veille technologique pour le CSS :

SS-Tricks : <https://css-tricks.com/>

Smashing Magazine : <https://www.smashingmagazine.com/category/css/>

2. Documentation officielle de Bootstrap :

- Site web de Bootstrap : <https://getbootstrap.com/>
- Documentation de Bootstrap : <https://getbootstrap.com/docs/>