

Les containers

Les classes `.container` et `.container-fluid` sont des éléments fondamentaux du système de mise en page de Bootstrap. Elles servent de conteneurs principaux pour structurer votre contenu et le rendre réactif en s'adaptant à différentes tailles d'écran. Cet article explore en détail ces classes, leur rôle, leurs différences, et comment les utiliser efficacement pour créer des mises en page modernes et adaptatives.

4TTR

 Intermédiaire

Les Bases des Conteneurs Bootstrap

Un **conteneur** est un élément HTML qui encapsule le contenu de votre site. Dans Bootstrap, il existe deux types principaux de conteneurs :

1. `.container` : Ce conteneur a une largeur fixe qui varie selon la taille de l'écran.
2. `.container-fluid` : Ce conteneur s'étend toujours sur 100% de la largeur de la fenêtre, quelle que soit sa taille.

Les conteneurs sont essentiels car ils servent de base pour organiser les colonnes et les rangées de la grille Bootstrap.

Utilisation de `.container`

La classe `.container` est idéale lorsque vous souhaitez que votre contenu soit centré et limité en largeur pour garantir une lisibilité optimale. Elle s'adapte automatiquement aux différentes tailles d'écran selon les points de rupture (breakpoints) définis dans Bootstrap.

Largeurs des Conteneurs Fixes

Pour info, les largeurs par défaut de `.container` sont définies comme suit :

Taille de l'écran	Breakpoint	Largeur du conteneur
Extra small (xs)	< 576px	Auto
Small (sm)	≥ 576px	540px

Taille de l'écran	Breakpoint	Largeur du conteneur
Medium (md)	≥ 768px	720px
Large (lg)	≥ 992px	960px
Extra large (xl)	≥ 1200px	1140px
Extra extra large (xxl)	≥ 1400px	1320px

Exemple de Code

Voici un exemple d'utilisation de `.container` :

```

1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Exemple de Conteneur</title>
7    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="st
8  </head>
9  <body>
10   <div class="container">
11     <h1>Bienvenue dans Bootstrap</h1>
12     <p>Ceci est un conteneur fixe qui s'adapte à la taille de l'écran.</p>
13   </div>
14 </body>
15 </html>

```

Utilisation de `.container-fluid`

La classe `.container-fluid` crée un conteneur qui s'étend sur toute la largeur de l'écran, quelle que soit la taille de ce dernier. Elle est idéale pour les mises en page pleine largeur, comme des en-têtes ou des pieds de page qui occupent tout l'espace horizontal.

Exemple de Code

```

1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Exemple de Conteneur Fluide</title>
7    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="st
8  </head>

```

```

9    <body>
10   <div class="container-fluid bg-primary text-white">
11     <h1 class="text-center">Mise en page pleine largeur</h1>
12     <p class="text-center">Ce conteneur occupe toute la largeur de l'écran.</p>
13   </div>
14 </body>
15 </html>

```

Comparaison Visuelle : `.container` vs `.container-fluid`

Caractéristique	<code>.container</code>	<code>.container-fluid</code>
Largeur	Fixe (varie par breakpoint)	100% de la largeur
Utilisation	Contenu centré et limité	Contenu pleine largeur
Adaptabilité	Oui, selon les breakpoints	Oui, toujours fluide

Applications Pratiques

En-Têtes et Pieds de Page Pleine Largeur : Utilisez `.container-fluid` pour les sections qui doivent occuper toute la largeur de l'écran.

Sections de Contenu Centré : Utilisez `.container` pour les sections principales afin de centrer et limiter la largeur du contenu.

Mise en Page Réactive : Combinez les breakpoints et les conteneurs pour ajuster dynamiquement les mises en page en fonction de la taille de l'écran.

Conclusion

Les classes `.container` et `.container-fluid` de Bootstrap 5 sont des outils puissants pour structurer et organiser vos mises en page. Tandis que `.container` offre une largeur fixe et centrée idéale pour un contenu bien structuré, `.container-fluid` s'adapte à toute la largeur de l'écran, parfait pour des designs immersifs et dynamiques. En maîtrisant ces outils, vous pouvez créer des mises en page réactives et esthétiques adaptées à tous les appareils.

