

Aww ouais, Bootstrap 4 est à venir!

CSS

Les paramètres globaux CSS, les éléments HTML fondamentaux style et améliorés avec des classes extensibles, et un système de grille avancée.



Le Cloud n'a pas à être nuageux. Obtenez le Cloud avec Rien à cacher!

Annonces via Carbon

aperçu

Obtenez la vérité sur les éléments clés de l'infrastructure de Bootstrap, y compris notre approche mieux plus rapidement, le développement, plus fort web.

HTML5 doctype

Bootstrap fait usage de certains éléments HTML et des propriétés CSS qui nécessitent l'utilisation de la doctype HTML5. Inclure au début de tous vos projets.

```
<!DOCTYPE html>
<html lang="en">
...
</html>
```

mobile premier

Avec Bootstrap 2, nous avons ajouté des styles amicaux mobiles en option pour les aspects clés du cadre. Avec Bootstrap 3, nous avons réécrit le projet pour être mobile convivial dès le début. Au lieu d'ajouter des styles mobiles en option, ils sont cuits directement dans le noyau. En fait, **Bootstrap est premier mobile**. Mobiles premiers styles peuvent être trouvés dans l'ensemble de la bibliothèque plutôt que dans des fichiers séparés.

Afin d'assurer un rendu correct et toucher le zoom, **ajouter la balise meta viewport** à votre `<head>`.

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

Vous pouvez désactiver les fonctionnalités de zoom sur les appareils mobiles en ajoutant `user-scalable=no` la balise meta viewport. Cela désactive le zoom, ce qui signifie les utilisateurs ne sont en mesure de faire défiler, et les résultats dans votre site se sentir un peu plus comme une application native. Dans l'ensemble, nous ne recommandons pas ce sur chaque site, donc faites attention!

```
<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no">
```

Typographie et liens

Bootstrap fixe de base globaux styles d'affichage, typographie, et de liaison. Plus précisément, nous:

- Situé `background-color: #fff;` sur la `body`
- Utilisez les `@font-family-base`, `@font-size-base` et les `@line-height-base` attributs que notre base typographique

- Régler la couleur globale de lien via `@link-color` et appliquer le lien souligne que sur `:hover`

Ces styles peuvent être trouvés dans `scaffolding.less`.

Normalize.css

Pour un meilleur rendu cross-browser, nous utilisons Normalize.css , un projet par Nicolas Gallagher et Jonathan Neal .

Containers

Bootstrap nécessite un élément contenant pour envelopper le contenu du site et accueillir notre système de grille. Vous pouvez choisir l' un des deux conteneurs à utiliser dans vos projets. Notez que, en raison de `padding` plus, ni récipient est emboîtable.

Utiliser `.container` pour un conteneur de largeur fixe sensible.

```
<div class="container">
  ...
</div>
```

Utiliser `.container-fluid` pour un conteneur de largeur, couvrant toute la largeur de votre fenêtre.

```
<div class="container-fluid">
  ...
</div>
```

Système de grille

Bootstrap comprend un mobile premier système de grille souple, fluide qui évolue de manière appropriée jusqu'à 12 colonnes en tant que périphérique ou viewport taille augmente. Il comprend des classes prédéfinies pour les options de mise en page facile, ainsi que de puissants mixins pour générer dispositions plus sémantiques .

introduction

systèmes de grille sont utilisés pour créer des mises en page par le biais d'une série de lignes et de colonnes qui abritent votre contenu. Voici comment le système de grille Bootstrap fonctionne:

- Les lignes doivent être placées dans un `.container` (largeur fixe) ou `.container-fluid` (pleine largeur) pour l' alignement et le rembourrage approprié.
- Utilisez les lignes pour créer des groupes horizontaux de colonnes.
- Le contenu devrait être placé à l'intérieur des colonnes, et seules les colonnes peuvent être des enfants immédiats de lignes.
- Classes de grille prédéfinis comme `.row` et `.col-xs-4` sont disponibles pour faire rapidement mises en page de la grille. Moins mixins peuvent également être utilisés pour plus de dispositions sémantiques.
- Colonnes créent des gouttières (lacunes entre le contenu de la colonne) par l' intermédiaire `padding` . Ce rembourrage est compensé en rangées pour la première et dernière colonne via marge négative sur `.row`s.
- La marge négative est pourquoi les exemples ci-dessous sont outdented. Il est pour que le contenu dans les colonnes de la grille est alignée avec le contenu non-grille.
- Colonnes de grille sont créées en spécifiant le nombre de douze colonnes disponibles que vous souhaitez étendre. Par exemple, trois colonnes égales utiliseraient trois `.col-xs-4` .
- Si plus de 12 colonnes sont placées dans une seule rangée, chaque groupe de colonnes supplémentaires seront, comme une seule unité, envelopper sur une nouvelle ligne.
- Classes de grille appliquent aux appareils avec des largeurs d'écran supérieure ou égale à la taille des points d' arrêt, et les classes de la grille de remplacement destinés aux appareils plus petits. Par conséquent, par exemple , l' application de toute `.col-md-*` classe à un élément non seulement affecte son style sur les appareils moyens , mais aussi sur de grands appareils si une `.col-lg-*` classe est pas présent.

Regardez les exemples d'application de ces principes à votre code.

requêtes des médias

Nous utilisons les requêtes des médias suivants dans nos fichiers pour créer les points d'arrêt clés de notre système de grille.	Extra petits appareils portables (<768px)	Petits appareils Tablets (≥768px)	Appareils Moyen Desktops (≥992px)	Les grands appareils de bureau (≥1200px)
/* Extra small devices (phones, less than 768px) */				
/* No media query since this is the default in Bootstrap */				
/* Small devices (tablets, 768px and up) */				
@media (min-width: @screen-sm-min) { ... }				
/* Medium devices (desktops, 992px and up) */				
@media (min-width: @screen-md-min) { ... }				
/* Large devices (large desktops, 1200px and up) */				
@media (min-width: @screen-lg-min) { ... }				

Nous étendons occasionnellement sur ces questions des médias à inclure une `max-width` de limiter CSS à un ensemble plus restreint de dispositifs.

```
@media (max-width: @screen-xs-max) { ... }
@media (min-width: @screen-sm-min) and (max-width: @screen-sm-max) { ... }
@media (min-width: @screen-md-min) and (max-width: @screen-md-max) { ... }
@media (min-width: @screen-lg-min) { ... }
```

Options de la grille

Voir comment les aspects du travail du système de grille Bootstrap sur plusieurs appareils avec une table à portée de main.

	Extra petits appareils portables (<768px)	Petits appareils Tablets (≥768px)	Appareils Moyen Desktops (≥992px)	Les grands appareils de bureau (≥1200px)
le comportement de la grille	Horizontal en tout temps	Collapsed pour commencer, au-dessus des points d'arrêt horizontaux		
Largeur de conteneurs	Aucun (auto)	750px	970 pixels	1170px
préfixe de classe	.col-xs-	.col-sm-	.col-md-	.col-lg-
Nombre de colonnes	12			
Largeur de colonne	Auto	~ 62px	~ 81px	~ 97px
Largeur Gutter	30px (15px de chaque côté d'une colonne)			
Nestable	Oui			
décalages	Oui			
la commande de la colonne	Oui			

Exemple: Stacked à horizontale

L'utilisation d'un seul ensemble de `.col-md-*` classes de grille, vous pouvez créer un système de grille de base qui commence à empiler sur les appareils mobiles et les tablettes (extra petit à petit intervalle) avant de devenir horizontale sur le bureau (moyenne) appareils. Colonnes Placer la grille dans tout `.row`.

```
.col-md-1
```

```
.col-md-8
```

```
.col-md-4
```

```
.col-md-4
```

```
.col-md-4
```

```
.col-md-4
```

```
.col-md-6
```

```
.col-md-6
```

```
<div class="row">
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
</div>
<div class="row">
  <div class="col-md-8">.col-md-8</div>
  <div class="col-md-4">.col-md-4</div>
</div>
<div class="row">
  <div class="col-md-4">.col-md-4</div>
  <div class="col-md-4">.col-md-4</div>
  <div class="col-md-4">.col-md-4</div>
</div>
<div class="row">
  <div class="col-md-6">.col-md-6</div>
  <div class="col-md-6">.col-md-6</div>
</div>
```

Exemple: conteneur de liquide

Tournez toute disposition de grille-largeur fixe dans une mise en page pleine largeur en changeant votre plus à l' extérieur .container de .container-fluid .

```
<div class="container-fluid">
  <div class="row">
    ...
  </div>
</div>
```

Exemple: Mobile et bureau

Vous ne voulez pas vos colonnes simplement empiler dans des dispositifs plus petits? Utilisez les petites et moyennes classes de grille de dispositif supplémentaire en ajoutant `.col-xs-* .col-md-*` à vos colonnes. Voir l'exemple ci - dessous pour une meilleure idée de la façon dont tout cela fonctionne.

`.col-xs-12 .col-md-8`

`.col-xs-6 .col-md-4`

`.col-xs-6 .col-md-4`

`.col-xs-6 .col-md-4`

`.col-xs-6 .col-md-4`

`.col-xs-6`

```
<!-- Stack the columns on mobile by making one full-width and the other half-width -->
<div class="row">
  <div class="col-xs-12 col-md-8">.col-xs-12 .col-md-8</div>
  <div class="col-xs-6 col-md-4">.col-xs-6 .col-md-4</div>
</div>

<!-- Columns start at 50% wide on mobile and bump up to 33.3% wide on desktop -->
<div class="row">
  <div class="col-xs-6 col-md-4">.col-xs-6 .col-md-4</div>
  <div class="col-xs-6 col-md-4">.col-xs-6 .col-md-4</div>
  <div class="col-xs-6 col-md-4">.col-xs-6 .col-md-4</div>
</div>

<!-- Columns are always 50% wide, on mobile and desktop -->
<div class="row">
  <div class="col-xs-6">.col-xs-6</div>
  <div class="col-xs-6">.col-xs-6</div>
</div>
```

Exemple: Mobile, tablette, ordinateur

Miser sur l'exemple précédent en créant des dispositions encore plus dynamiques et puissantes avec tablettes `.col-sm-*` classes.

`.col-xs-12 .col-sm-6 .col-md-8`

`.col-xs-6 .col-md-4`

`.col-xs-6 .col-sm-4`

`.col-xs-6 .col-sm-4`

```
<div class="row">
  <div class="col-xs-12 col-sm-6 col-md-8">.col-xs-12 .col-sm-6 .col-md-8</div>
  <div class="col-xs-6 col-md-4">.col-xs-6 .col-md-4</div>
</div>
<div class="row">
  <div class="col-xs-6 col-sm-4">.col-xs-6 .col-sm-4</div>
  <div class="col-xs-6 col-sm-4">.col-xs-6 .col-sm-4</div>
  <!-- Optional: clear the XS cols if their content doesn't match in height -->
  <div class="clearfix visible-xs-block"></div>
  <div class="col-xs-6 col-sm-4">.col-xs-6 .col-sm-4</div>
</div>
```

Exemple: Colonne emballage

Si plus de 12 colonnes sont placées dans une seule rangée, chaque groupe de colonnes supplémentaires seront, comme une seule unité, envelopper sur une nouvelle ligne.

.col-xs-9

.col-xs-4

Depuis $9 + 4 = 13 > 12$, ce 4-colonne large div obtient enveloppé sur une nouvelle ligne comme une unité contiguë.

.col-xs-6

colonnes suivantes continuent le long de la nouvelle ligne.

```
<div class="row">
  <div class="col-xs-9">.col-xs-9</div>
  <div class="col-xs-4">.col-xs-4<br>Since 9 + 4 = 13 > 12, this 4-column-wide div gets wrapped onto a new line as one contiguous unit.</div>
  <div class="col-xs-6">.col-xs-6<br>Subsequent columns continue along the new line.</div>
</div>
```

réinitialise colonne Responsive

Avec les quatre niveaux de grilles disponibles , vous êtes lié à rencontrer des problèmes où, à certains points d' arrêt, vos colonnes ne désactivez pas tout à fait raison que l' on est plus grand que l'autre. Pour résoudre ce problème, utilisez une combinaison d'un `.clearfix` et de nos classes utilitaires sensibles .

.col-xs-6 .col-sm-3

Redimensionner votre fenêtre ou le vérifier sur votre téléphone pour un exemple.

.col-xs-6 .col-sm-3

.col-xs-6 .col-sm-3

.col-xs-6 .col-sm-3

```
<div class="row">
  <div class="col-xs-6 col-sm-3">.col-xs-6 .col-sm-3</div>
  <div class="col-xs-6 col-sm-3">.col-xs-6 .col-sm-3</div>

  <!-- Add the extra clearfix for only the required viewport -->
  <div class="clearfix visible-xs-block"></div>

  <div class="col-xs-6 col-sm-3">.col-xs-6 .col-sm-3</div>
  <div class="col-xs-6 col-sm-3">.col-xs-6 .col-sm-3</div>
</div>
```

En plus de la colonne de compensation à des points d' arrêt sensibles, vous devrez peut - être **réinitialiser les décalages, pousse ou tire** . Voir en action dans l'exemple de la grille .

```
<div class="row">
  <div class="col-sm-5 col-md-6">.col-sm-5 .col-md-6</div>
  <div class="col-sm-5 col-sm-offset-2 col-md-6 col-md-offset-0">.col-sm-5 .col-sm-offset-2 .col-md-6 .col-md-offset-0</div>
</div>

<div class="row">
  <div class="col-sm-6 col-md-5 col-lg-6">.col-sm-6 .col-md-5 .col-lg-6</div>
  <div class="col-sm-6 col-md-5 col-md-offset-2 col-lg-6 col-lg-offset-0">.col-sm-6 .col-md-5 .col-md-offset-2 .col-lg-6 .col-lg-offset-0</div>
</div>
```

colonnes Compenser

Déplacez des colonnes vers la droite en utilisant les `.col-md-offset-*` classes. Ces classes augmentent la marge de gauche d'une colonne en * colonnes. Par exemple, `.col-md-offset-4` se déplace `.col-md-4` sur quatre colonnes.

`.col-md-4`

`.col-md-4 .col-md-offset-4`

`.col-md-3 .col-md-offset-3`

`.col-md-3 .col-md-offset-3`

`.col-md-6 .col-md-offset-3`

```
<div class="row">
  <div class="col-md-4">.col-md-4</div>
  <div class="col-md-4 col-md-offset-4">.col-md-4 .col-md-offset-4</div>
</div>
<div class="row">
  <div class="col-md-3 col-md-offset-3">.col-md-3 .col-md-offset-3</div>
  <div class="col-md-3 col-md-offset-3">.col-md-3 .col-md-offset-3</div>
</div>
<div class="row">
  <div class="col-md-6 col-md-offset-3">.col-md-6 .col-md-offset-3</div>
</div>
```

Vous pouvez également remplacer les décalages de niveaux inférieurs de la grille avec les `.col-*-offset-0` classes.

```
<div class="row">
  <div class="col-xs-6 col-sm-4">
  </div>
  <div class="col-xs-6 col-sm-4">
  </div>
  <div class="col-xs-6 col-xs-offset-3 col-sm-4 col-sm-offset-0">
  </div>
</div>
```

colonnes Nesting

Pour imbriquer votre contenu avec le réseau par défaut, ajouter une nouvelle `.row` et un ensemble de `.col-sm-*` colonnes dans une existante `.col-sm-*` colonne. Lignes imbriquées devraient inclure un ensemble de colonnes qui ajoutent jusqu'à 12 ou moins (il est pas nécessaire que vous utilisez tous les 12 colonnes disponibles).

Niveau 1: `.col-sm-9`

Niveau 2: `.col-xs-8 .col-sm-6`

Niveau 2: `.col-xs-4 .col-sm-6`

```
<div class="row">
  <div class="col-sm-9">
    Level 1: .col-sm-9
    <div class="row">
      <div class="col-xs-8 col-sm-6">
        Level 2: .col-xs-8 .col-sm-6
      </div>
      <div class="col-xs-4 col-sm-6">
        Level 2: .col-xs-4 .col-sm-6
      </div>
    </div>
  </div>
</div>
```

la commande de la colonne

Changez facilement l'ordre de nos intégrés dans les colonnes de la grille avec `.col-md-push-*` et `.col-md-pull-*` classes de modification.

```
.col-md-9 .col-md-push-3
.col-md-3 .col-md-pull-9
```

```
<div class="row">
  <div class="col-md-9 col-md-push-3">.col-md-9 .col-md-push-3</div>
  <div class="col-md-3 col-md-pull-9">.col-md-3 .col-md-pull-9</div>
</div>
```

Moins mixins et variables

En plus des classes de grille préconstruits pour mise en page rapide, Bootstrap comprend moins variables et mixins pour générer rapidement vos propres dispositions simples, sémantiques.

Variables

Variables déterminent le nombre de colonnes, la largeur de la gouttière, et le point à partir duquel commencer colonnes flottantes de requête des médias. Nous utilisons pour générer les classes de grille prédéfinies documentées ci-dessus, ainsi que pour les mixins personnalisés ci-dessous.

```
@grid-columns:          12;
@grid-gutter-width:     30px;
@grid-float-breakpoint: 768px;
```

mixins

Mixins sont utilisés en conjonction avec les variables de grille pour générer CSS sémantique pour les colonnes de grille individuelles.

```
// Creates a wrapper for a series of columns
.make-row(@gutter: @grid-gutter-width) {
  // Then clear the floated columns
  .clearfix();

  @media (min-width: @screen-sm-min) {
    margin-left: (@gutter / -2);
    margin-right: (@gutter / -2);
  }

  // Negative margin nested rows out to align the content of
  // columns
  .row {
    margin-left: (@gutter / -2);
    margin-right: (@gutter / -2);
  }
}

// Generate the extra small columns
.make-xs-column(@columns; @gutter: @grid-gutter-width) {
  position: relative;
  // Prevent columns from collapsing when empty
  min-height: 1px;
  // Inner gutter via padding
  padding-left: (@gutter / 2);
  padding-right: (@gutter / 2);

  // Calculate width based on number of columns available
  @media (min-width: @grid-float-breakpoint) {
    float: left;
    width: percentage((@columns / @grid-columns));
  }
}

// Generate the small columns
.make-sm-column(@columns; @gutter: @grid-gutter-width) {
  position: relative;
  // Prevent columns from collapsing when empty
  min-height: 1px;
  // Inner gutter via padding
  padding-left: (@gutter / 2);
  padding-right: (@gutter / 2);

  // Calculate width based on number of columns available
  @media (min-width: @screen-sm-min) {
    float: left;
    width: percentage((@columns / @grid-columns));
  }
}

// Generate the small column offsets
.make-sm-column-offset(@columns) {
  @media (min-width: @screen-sm-min) {
    margin-left: percentage((@columns / @grid-columns));
  }
}

.make-sm-column-push(@columns) {
  @media (min-width: @screen-sm-min) {
    left: percentage((@columns / @grid-columns));
  }
}

.make-sm-column-pull(@columns) {
  @media (min-width: @screen-sm-min) {
    right: percentage((@columns / @grid-columns));
  }
}

// Generate the medium columns
.make-md-column(@columns; @gutter: @grid-gutter-width) {
  position: relative;
```

```

position: relative;
// Prevent columns from collapsing when empty
min-height: 1px;
// Inner gutter via padding
padding-left: (@gutter / 2);
padding-right: (@gutter / 2);

// Calculate width based on number of columns available
@media (min-width: @screen-md-min) {
  float: left;
  width: percentage(@columns / @grid-columns);
}

// Generate the medium column offsets
.make-md-column-offset(@columns) {
  @media (min-width: @screen-md-min) {
    margin-left: percentage(@columns / @grid-columns);
  }
}
.make-md-column-push(@columns) {
  @media (min-width: @screen-md-min) {
    left: percentage(@columns / @grid-columns);
  }
}
.make-md-column-pull(@columns) {
  @media (min-width: @screen-md-min) {
    right: percentage(@columns / @grid-columns);
  }
}

// Generate the large columns
.make-lg-column(@columns; @gutter: @grid-gutter-width) {
  position: relative;
  // Prevent columns from collapsing when empty
  min-height: 1px;
  // Inner gutter via padding
  padding-left: (@gutter / 2);
  padding-right: (@gutter / 2);

  // Calculate width based on number of columns available
  @media (min-width: @screen-lg-min) {
    float: left;
    width: percentage(@columns / @grid-columns);
  }
}

// Generate the large column offsets
.make-lg-column-offset(@columns) {
  @media (min-width: @screen-lg-min) {
    margin-left: percentage(@columns / @grid-columns);
  }
}
.make-lg-column-push(@columns) {
  @media (min-width: @screen-lg-min) {
    left: percentage(@columns / @grid-columns);
  }
}
.make-lg-column-pull(@columns) {
  @media (min-width: @screen-lg-min) {
    right: percentage(@columns / @grid-columns);
  }
}

```

Exemple d'utilisation

Vous pouvez modifier les variables à vos propres valeurs personnalisées, ou tout simplement utiliser les mixins avec leurs valeurs par défaut. Voici un exemple d'utilisation des paramètres par défaut pour créer une mise en page sur deux colonnes avec un écart entre les deux.

```
.wrapper {
  .make-row();
}
.content-main {
  .make-lg-column(8);
}
.content-secondary {
  .make-lg-column(3);
  .make-lg-column-offset(1);
}
```

```
<div class="wrapper">
  <div class="content-main">...</div>
  <div class="content-secondary">...</div>
</div>
```

Typographie

Rubriques

Toutes les rubriques HTML, `<h1>` par le biais `<h6>`, sont disponibles. `.h1` Par le biais des `.h6` classes sont également disponibles, pour quand vous voulez faire correspondre le style de police d'un titre, mais veulent toujours votre texte à afficher en ligne.

EXAMPLE

h1. rubrique Bootstrap

36px Semibold

h2. rubrique Bootstrap

30px Semibold

h3. rubrique Bootstrap

24px Semibold

h4. rubrique Bootstrap

18px Semibold

h5. rubrique Bootstrap

14px Semibold

h6. rubrique Bootstrap

12px Semibold

```
<h1>h1. Bootstrap heading</h1>
<h2>h2. Bootstrap heading</h2>
<h3>h3. Bootstrap heading</h3>
<h4>h4. Bootstrap heading</h4>
<h5>h5. Bootstrap heading</h5>
<h6>h6. Bootstrap heading</h6>
```

Créer plus léger, le texte secondaire dans toute position avec un générique `<small>` tag ou de la `.small` classe.

EXAMPLE

h1. Bootstrap rubrique texte secondaire

h2. Bootstrap rubrique texte secondaire

h3. Bootstrap rubrique texte secondaire

h4. Bootstrap rubrique texte secondaire

h5. Bootstrap rubrique texte secondaire

h6. Bootstrap rubrique texte secondaire

```
<h1>h1. Bootstrap heading <small>Secondary text</small></h1>
<h2>h2. Bootstrap heading <small>Secondary text</small></h2>
<h3>h3. Bootstrap heading <small>Secondary text</small></h3>
<h4>h4. Bootstrap heading <small>Secondary text</small></h4>
<h5>h5. Bootstrap heading <small>Secondary text</small></h5>
<h6>h6. Bootstrap heading <small>Secondary text</small></h6>
```

copie du corps

Défaut global de Bootstrap `font-size` est **14px**, avec `line-height` des **1.428**. Ceci est appliqué aux `<body>` et tous les paragraphes. En outre, `<p>` (paragraphes) reçoivent une marge inférieure de la moitié de leur hauteur de ligne calculée (10px par défaut).

EXAMPLE

Nullam quis risus eget urna mollis ornare vel eu leo. Cum sociis natoque penatibus et magnis dis parturiente montes, nascetur ridiculus Mus. Nullam id dolor ultricies id NIBH vehicula.

Cum sociis natoque penatibus et magnis dis parturiente montes, nascetur ridiculus Mus. Donec ullamcorper nulla non metus auctor fringilla. mollis de Duis, HNE luctus non commodo, nisi erat porttitor Ligula, eget lacinia odio sem nca elit. Donec ullamcorper nulla non metus auctor fringilla.

Mécène sed diam eget risus varius blandit sit amet non magna. Donec id elit non mi porta gravida à eget metus. mollis de Duis, HNE luctus non commodo, nisi erat porttitor Ligula, eget lacinia odio sem nca elit.

`<p>...</p>`

Lead copie de corps

Faire un paragraphe se démarquer en ajoutant `.lead`.

EXAMPLE

Vivamus sagittis lacus vel augue laoreet rutrum faucibus dolor auctor. mollis de Duis, is non luctus de commodo.

`<p class="lead">...</p>`

Construit avec moins

L'échelle typographique est basée sur deux variables Moins **variables.less** : `@font-size-base` et `@line-height-base`. Le premier est la taille de police de base utilisée partout et la seconde est la hauteur de ligne de base. Nous utilisons ces variables et quelques calculs simples pour créer les marges, rembourrages, et les files d'hauteurs de toute notre nature et plus encore. Personnalisez - les et adapte bootstrap.

Inline éléments de texte

Le texte marqué

Pour mettre en évidence une série de texte en raison de sa pertinence dans un autre contexte, utilisez la `<mark>` balise.

EXAMPLE

Vous pouvez utiliser la balise de marque pour surligner texte.

You can use the `mark` tag to `<mark>highlight</mark>` text.

Le texte supprimé

Pour indiquer des blocs de texte qui ont été supprimés l' utilisation de la `` balise.

EXAMPLE

Cette ligne de texte est destiné à être traité comme texte supprimé.

`This line of text is meant to be treated as deleted text.`

texte Strikethrough

Pour indiquer des blocs de texte qui ne sont plus pertinentes d' usage sont le `<s>` tag.

EXAMPLE

Cette ligne de texte est destiné à être traité comme plus précis.

`<s>This line of text is meant to be treated as no longer accurate.</s>`

Le texte inséré

Pour indiquer les ajouts au document en utilisant la `<ins>` balise.

EXAMPLE

Cette ligne de texte est destiné à être traité comme un ajout au document.

`<ins>This line of text is meant to be treated as an addition to the document.</ins>`

texte souligné

Pour souligner le texte utiliser la `<u>` balise.

EXAMPLE

Cette ligne de texte render comme l'a souligné

`<u>This line of text will render as underlined</u>`

Exploitez les balises d'accentuation par défaut de HTML avec des styles légers.

Petit texte

Pour de-mettant l' accent en ligne ou des blocs de texte, utilisez la `<small>` balise pour définir le texte à 85% de la taille de la société mère. Éléments Rubrique reçoivent leur propre `font-size` pour emboîtés `<small>` éléments.

Vous pouvez également utiliser un élément en ligne avec `.small` à la place de tout `<small>`.

EXAMPLE

Cette ligne de texte est destiné à être traité comme petits caractères.

`<small>This line of text is meant to be treated as fine print.</small>`

Audacieux

Pour en insistant sur un extrait de texte avec une police-poids plus lourd.

EXAMPLE

L'extrait suivant du texte est **rendu en tant que texte en gras**.

```
<strong>rendered as bold text</strong>
```

Italique

Pour privilégier un extrait de texte avec italique.

EXAMPLE

L'extrait suivant du texte est *rendu sous forme de texte en italique*.

```
<em>rendered as italicized text</em>
```

Autres éléments

Ne hésitez pas à utiliser **** et **<i>** en HTML5. **** Est destiné à mettre en évidence des mots ou des phrases sans transporter importance supplémentaire tout **<i>** est surtout pour la voix, les termes techniques, etc.

classes d'alignement

réaligner facilement le texte à des composants avec des classes d'alignement du texte.

EXAMPLE

Gauche texte aligné.

Centre aligné texte.

Aligné à droite du texte.

Texte justifié.

Aucun texte d'emballage.

```
<p class="text-left">Left aligned text.</p>
<p class="text-center">Center aligned text.</p>
<p class="text-right">Right aligned text.</p>
<p class="text-justify">Justified text.</p>
<p class="text nowrap">No wrap text.</p>
```

classes de transformation

Transformer le texte dans les composants avec des classes texte de capitalisation.

EXAMPLE

texte en minuscules.

TEXTE EN MAJUSCULE.

Texte Capitalisée.

```
<p class="text-lowercase">Lowercased text.</p>
<p class="text-uppercase">Uppercased text.</p>
<p class="text-capitalize">Capitalized text.</p>
```

Abréviations

La mise en œuvre de Stylisé de HTML `<abbr>` de l' élément pour les abréviations et acronymes pour montrer la version élargie sur le vol stationnaire. Les abréviations avec un `title` attribut ont une bordure inférieure de la lumière en pointillés et un curseur d' aide sur le vol stationnaire, en fournissant un contexte supplémentaire sur le vol stationnaire et pour les utilisateurs de technologies d' assistance.

abréviation de base

EXAMPLE

Une abréviation de l'attribut de mot est `attr (attribut)` .

```
<abbr title="attribute">attr</abbr>
```

Initiales

Ajouter `.initialism` à une abréviation pour une police de taille légèrement plus petite.

EXAMPLE

HTML (LANGAGE SIGNALÉTIQUE HYPER TEXT) est la meilleure chose depuis le pain tranché.

```
<abbr title="HyperText Markup Language" class="initialism">HTML</abbr>
```

Adresses

Coordonnées Présent pour l'ancêtre le plus proche ou le corps entier de travail. Préserver le formatage en mettant fin à toutes les lignes avec `
` .

EXAMPLE

Twitter, Inc.

1355 Market Street, Suite 900
San Francisco, CA 94103 P: (Phone) (123) 456-7890
(Téléphone)

Nom complet first.last@example.com

```
<address>
  <strong>Twitter, Inc.</strong><br>
  1355 Market Street, Suite 900<br>
  San Francisco, CA 94103<br>
  <abbr title="Phone">P:</abbr> (123) 456-7890
</address>

<address>
  <strong>Full Name</strong><br>
  <a href="mailto:#">first.last@example.com</a>
</address>
```

Blocs de Texte

Pour citer des blocs de contenu à partir d'une autre source au sein de votre document.

Par défaut blockquote

Enrouler `<blockquote>` autour de tout HTML (Langage Signalétique Hyper Text) comme la citation. Pour des citations droites, nous recommandons un `<p>` .

EXAMPLE

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Entier posuere erat ante.

```
<blockquote>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer posuere erat a ante.</p>
</blockquote>
```

Options Blockquote

Style et contenu change pour de simples variations sur une norme `<blockquote>`.

Nommer une source

Ajouter un `<footer>` pour identifier la source. Enroulez le nom de l'œuvre de source `<cite>`.

EXAMPLE

```
  Lorem ipsum dolor sit amet, consectetur adipiscing elit. Entier posuere erat ante.
```

```
  — Quelqu'un célèbre Source Titre
```

```
<blockquote>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer posuere erat a ante.</p>
  <footer>Someone famous in <cite title="Source Title">Source Title</cite></footer>
</blockquote>
```

Autres affichages

Ajouter `.blockquote-reverse` un blockquote avec un contenu aligné à droite.

EXAMPLE

```
  Lorem ipsum dolor sit amet, consectetur adipiscing elit. Entier posuere erat ante.
```

```
  Quelqu'un célèbre Source Titre —
```

```
<blockquote class="blockquote-reverse">
  ...
</blockquote>
```

Listes

Unordered

Une liste d'éléments dont l'ordre ne *pas* explicitement d' importance.

EXAMPLE

- Lorem ipsum dolor sit amet
- Consectetur adipiscing elit
- Integer lorem molestie à massa
- Facilisis dans aliquet nisl pretium
- Nulla volutpat aliquam velit
 - Phasellus iaculis neque
 - Purus sodales ultricies
 - porttitor MEB de Vestibulum
 - Ac tristique libero volutpat au
- porta Faucibus lacus fringilla vel
- Aenean sit amet erat nunc
- Eget porttitor lorem

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

Commandé

Une liste d'éléments dont l'ordre n'importe explicitement.

EXAMPLE

1. Lorem ipsum dolor sit amet
2. Consectetur adipiscing elit
3. Integer lorem molestie à massa
4. Facilisis dans aliquet nisl pretium
5. Nulla volutpat aliquam velit
6. porta Faucibus lacus fringilla vel
7. Aenean sit amet erat nunc
8. Eget porttitor lorem

```
<ol>
  <li>...</li>
</ol>
```

sans style

Retirez le défaut `list-style` et la marge gauche sur la liste des articles (seuls les enfants immédiats). **Cela ne concerne que immédiats éléments de la liste des enfants**, ce qui signifie que vous devrez ajouter la classe pour toutes les listes imbriquées ainsi.

EXAMPLE

Lorem ipsum dolor sit amet
 Consectetur adipiscing elit
 Integer lorem molestie à massa
 Facilisis dans aliquet nisl pretium
 Nulla volutpat aliquam velit

- Phasellus iaculis neque
- Purus sodales ultricies
- porttitor MEB de Vestibulum
- Ac tristique libero volutpat au

 porta Faucibus lacus fringilla vel
 Aenean sit amet erat nunc
 Eget porttitor lorem

```
<ul class="list-unstyled">
  <li>...</li>
</ul>
```

En ligne

Placez tous les éléments de la liste sur une seule ligne avec `display: inline-block;` et certains léger rembourrage.

EXAMPLE

Lorem ipsum Phasellus iaculis Nulla volutpat

```
<ul class="list-inline">
  <li>...</li>
</ul>
```

La description

Une liste de termes avec leurs descriptions associées.

EXAMPLE

description de listes

Une liste de description est parfait pour la définition des termes.

euismod

Vestibulum id ligula porta felis euismod eget semper lacinia odio sem nca elit.

Donec id elit non mi porta gravida à eget metus.

Malesuada porta

Etiam Malesuada magna porta sem mollis euismod.

```
<dl>
  <dt>...</dt>
  <dd>...</dd>
</dl>
```

Description Horizontal

Faire termes et descriptions de `<dl>` line up side-by-side. Démarre off empilés comme défaut `<dl>`s, mais quand la barre de navigation se développe, de même ces.

EXAMPLE**description de listes**

Une liste de description est parfait pour la définition des termes.

euismod

Vestibulum id ligula porta felis euismod eget semper lacinia odio sem nca elit.

Donec id elit non mi porta gravida à eget metus.

Malesuada porta

Etiam Malesuada magna porta sem mollis euismod.

Felis euismod eget semper lacinia

Fusce dapibus, tellus ac Cursus commodo, tortor mauris condimentum nibh, fermentum ut massa justo sit amet risus.

```
<dl class="dl-horizontal">
  <dt>...</dt>
  <dd>...</dd>
</dl>
```

Auto-tronquant

Description horizontales listes tronquer termes trop long pour tenir dans la colonne de gauche avec `text-overflow`. Dans les fenêtres plus étroites, ils vont changer la mise en page par défaut empilés.

Code

En ligne

Enveloppez extraits inline de code avec `<code>`.

EXAMPLE

Par exemple, `<section>` doit être enveloppé inline.

For example, `<code><section></code>` should be wrapped as inline.

Entrée utilisateur

Utilisez la `<kbd>` pour indiquer l' entrée qui est généralement entré via le clavier.

EXAMPLE

#	Prénom	nom de famille	Nom d'utilisateur
Pour modifier les paramètres, appuyez sur ctrl + ,			

To switch directories, type **<kbd>cd</kbd>** followed by the name of the directory.
To edit settings, press **<kbd><ctrl>+<kb></kb>**

bloc de base

Utilisez **<pre>** pour plusieurs lignes de code. Assurez - vous d'échapper à tous les équerres dans le code pour un rendu correct.

EXAMPLE

```
<p> Exemple de texte ici ... </p>
```

```
<pre>&lt;p&gt;Sample text here...&lt;/p&gt;</pre>
```

Vous pouvez éventuellement ajouter la **.pre-scrollable** classe, qui définira une hauteur maximum de 350px et de fournir une barre de défilement de l' axe y.

Variables

Pour indiquer les variables utilisent la **<var>** balise.

EXAMPLE

$$y = m x + b$$

```
<var>y</var> = <var>m</var><var>x</var> + <var>b</var>
```

Exemple de sortie

Pour indiquer des blocs exemple de sortie d'un programme utiliser la **<samp>** balise.

EXAMPLE

Ce texte est destiné à être traité en tant que sortie de l'échantillon à partir d'un programme d'ordinateur.

```
<samp>This text is meant to be treated as sample output from a computer program.</samp>
```

les tables

exemple de base

Pour base rembourrage style léger et seulement horizontales diviseurs-ajouter la classe de base **.table** à tout **<table>** . Il peut sembler super - redondante, mais compte tenu de l'utilisation généralisée des tables pour les autres plugins comme des calendriers et des sélecteurs de date, nous avons opté pour isoler nos styles de tableau personnalisé.

EXAMPLE

Table en option légende.

#	Prénom	nom de famille	Nom d'utilisateur
Table en option légende.			

#	Prénom	nom de famille	Nom d'utilisateur
1	marque	Otto	@mdo
2	Jacob	Thornton	@graisse
3	Larry	l'oiseau	@gazouillement

```
<table class="table">
...
</table>
```

lignes rayées

Utilisez `.table-striped` pour ajouter zebra-striping à une ligne du tableau dans le `<tbody>`.

Compatibilité inter-navigateur

Tables rayées sont de style via le `:nth-child` sélecteur CSS, qui ne sont pas disponibles dans Internet Explorer 8.

EXAMPLE

#	Prénom	nom de famille	Nom d'utilisateur
1	marque	Otto	@mdo
2	Jacob	Thornton	@graisse
3	Larry	l'oiseau	@gazouillement

```
<table class="table table-striped">
...
</table>
```

Table bordé

Ajouter `.table-bordered` des frontières sur tous les côtés de la table et les cellules.

EXAMPLE

#	Prénom	nom de famille	Nom d'utilisateur
1	marque	Otto	@mdo
2	Jacob	Thornton	@graisse
3	Larry	l'oiseau	@gazouillement

```
<table class="table table-bordered">
...
</table>
```

lignes Hover

Ajouter `.table-hover` à permettre un état stationnaire sur les lignes de table dans un `<tbody>`.

EXAMPLE

#	tête de colonne	tête de colonne	tête de colonne
1	marque	Otto	@mdo
2	Jacob	Thornton	@graisse
3	Larry	l'oiseau	@gazouillement

```
<table class="table table-hover">
...
</table>
```

Table Condensed

Ajouter `.table-condensed` à faire des tableaux plus compact en coupant le remplissage des cellules dans la moitié.

EXAMPLE

#	Prénom	nom de famille	Nom d'utilisateur
1	marque	Otto	@mdo
2	Jacob	Thornton	@graisse
3	Larry Bird		@gazouillement

```
<table class="table table-condensed">
...
</table>
```

les classes contextuelles

Utilisez les classes contextuelles pour colorer les lignes de table ou des cellules individuelles.

Classe	La description
.active	Applique la couleur de vol stationnaire à une ligne ou une cellule particulière
.success	Indique une action réussie ou positive
.info	Indique un changement ou une action d'information neutre
.warning	Indique un avertissement qui pourrait avoir besoin d'attention
.danger	Indique une action dangereuse ou potentiellement négative

EXAMPLE

#	tête de colonne	tête de colonne	tête de colonne
1	contenu de la colonne	contenu de la colonne	contenu de la colonne
2	contenu de la colonne	contenu de la colonne	contenu de la colonne
3	contenu de la colonne	contenu de la colonne	contenu de la colonne
4	contenu de la colonne	contenu de la colonne	contenu de la colonne
5	contenu de la colonne	contenu de la colonne	contenu de la colonne
6	contenu de la colonne	contenu de la colonne	contenu de la colonne

#	Table rubrique	Table rubrique	Table rubrique	Table rubrique	Table rubrique	Table rubrique
7	contenu de la colonne		contenu de la colonne		contenu de la colonne	
8	contenu de la colonne		contenu de la colonne		contenu de la colonne	
9	contenu de la colonne		contenu de la colonne		contenu de la colonne	

```
<!-- On rows -->
<tr class="active">....</tr>
<tr class="success">....</tr>
<tr class="warning">....</tr>
<tr class="danger">....</tr>
<tr class="info">....</tr>

<!-- On cells (`td` or `th`) -->
<tr>
  <td class="active">....</td>
  <td class="success">....</td>
  <td class="warning">....</td>
  <td class="danger">....</td>
  <td class="info">....</td>
</tr>
```

Convoyage sens aux technologies d'assistance

Utilisation de la couleur pour ajouter un sens à une ligne de table ou d'une cellule individuelle ne fournit une indication visuelle, qui ne seront pas transmis aux utilisateurs de technologies d' assistance - tels que les lecteurs d'écran. Veiller à ce que l' information désignée par la couleur est soit évidente du contenu lui - même (le texte visible dans le tableau pertinent rangée / cellule), ou est inclus par d'autres moyens, tels que le texte supplémentaire caché à la `.sr-only` classe.

tables Responsive

Créer des tableaux sensibles en enveloppant tout `.table` en `.table-responsive` pour les faire défiler horizontalement sur les petits appareils (sous 768px). Lors de la visualisation sur quoi que ce soit plus grand que 768px de large, vous ne verrez aucune différence dans ces tableaux.

écrêtage vertical / troncature

Tables Responsive utilisent `overflow-y: hidden`, qui clips hors tout contenu qui va au - delà du fond ou des bords supérieurs de la table. En particulier, cela peut couper outre des menus déroulants et autres widgets tiers.

Firefox et fieldsets

Firefox a un certain style de fieldset maladroit impliquant `width` qui interfère avec la table réactive. Cela ne peut pas être annulé sans un hack Firefox spécifique que nous ne fournissons dans Bootstrap:

```
@-moz-document url-prefix() {
  fieldset { display: table-cell; }
}
```

Pour plus d' informations, lisez cette réponse Stack Overflow .

EXAMPLE

| # | Table rubrique |
|---|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| 1 | Tableau cellulaire |

| # | Table rubrique |
|---|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| 2 | Tableau cellulaire |
| 3 | Tableau cellulaire |

| # | Table rubrique |
|---|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| 1 | Tableau cellulaire |
| 2 | Tableau cellulaire |
| 3 | Tableau cellulaire |

```
<div class="table-responsive">
  <table class="table">
    ...
  </table>
</div>
```

Formes

exemple de base

Individuels contrôles de formulaire reçoivent automatiquement un peu de style global. Tous les textes `<input>`, `<textarea>` et les `<select>` éléments avec `.form-control` sont mis à `width: 100%`; par défaut. Enveloppez les étiquettes et les contrôles `.form-group` pour l'espacement optimal.

EXAMPLE

Adresse e-mail

Mot de passe

entrée du fichier

 Aucun fichier choisi

Exemple de niveau bloc texte d'aide ici.

Check me out

```
<form>
  <div class="form-group">
    <label for="exampleInputEmail1">Email address</label>
    <input type="email" class="form-control" id="exampleInputEmail1" placeholder="Email">
  </div>
  <div class="form-group">
    <label for="exampleInputPassword1">Password</label>
    <input type="password" class="form-control" id="exampleInputPassword1" placeholder="Password">
  </div>
  <div class="form-group">
    <label for="exampleInputFile">File input</label>
    <input type="file" id="exampleInputFile">
    <p class="help-block">Example block-level help text here.</p>
  </div>
  <div class="checkbox">
    <label>
      <input type="checkbox"> Check me out
    </label>
  </div>
  <button type="submit" class="btn btn-default">Submit</button>
</form>
```

Ne pas mélanger de former des groupes avec des groupes d'entrée

Ne pas mélanger les groupes de formulaires directement avec les groupes d'entrée . Au lieu de cela, le groupe nid d'entrée à l' intérieur du groupe de forme.

forme Inline

Ajouter `.form-inline` à votre formulaire (qui ne doit pas être un `<form>`) pour les contrôles laissés-alignés et inline-block. **Ceci ne concerne que les formes dans les fenêtres qui sont au moins 768px de large.**

Peut nécessiter des largeurs personnalisées

Entrées et sélectionne ont `width: 100%`; appliquée par défaut dans Bootstrap. Dans formulaires en ligne, nous réinitialisons que de `width: auto;` sorte que plusieurs contrôles peuvent résider sur la même ligne. En fonction de votre mise en page, des largeurs de mesure supplémentaires peuvent être nécessaires.

Toujours ajouter des étiquettes

Les lecteurs d'écran auront des problèmes avec vos formulaires si vous ne pas inclure une étiquette pour chaque entrée. Pour ces formulaires en ligne, vous pouvez masquer les étiquettes en utilisant la `.sr-only` classe. Il existe d'autres méthodes alternatives de fournir une étiquette pour les technologies d' assistance, tels que le `aria-label`, `aria-labelledby` ou `title` attribut. Si aucun de ceux - ci est présente, les lecteurs d'écran peuvent avoir recours à l' utilisation de l' `placeholder` attribut, si elle est présente, mais il faut noter que l' utilisation de `placeholder` comme un remplacement pour d' autres méthodes d'étiquetage est déconseillé.

EXAMPLE

prénom

Jane Doe

Email

jane.doe@example.com

Envoyer une invitation

```
<form class="form-inline">
  <div class="form-group">
    <label for="exampleInputName2">Name</label>
    <input type="text" class="form-control" id="exampleInputName2" placeholder="Jane Doe">
  </div>
  <div class="form-group">
    <label for="exampleInputEmail2">Email</label>
    <input type="email" class="form-control" id="exampleInputEmail2" placeholder="jane.doe@example.com">
  </div>
  <button type="submit" class="btn btn-default">Send invitation</button>
</form>
```

EXAMPLE

Email

Password

 Souviens-toi de moi

se connecter

```
<form class="form-inline">
  <div class="form-group">
    <label class="sr-only" for="exampleInputEmail3">Email address</label>
    <input type="email" class="form-control" id="exampleInputEmail3" placeholder="Email">
  </div>
  <div class="form-group">
    <label class="sr-only" for="exampleInputPassword3">Password</label>
    <input type="password" class="form-control" id="exampleInputPassword3" placeholder="Password">
  </div>
  <div class="checkbox">
    <label>
      <input type="checkbox" /> Remember me
    </label>
  </div>
  <button type="submit" class="btn btn-default">Sign in</button>
</form>
```

EXAMPLE

\$ Amount .00

trésorerie de transfert

```
<form class="form-inline">
  <div class="form-group">
    <label class="sr-only" for="exampleInputAmount">Amount (in dollars)</label>
    <div class="input-group">
      <div class="input-group-addon">$</div>
      <input type="text" class="form-control" id="exampleInputAmount" placeholder="Amount">
      <div class="input-group-addon">.00</div>
    </div>
  </div>
  <button type="submit" class="btn btn-primary">Transfer cash</button>
</form>
```

forme horizontale

Utilisez les classes de grille prédéfinies de bootstrap pour aligner les étiquettes et les groupes de contrôles de formulaire dans une disposition horizontale en ajoutant `.form-horizontal` à la forme (qui n'a pas à être un `<form>`). Ce faisant les changements `.form-group` de se comporter comme des lignes de la grille, donc pas besoin de `.row`.

EXAMPLE

Email

Email

Mot de passe

Password

Souviens-toi de moi

se connecter

```
<form class="form-horizontal">
  <div class="form-group">
    <label for="inputEmail3" class="col-sm-2 control-label">Email</label>
    <div class="col-sm-10">
      <input type="email" class="form-control" id="inputEmail3" placeholder="Email">
    </div>
  </div>
  <div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-label">Password</label>
    <div class="col-sm-10">
      <input type="password" class="form-control" id="inputPassword3" placeholder="Password">
    </div>
  </div>
  <div class="form-group">
    <div class="col-sm-offset-2 col-sm-10">
      <div class="checkbox">
        <label>
          <input type="checkbox" /> Remember me
        </label>
      </div>
    </div>
  </div>
  <div class="form-group">
    <div class="col-sm-offset-2 col-sm-10">
      <button type="submit" class="btn btn-default">Sign in</button>
    </div>
  </div>
</form>
```

contrôles pris en charge

Des exemples de contrôles de formulaire standards pris en charge dans une mise en page par exemple de formulaire.

Entrées

La plupart de contrôle de forme commune, les champs de saisie de texte. Inclut le support pour tous les types HTML5: `text` , `password` , `datetime` , `datetime-local` , `date` , `month` , `time` , `week` , `number` , `email` , `url` , `search` , `tel` , et `color` .

Type de déclaration obligatoire

Les entrées ne seront pleinement décorées si leur `type` est correctement déclarée.

EXAMPLE

```
<input type="text" class="form-control" placeholder="Text input">
```

Groupes d'entrée

Pour ajouter du texte ou des boutons intégrés avant et / ou après tout basé sur le texte `<input>`, consultez le composant de groupe d'entrée .

Textarea

Contrôle de formulaire qui prend en charge plusieurs lignes de texte. Changement d' `rows` attribut si nécessaire.

EXAMPLE

```
Textarea
```

```
<textarea class="form-control" rows="3"></textarea>
```

Checkboxes et radios

Les cases à cocher permettent de sélectionner une ou plusieurs options dans une liste, tandis que les radios sont pour sélectionner une option parmi plusieurs.

Cases à cocher et les radios handicapées sont pris en charge, mais de fournir un curseur "non autorisé" sur le vol stationnaire du parent `<label>`, vous aurez besoin d'ajouter la `.disabled` classe à la société mère `.radio`, `.radio-inline`, `.checkbox`, ou `.checkbox-inline`.

Par défaut (empilés)

EXAMPLE

- La première option est ceci et cela, assurez-vous d'inclure pourquoi il est grand
- Option deux est désactivé

- La première option est ceci et cela, assurez-vous d'inclure pourquoi il est grand
- Option deux peut être autre chose et en le sélectionnant désélectionner l'option un
- La troisième option est désactivée

```

<div class="checkbox">
  <label>
    <input type="checkbox" value="" checked>
    Option one is this and that—be sure to include why it's great
  </label>
</div>
<div class="checkbox disabled">
  <label>
    <input type="checkbox" value="" disabled>
    Option two is disabled
  </label>
</div>

<div class="radio">
  <label>
    <input type="radio" name="optionsRadios" id="optionsRadios1" value="option1" checked>
    Option one is this and that—be sure to include why it's great
  </label>
</div>
<div class="radio">
  <label>
    <input type="radio" name="optionsRadios" id="optionsRadios2" value="option2">
    Option two can be something else and selecting it will deselect option one
  </label>
</div>
<div class="radio disabled">
  <label>
    <input type="radio" name="optionsRadios" id="optionsRadios3" value="option3" disabled>
    Option three is disabled
  </label>
</div>

```

checkboxes et radios Inline

Utilisez la `.checkbox-inline` ou des `.radio-inline` classes sur une série de cases à cocher ou des radios pour les contrôles qui apparaissent sur la même ligne.

EXAMPLE

1 2 3

1 2 3

```

<label class="checkbox-inline">
  <input type="checkbox" id="inlineCheckbox1" value="option1"> 1
</label>
<label class="checkbox-inline">
  <input type="checkbox" id="inlineCheckbox2" value="option2"> 2
</label>
<label class="checkbox-inline">
  <input type="checkbox" id="inlineCheckbox3" value="option3"> 3
</label>

<label class="radio-inline">
  <input type="radio" name="inlineRadioOptions" id="inlineRadio1" value="option1"> 1
</label>
<label class="radio-inline">
  <input type="radio" name="inlineRadioOptions" id="inlineRadio2" value="option2"> 2
</label>
<label class="radio-inline">
  <input type="radio" name="inlineRadioOptions" id="inlineRadio3" value="option3"> 3
</label>

```

Checkboxes et radios sans texte de l'étiquette

Si vous avez pas de texte dans le `<label>`, l'entrée est positionné comme on pouvait s'y attendre. **Fonctionne actuellement que sur les cases et les radios non-inline.** Rappelez - vous de toujours fournir une certaine forme de label pour les technologies d' assistance (par exemple, en utilisant `aria-label`).

EXAMPLE

```
<div class="checkbox">
  <label>
    <input type="checkbox" id="blankCheckbox" value="option1" aria-label="...">
  </label>
</div>
<div class="radio">
  <label>
    <input type="radio" name="blankRadio" id="blankRadio1" value="option1" aria-label="...">
  </label>
</div>
```

Sélectionne

Notez que de nombreux menus de sélection , à savoir indigènes dans Safari et Chrome coins arrondis ont-qui ne peuvent être modifiés via les border-radius propriétés.

EXAMPLE

1



```
<select class="form-control">
  <option>1</option>
  <option>2</option>
  <option>3</option>
  <option>4</option>
  <option>5</option>
</select>
```

Pour les `<select>` commandes avec l' `multiple` attribut, plusieurs options sont affichées par défaut.

EXAMPLE1
2
3
4
5

```
<select multiple class="form-control">
  <option>1</option>
  <option>2</option>
  <option>3</option>
  <option>4</option>
  <option>5</option>
</select>
```

contrôle statique

Lorsque vous avez besoin de placer le texte brut à côté d'une étiquette de formulaire dans un formulaire, utilisez la `.form-control-static` classe sur un `<p>` .

EXAMPLE**Email**

email@example.com

Mot de passe

Password

```
<form class="form-horizontal">
  <div class="form-group">
    <label class="col-sm-2 control-label">Email</label>
    <div class="col-sm-10">
      <p class="form-control-static">email@example.com</p>
    </div>
  </div>
  <div class="form-group">
    <label for="inputPassword" class="col-sm-2 control-label">Password</label>
    <div class="col-sm-10">
      <input type="password" class="form-control" id="inputPassword" placeholder="Password">
    </div>
  </div>
</form>
```

EXAMPLE

email@example.com

Password

Confirmer l'identité

```
<form class="form-inline">
  <div class="form-group">
    <label class="sr-only">Email</label>
    <p class="form-control-static">email@example.com</p>
  </div>
  <div class="form-group">
    <label for="inputPassword2" class="sr-only">Password</label>
    <input type="password" class="form-control" id="inputPassword2" placeholder="Password">
  </div>
  <button type="submit" class="btn btn-default">Confirm identity</button>
</form>
```

état Mise au point

Nous enlevons les défaut outline styles sur certains contrôles de formulaire et d' appliquer un box-shadow à sa place pour :focus .

EXAMPLE

Demonstrative focus state

Demo :focus Etat

L'exemple d' entrée ci - dessus utilise des styles personnalisés dans notre documentation pour démontrer l' :focus état sur un .form-control .

Etat désactivé

Ajouter l' disabled attribut booléen sur une entrée pour éviter les interactions de l' utilisateur. Entrées désactivées apparaissent plus léger et ajouter un not-allowed curseur.

EXAMPLE

Disabled input here...

```
<input class="form-control" id="disabledInput" type="text" placeholder="Disabled input here..." disabled>
```

fieldsets mobilité

Ajouter l' `disabled` attribut à un `<fieldset>` désactiver tous les contrôles dans le `<fieldset>` même temps.

Caveat sur la fonctionnalité de lien `<a>`

Par défaut, les navigateurs vont traiter tous les contrôles natifs de formulaire (`<input>`, `<select>` et `<button>` éléments) à l' intérieur d' un `<fieldset disabled>` comme handicapé, empêchant les deux interactions clavier et souris sur eux. Toutefois, si votre formulaire comprend également des `<a ... class="btn btn-*">` éléments, ceux - ci ne seront donnés un style de `pointer-events: none`. Comme indiqué dans la section sur l' état désactivé pour les boutons (et plus précisément dans la sous-section pour les éléments d'ancrage), cette propriété CSS est pas encore normalisé et ne sont pas entièrement pris en charge dans Opera 18 et ci - dessous, ou dans Internet Explorer 11, et a gagné 't empêcher les utilisateurs de clavier d'être en mesure de se concentrer ou d' activer ces liens. Donc , pour être sûr, utiliser JavaScript personnalisé pour désactiver ces liens.

Compatibilité inter-navigateur

Alors que Bootstrap appliquera ces styles dans tous les navigateurs, Internet Explorer 11 et ci - dessous ne prennent pas complètement l' `disabled` attribut sur un `<fieldset>`. Utilisez JavaScript personnalisé pour désactiver le fieldset dans ces navigateurs.

EXAMPLE

entrée handicapés

Disabled input

menu, sélectionnez Désactivé

Disabled sélectionnez

Vous ne pouvez pas vérifier

Soumettre

```
<form>
  <fieldset disabled>
    <div class="form-group">
      <label for="disabledTextInput">Disabled input</label>
      <input type="text" id="disabledTextInput" class="form-control" placeholder="Disabled input">
    </div>
    <div class="form-group">
      <label for="disabledSelect">Disabled select menu</label>
      <select id="disabledSelect" class="form-control">
        <option>Disabled select</option>
      </select>
    </div>
    <div class="checkbox">
      <label>
        <input type="checkbox"> Can't check this
      </label>
    </div>
    <button type="submit" class="btn btn-primary">Submit</button>
  </fieldset>
</form>
```

état Readonly

Ajouter l' `readonly` attribut booléen sur une entrée pour empêcher la modification de la valeur de l'entrée. Lecture seule entrées apparaissent plus léger (tout comme les entrées invalides), mais conservent le curseur standard.

EXAMPLE

```
Readonly input here...
```

```
<input class="form-control" type="text" placeholder="Readonly input here..." readonly>
```

Texte d'aide

niveau bloc texte d'aide pour les contrôles de formulaire.

Associant texte d'aide avec les contrôles de formulaire

Le texte d'aide doit être explicitement associé au contrôle de la forme qu'elle se rapporte à l'utilisation de l' `aria-describedby` attribut. Cela garantira que les technologies d'assistance - tels que les lecteurs d'écran - annonceront cette aide du texte lorsque l'utilisateur se concentre ou dans le contrôle.

EXAMPLE

Entrée avec le texte d'aide

```
Un bloc de texte d'aide qui se brise sur une nouvelle ligne et peut se prolonger au-delà d'une ligne.
```

```
<label class="sr-only" for="inputHelpBlock">Input with help text</label>
<input type="text" id="inputHelpBlock" class="form-control" aria-describedby="helpBlock">
...
<span id="helpBlock" class="help-block">A block of help text that breaks onto a new line and may extend beyond one line.
</span>
```

états de validation

Bootstrap comprend des styles de validation pour l'erreur, d'avertissement et les états de succès sur les contrôles de formulaire. Pour l'utiliser, ajouter `.has-warning`, `.has-error` ou `.has-success` à l'élément parent. Tout `.control-label`, `.form-control` et au `.help-block` sein de cet élément recevra les styles de validation.

Convoyage état de validation de technologies d'assistance et les utilisateurs daltoniens

L'utilisation de ces styles de validation pour indiquer l'état d'un contrôle de formulaire fournit seulement une indication sur la base de couleur visuelle, qui ne seront pas transmis aux utilisateurs de technologies d'assistance - tels que les lecteurs d'écran - ou daltonien utilisateurs.

Assurez-vous que une alternative indication de l'état est également fourni. Par exemple, vous pouvez inclure une indication sur l'état dans le contrôle de la forme `<label>` du texte lui-même (comme cela est le cas dans l'exemple de code suivant), inclure un Glyphicon (avec texte alternatif approprié à l'aide de la `.sr-only` classe - voir les exemples de Glyphicon), ou en fournissant un supplémentaire le texte d'aide bloc. Spécifiquement pour les technologies d'assistance, invalides contrôles de formulaire peuvent également être affectés d'un `aria-invalid="true"` attribut.

EXAMPLE

Entrée avec succès

Un bloc de texte d'aide qui se brise sur une nouvelle ligne et peut se prolonger au-delà d'une ligne.

Entrée avec avertissement

Entrée avec l'erreur

- Checkbox avec succès
- Checkbox avec avertissement
- Checkbox avec l'erreur

```
<div class="form-group has-success">
  <label class="control-label" for="inputSuccess1">Input with success</label>
  <input type="text" class="form-control" id="inputSuccess1" aria-describedby="helpBlock2">
  <span id="helpBlock2" class="help-block">A block of help text that breaks onto a new line and may extend beyond one line.</span>
</div>
<div class="form-group has-warning">
  <label class="control-label" for="inputWarning1">Input with warning</label>
  <input type="text" class="form-control" id="inputWarning1">
</div>
<div class="form-group has-error">
  <label class="control-label" for="inputError1">Input with error</label>
  <input type="text" class="form-control" id="inputError1">
</div>
<div class="has-success">
  <div class="checkbox">
    <label>
      <input type="checkbox" id="checkboxSuccess" value="option1">
      Checkbox with success
    </label>
  </div>
</div>
<div class="has-warning">
  <div class="checkbox">
    <label>
      <input type="checkbox" id="checkboxWarning" value="option1">
      Checkbox with warning
    </label>
  </div>
</div>
<div class="has-error">
  <div class="checkbox">
    <label>
      <input type="checkbox" id="checkboxError" value="option1">
      Checkbox with error
    </label>
  </div>
</div>
```

Avec des icônes en option

Vous pouvez également ajouter des icônes de rétroaction en option avec l'ajout de `.has-feedback` et l'icône de droite.

Commentaires icônes fonctionnent uniquement avec textuels `<input class="form-control">` éléments.

Icônes, étiquettes, et des groupes d'entrée

Positionnement manuel des icônes de rétroaction est nécessaire pour les entrées sans étiquette et pour les groupes d'entrée avec un add-on à droite. Vous êtes fortement encouragés à fournir des étiquettes pour toutes les entrées pour des raisons d'accessibilité. Si vous souhaitez éviter que les étiquettes soient affichées, les cacher à la `.sr-only` classe. Si vous devez faire sans étiquettes, ajuster la

top valeur de l'icône de rétroaction. Pour les groupes d'entrée, ajuster la right valeur à une valeur de pixel appropriée en fonction de la largeur de votre addon.

Convoyage le sens de l'icône pour les technologies d'assistance

Pour veiller à ce que les technologies d' assistance - tels que les lecteurs d'écran - transmettre correctement la signification d'une icône, texte complémentaire caché devrait être inclus dans la .sr-only classe et explicitement associée au contrôle de la forme , il se rapporte à l' utilisation aria-describedby . Sinon, assurez-vous que le sens (par exemple, le fait qu'il y ait un avertissement pour un champ de saisie de texte particulier) est transporté sous une autre forme, par exemple modifier le texte de la réelle <label> associée à la commande de formulaire.

Bien que les exemples suivants mentionnent déjà l'état de leurs contrôles de formulaire respectifs de validation dans le <label> texte lui-même, la technique ci-dessus (à l'aide du .sr-only texte et aria-describedby) a été inclus à des fins d' illustration.

EXAMPLE

Entrée avec succès



Entrée avec avertissement



Entrée avec l'erreur



Groupe d'entrée avec succès



```
<div class="form-group has-success has-feedback">
  <label class="control-label" for="inputSuccess2">Input with success</label>
  <input type="text" class="form-control" id="inputSuccess2" aria-describedby="inputSuccess2Status">
  <span class="glyphicon glyphicon-ok form-control-feedback" aria-hidden="true"></span>
  <span id="inputSuccess2Status" class="sr-only">(success)</span>
</div>
<div class="form-group has-warning has-feedback">
  <label class="control-label" for="inputWarning2">Input with warning</label>
  <input type="text" class="form-control" id="inputWarning2" aria-describedby="inputWarning2Status">
  <span class="glyphicon glyphicon-warning-sign form-control-feedback" aria-hidden="true"></span>
  <span id="inputWarning2Status" class="sr-only">(warning)</span>
</div>
<div class="form-group has-error has-feedback">
  <label class="control-label" for="inputError2">Input with error</label>
  <input type="text" class="form-control" id="inputError2" aria-describedby="inputError2Status">
  <span class="glyphicon glyphicon-remove form-control-feedback" aria-hidden="true"></span>
  <span id="inputError2Status" class="sr-only">(error)</span>
</div>
<div class="form-group has-success has-feedback">
  <label class="control-label" for="inputGroupSuccess1">Input group with success</label>
  <div class="input-group">
    <span class="input-group-addon">@</span>
    <input type="text" class="form-control" id="inputGroupSuccess1" aria-describedby="inputGroupSuccess1Status">
  </div>
  <span class="glyphicon glyphicon-ok form-control-feedback" aria-hidden="true"></span>
  <span id="inputGroupSuccess1Status" class="sr-only">(success)</span>
</div>
```

icônes en option dans les formes horizontales et inline

EXAMPLE

Entrée avec succès**Groupe d'entrée avec succès**

@	
---	--

```
<form class="form-horizontal">
  <div class="form-group has-success has-feedback">
    <label class="control-label col-sm-3" for="inputSuccess3">Input with success</label>
    <div class="col-sm-9">
      <input type="text" class="form-control" id="inputSuccess3" aria-describedby="inputSuccess3Status">
      <span class="glyphicon glyphicon-ok form-control-feedback" aria-hidden="true"></span>
      <span id="inputSuccess3Status" class="sr-only">(success)</span>
    </div>
  </div>
  <div class="form-group has-success has-feedback">
    <label class="control-label col-sm-3" for="inputGroupSuccess2">Input group with success</label>
    <div class="col-sm-9">
      <div class="input-group">
        <span class="input-group-addon">@</span>
        <input type="text" class="form-control" id="inputGroupSuccess2" aria-describedby="inputGroupSuccess2Status">
      </div>
      <span class="glyphicon glyphicon-ok form-control-feedback" aria-hidden="true"></span>
      <span id="inputGroupSuccess2Status" class="sr-only">(success)</span>
    </div>
  </div>
</form>
```

EXAMPLE**Entrée avec succès****Groupe d'entrée avec succès**

@	
---	--

```
<form class="form-inline">
  <div class="form-group has-success has-feedback">
    <label class="control-label" for="inputSuccess4">Input with success</label>
    <input type="text" class="form-control" id="inputSuccess4" aria-describedby="inputSuccess4Status">
    <span class="glyphicon glyphicon-ok form-control-feedback" aria-hidden="true"></span>
    <span id="inputSuccess4Status" class="sr-only">(success)</span>
  </div>
</form>
<form class="form-inline">
  <div class="form-group has-success has-feedback">
    <label class="control-label" for="inputGroupSuccess3">Input group with success</label>
    <div class="input-group">
      <span class="input-group-addon">@</span>
      <input type="text" class="form-control" id="inputGroupSuccess3" aria-describedby="inputGroupSuccess3Status">
    </div>
    <span class="glyphicon glyphicon-ok form-control-feedback" aria-hidden="true"></span>
    <span id="inputGroupSuccess3Status" class="sr-only">(success)</span>
  </div>
</form>
```

Icônes en option avec cachés .sr-only étiquettes

Si vous utilisez la `.sr-only` classe pour cacher son un contrôle de formulaire `<label>` (plutôt que d'utiliser d'autres options d'étiquetage, telles que l' `aria-label` attribut), Bootstrap ajustera automatiquement la position de l'icône une fois qu'il a été ajouté.

EXAMPLE

✓

✓

```
<div class="form-group has-success has-feedback">
  <label class="control-label sr-only" for="inputSuccess5">Hidden label</label>
  <input type="text" class="form-control" id="inputSuccess5" aria-describedby="inputSuccess5Status">
  <span class="glyphicon glyphicon-ok form-control-feedback" aria-hidden="true"></span>
  <span id="inputSuccess5Status" class="sr-only">(success)</span>
</div>
<div class="form-group has-success has-feedback">
  <label class="control-label sr-only" for="inputGroupSuccess4">Input group with success</label>
  <div class="input-group">
    <span class="input-group-addon">@</span>
    <input type="text" class="form-control" id="inputGroupSuccess4" aria-describedby="inputGroupSuccess4Status">
  </div>
  <span class="glyphicon glyphicon-ok form-control-feedback" aria-hidden="true"></span>
  <span id="inputGroupSuccess4Status" class="sr-only">(success)</span>
</div>
```

dimensionnement de commande

Définir des hauteurs en utilisant des classes comme `.input-lg`, et définir la largeur en utilisant des classes de colonnes de la grille comme `.col-lg-*`.

Hauteur dimensionnement

Créer plus grand ou plus des contrôles de formulaire qui correspondent à la taille des boutons.

EXAMPLE

▼

▼

▼

```
<input class="form-control input-lg" type="text" placeholder=".input-lg">
<input class="form-control" type="text" placeholder="Default input">
<input class="form-control input-sm" type="text" placeholder=".input-sm">

<select class="form-control input-lg">...</select>
<select class="form-control">...</select>
<select class="form-control input-sm">...</select>
```

Horizontal tailles de groupe de forme

Rapidement des étiquettes de taille et les contrôles de formulaire au sein `.form-horizontal` en ajoutant `.form-group-lg` ou `.form-group-sm`.

EXAMPLE

Grande étiquette

Large input

Petite étiquette

Small input

```
<form class="form-horizontal">
  <div class="form-group form-group-lg">
    <label class="col-sm-2 control-label" for="formGroupInputLarge">Large label</label>
    <div class="col-sm-10">
      <input class="form-control" type="text" id="formGroupInputLarge" placeholder="Large input">
    </div>
  </div>
  <div class="form-group form-group-sm">
    <label class="col-sm-2 control-label" for="formGroupInputSmall">Small label</label>
    <div class="col-sm-10">
      <input class="form-control" type="text" id="formGroupInputSmall" placeholder="Small input">
    </div>
  </div>
</form>
```

Colonne dimensionnement

Enveloppez entrées dans les colonnes de la grille, ou tout élément parent personnalisé, d'appliquer facilement des largeurs souhaitées.

EXAMPLE

.col-xs-2

.col-xs-3

.col-xs-4

```
<div class="row">
  <div class="col-xs-2">
    <input type="text" class="form-control" placeholder=".col-xs-2">
  </div>
  <div class="col-xs-3">
    <input type="text" class="form-control" placeholder=".col-xs-3">
  </div>
  <div class="col-xs-4">
    <input type="text" class="form-control" placeholder=".col-xs-4">
  </div>
</div>
```

Boutons

balises Button

Utilisez les classes de boutons sur un `<a>`, `<button>` ou `<input>` élément.

EXAMPLE

Lien

Bouton

entrée

Soumettre

```
<a class="btn btn-default" href="#" role="button">Link</a>
<button class="btn btn-default" type="submit">Button</button>
<input class="btn btn-default" type="button" value="Input">
<input class="btn btn-default" type="submit" value="Submit">
```

utilisation spécifiques Contexte

Alors que les classes de boutons peuvent être utilisés sur `<a>` et `<button>` éléments, seuls `<button>` éléments sont pris en charge au sein de nos nav et navbar composants.

Liens agissant comme des boutons

Si les `<a>` éléments sont utilisés pour agir en tant que boutons - la fonctionnalité de déclenchement en page, plutôt que de naviguer vers un autre document ou une section au sein de la page en cours - ils devraient également recevoir une appropriée `role="button"`.

Cross-browser rendu

En tant que meilleure pratique, **nous vous recommandons fortement d'utiliser l' `<button>` élément autant que possible** pour assurer correspondant cross-browser rendu.

Entre autres choses, il y a un bug dans Firefox <30 qui nous empêche de régler le `line-height` des `<input>` boutons à base, les obligeant à ne pas correspondre exactement à la hauteur des autres boutons de Firefox.

options de bouton

Utilisez l'une des classes de boutons disponibles pour créer rapidement un bouton de style.

EXAMPLE



```
<!-- Standard button -->
<button type="button" class="btn btn-default">Default</button>

<!-- Provides extra visual weight and identifies the primary action in a set of buttons -->
<button type="button" class="btn btn-primary">Primary</button>

<!-- Indicates a successful or positive action -->
<button type="button" class="btn btn-success">Success</button>

<!-- Contextual button for informational alert messages -->
<button type="button" class="btn btn-info">Info</button>

<!-- Indicates caution should be taken with this action -->
<button type="button" class="btn btn-warning">Warning</button>

<!-- Indicates a dangerous or potentially negative action -->
<button type="button" class="btn btn-danger">Danger</button>

<!-- Deemphasize a button by making it look like a link while maintaining button behavior -->
<button type="button" class="btn btn-link">Link</button>
```

Convoyage sens aux technologies d'assistance

Utilisation de la couleur pour ajouter un sens à un bouton seulement fournit une indication visuelle, qui ne seront pas transmis aux utilisateurs de technologies d' assistance - tels que les lecteurs d'écran. Veiller à ce que l' information notée par la couleur est soit évidente du contenu lui - même (le texte visible du bouton), ou est inclus par d'autres moyens, tels que le texte supplémentaire caché à la `.sr-only` classe.

Tailles

Fancy boutons plus ou moins? Ajouter `.btn-lg`, `.btn-sm` ou `.btn-xs` pour les tailles supplémentaires.

EXAMPLE

```
<p>
  <button type="button" class="btn btn-primary btn-lg">Large button</button>
  <button type="button" class="btn btn-default btn-lg">Large button</button>
</p>
<p>
  <button type="button" class="btn btn-primary">Default button</button>
  <button type="button" class="btn btn-default">Default button</button>
</p>
<p>
  <button type="button" class="btn btn-primary btn-sm">Small button</button>
  <button type="button" class="btn btn-default btn-sm">Small button</button>
</p>
<p>
  <button type="button" class="btn btn-primary btn-xs">Extra small button</button>
  <button type="button" class="btn btn-default btn-xs">Extra small button</button>
</p>
```

Créer des boutons-ces de niveau de bloc qui couvrent toute la largeur d'un parent- en ajoutant .btn-block .

EXAMPLE

```
<button type="button" class="btn btn-primary btn-lg btn-block">Block level button</button>
<button type="button" class="btn btn-default btn-lg btn-block">Block level button</button>
```

état actif

Les boutons apparaissent pressé (avec un fond plus sombre, plus sombre frontière, et encart ombre) quand elle est active. Pour les <button> éléments, cela se fait par l' intermédiaire :active . Pour les <a> éléments, il est fait avec .active . Cependant, vous pouvez utiliser .active le <button> s (et inclure l' aria-pressed="true" attribut) si vous avez besoin de reproduire l'état actif par programme.

élément Button

Pas besoin d'ajouter :active que c'est une pseudo-classe, mais si vous avez besoin de forcer la même apparence, allez - y et ajoutez .active .

EXAMPLE

```
<button type="button" class="btn btn-primary btn-lg active">Primary button</button>
<button type="button" class="btn btn-default btn-lg active">Button</button>
```

élément d'ancrage

Ajouter la `.active` classe à `<a>` boutons.

EXAMPLE

[Lien primaire](#) [Lien](#)

```
<a href="#" class="btn btn-primary btn-lg active" role="button">Primary link</a>
<a href="#" class="btn btn-default btn-lg active" role="button">Link</a>
```

Etat désactivé

Faites boutons regardent unclickable en les rendant en arrière avec `opacity`.

élément Button

Ajouter l' `disabled` attribut à `<button>` boutons.

EXAMPLE

[bouton primaire](#) [Bouton](#)

```
<button type="button" class="btn btn-lg btn-primary" disabled="disabled">Primary button</button>
<button type="button" class="btn btn-default btn-lg" disabled="disabled">Button</button>
```

Compatibilité inter-navigateur

Si vous ajoutez l' `disabled` attribut à un `<button>`, Internet Explorer 9 et ci-dessous vous rendre le texte gris avec un `text-shadow` méchant que nous ne pouvons pas réparer.

élément d'ancre

Ajouter la `.disabled` classe à `<a>` boutons.

EXAMPLE

[Lien primaire](#) [Lien](#)

```
<a href="#" class="btn btn-primary btn-lg disabled" role="button">Primary link</a>
<a href="#" class="btn btn-default btn-lg disabled" role="button">Link</a>
```

Nous utilisons en `.disabled` tant que classe d'utilité ici, similaire à la commune `.active` classe, donc pas de préfixe est nécessaire.

Lien fonctionnalité mise en garde

Cette classe utilise `pointer-events: none` pour essayer de désactiver la fonctionnalité de liaison de `<a>`s, mais que la propriété CSS n'a pas encore été standardisée et n'est pas entièrement pris en charge dans Opera 18 et ci-dessous, ou dans Internet Explorer 11. En outre, même dans les navigateurs qui ne supportent `pointer-events: none`, clavier la navigation reste inchangée, ce qui signifie que les utilisateurs et les utilisateurs des technologies d'assistance clavier voyants seront toujours en mesure d'activer ces liens. Donc, pour être sûr, utiliser JavaScript personnalisé pour désactiver ces liens.

Images

images Responsive

Images en Bootstrap 3 peuvent être faites en réponse conviviale via l'ajout de la `.img-responsive` classe. Cela vaut `max-width: 100%; height: auto;` et `display: block;` à l'image de sorte qu'elle adapte bien à l'élément parent.

Pour centrer les images qui utilisent la `.img-responsive` classe, utilisez au `.center-block` lieu de `.text-center`. Voir la section des classes d'aide pour plus de détails sur l'`.center-block` utilisation.

images SVG et IE 8-10

Dans Internet Explorer 8-10, les images SVG avec `.img-responsive` sont dimensionnés de manière disproportionnée. Pour résoudre ce problème, ajouter le `width: 100% \9;` cas échéant. Bootstrap ne concerne pas cela automatiquement car il provoque des complications à d'autres formats d'image.

```

```

formes d'image

Ajouter des classes à un `` élément images pour facilement le style dans tout projet.

Compatibilité inter-navigateur

Gardez à l'esprit que Internet Explorer 8 manque de soutien pour des coins arrondis.

EXAMPLE



```



```

Les classes utilitaires

couleurs contextuelles

Convey sens à travers la couleur avec une poignée de classes utilitaires d'accent. Ceux-ci peuvent également être appliquées aux liens et foncent sur le vol stationnaire, tout comme nos styles de lien par défaut.

EXAMPLE

Fusce dapibus, tellus ac Cursus commodo, tortor mauris nibh.

Nullam id dolor id nibh ultricies vehicula ut id elit.

mollis de Duis, HNE luctus non commodo, nisi erat porttitor Ligula.

Mécène sed diam eget risus varius blandit sit amet non magna.

Etiam Malesuada magna porta sem mollis euismod.

Donec ullamcorper nulla non metus auctor fringilla.

```
<p class="text-muted">...</p>
<p class="text-primary">...</p>
<p class="text-success">...</p>
<p class="text-info">...</p>
<p class="text-warning">...</p>
<p class="text-danger">...</p>
```

Faire face à la spécificité

Parfois, les classes d'accentuation ne peuvent pas être appliquées en raison de la spécificité d'un autre sélecteur. Dans la plupart des cas, une solution de contournement est suffisante pour envelopper votre texte dans un `` avec la classe.

Convoyage sens aux technologies d'assistance

Utilisation de la couleur pour ajouter un sens seulement fournit une indication visuelle, qui ne seront pas transmis aux utilisateurs de technologies d' assistance - tels que les lecteurs d'écran. Veiller à ce que l' information désignée par la couleur est soit évidente du contenu lui - même (les couleurs contextuelles ne sont utilisés que pour renforcer le sens qui est déjà présente dans le texte / balisage), ou est inclus par d'autres moyens, tels que le texte supplémentaire caché à la `.sr-only` classe .

origines contextuelles

Comme pour les classes contextuelles couleur du texte, facilement définir l'arrière-plan d'un élément à une classe contextuelle. composants d'ancre ont sur le vol stationnaire, tout comme les classes de texte.

EXAMPLE

Nullam id dolor id nibh ultricies vehicula ut id elit.

mollis de Duis, HNE luctus non commodo, nisi erat porttitor Ligula.

Mécène sed diam eget risus varius blandit sit amet non magna.

Etiam Malesuada magna porta sem mollis euismod.

Donec ullamcorper nulla non metus auctor fringilla.

```
<p class="bg-primary">...</p>
<p class="bg-success">...</p>
<p class="bg-info">...</p>
<p class="bg-warning">...</p>
<p class="bg-danger">...</p>
```

Faire face à la spécificité

Parfois, les classes de fond contextuelle ne peuvent pas être appliquées en raison de la spécificité d'un autre sélecteur. Dans certains cas, une solution de contournement est suffisante pour envelopper le contenu de votre élément dans un `<div>` avec la classe.

Convoyage sens aux technologies d'assistance

Comme avec les couleurs contextuelles , veiller à ce que toute signification transmise par la couleur est également transmis dans un format qui ne soit pas purement présentation.

icône de fermeture

Utilisez la fermeture icône générique pour rejeter le contenu comme modaux et des alertes.

EXAMPLE

×

```
<button type="button" class="close" aria-label="Close"><span aria-hidden="true">&times;</span></button>
```

CARETS

Utilisez carets pour indiquer la fonctionnalité et la direction déroulante. Notez que le caret par défaut inversera automatiquement dans les menus dropdown .

EXAMPLE

▼

```
<span class="caret"></span>
```

flotteurs rapides

Float un élément à gauche ou à droite avec une classe. !important Est inclus pour éviter les problèmes de spécificité. Les classes peuvent également être utilisés comme mixins.

```
<div class="pull-left">...</div>
<div class="pull-right">...</div>
```

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

// Usage as mixins
.element {
  .pull-left();
}
.another-element {
  .pull-right();
}
```

Ne pas utiliser dans navbars

Pour aligner les composants dans navbars avec des classes de services publics, l' utilisation .navbar-left ou au .navbar-right lieu. Voir les docs navbar pour plus de détails.

Centre des blocs de contenu

Définir un élément `display: block` et le centre via `margin` . Disponible en tant que mixin et de classe.

```
<div class="center-block">...</div>
```

```
// Class
.center-block {
  display: block;
  margin-left: auto;
  margin-right: auto;
}

// Usage as a mixin
.element {
  .center-block();
}
```

clearfix

Effacer facilement `float`s en ajoutant **de l'élément parent**. Utilise le micro clearfix comme popularisé par Nicolas Gallagher. Peut aussi être utilisé comme un mixin..`.clearfix`

```
<!-- Usage as a class -->
<div class="clearfix">...</div>
```

```
// Mixin itself
.clearfix() {
  &:before,
  &:after {
    content: " ";
    display: table;
  }
  &:after {
    clear: both;
  }
}

// Usage as a mixin
.element {
  .clearfix();
}
```

Affichage et masquage du contenu

Forcer un élément à afficher ou masquer (**y compris pour les lecteurs d'écran**) avec l'utilisation de `.show` et `.hidden` classes. Ces classes utilisent `!important` pour éviter les conflits de spécificité, tout comme les flotteurs rapides. Ils ne sont disponibles que pour le basculement de niveau bloc. Ils peuvent également être utilisés comme mixins.

`.hide` est disponible, mais il n'affecte pas toujours les lecteurs écran et est **dépréciée** à partir de v3.0.1. Utilisez `.hidden` ou à la `.sr-only` place.

En outre, `.invisible` peut être utilisé pour activer uniquement la visibilité d'un élément, ce qui signifie son `display` est pas modifiée et l'élément peut affecter le flux du document.

```
<div class="show">...</div>
<div class="hidden">...</div>
```

	Extra petits appareils portables (<768px)	Petits appareils Tablets (≥768px)	Appareils Moyen Desktops (≥992px)	Les grands appareils de bureau (≥1200px)
<pre>// Classes .show { display: block !important; } .hidden { display: none !important; } // Usage as mixins .element { .show(); } .another-element { .hidden(); }</pre>				

Lecteur d'écran et le clavier contenu de navigation

Cacher un élément à tous les appareils , à l' **exception des lecteurs d'écran** avec `.sr-only` . Combinez `.sr-only` avec `.sr-only-focusable` pour afficher l'élément à nouveau quand il est concentré (par exemple par un clavier seul utilisateur). Nécessaire pour suivre les meilleures pratiques en matière d'accessibilité . Peut aussi être utilisé comme mixins.

```
<a class="sr-only sr-only-focusable" href="#content">Skip to main content</a>
```

```
// Usage as a mixin
.skip-navigation {
  .sr-only();
  .sr-only-focusable();
}
```

le remplacement de l'image

Utiliser la `.text-hide` classe ou mixin pour aider à remplacer le contenu de texte d'un élément avec une image d'arrière - plan.

```
<h1 class="text-hide">Custom heading</h1>
```

```
// Usage as a mixin
.heading {
  .text-hide();
}
```

utilitaires Responsive

Pour le développement mobile-friendly plus rapide, utiliser ces classes utilitaires pour afficher et masquer le contenu par le dispositif via requête de médias. Sont également inclus les classes utilitaires pour basculer le contenu lors de l'impression.

Essayez d'utiliser ceux-ci sur une base limitée et éviter de créer entièrement différentes versions d'un même site. Au lieu de cela, les utiliser pour compléter la présentation de chaque appareil.

Classes disponibles

Utilisez un seul ou une combinaison des classes disponibles pour basculer contenu à travers des points d'arrêt de viewport.

Besoin de classes	Extra petits appareils portables (<768px)	Petits appareils CSS Tablets (≥768px)	Navigateurs Moyen Desktops (≥992px)	Les grands appareils de bureau (≥1200px)
.visible-xs-*	Extra petits appareils portables (<768px)	Petits appareils Tablets (≥768px)	Appareils Moyen Desktops (≥992px)	Les grands appareils de bureau (≥1200px)
.visible-sm-*	Caché	Visible	Caché	Caché
.visible-md-*	Caché	Caché	Visible	Caché
.visible-lg-*	Caché	Caché	Caché	Visible
.hidden-xs	Caché	Visible	Visible	Visible
.hidden-sm	Visible	Caché	Visible	Visible
.hidden-md	Visible	Visible	Caché	Visible
.hidden-lg	Visible	Visible	Visible	Caché

Au v3.2.0, les `.visible-*-*` classes pour chaque point d'arrêt sont disponibles en trois variantes, un pour chaque CSS `display` valeur de la propriété ci-dessous.

Groupe de classes	CSS display
<code>.visible-*block</code>	<code>display: block;</code>
<code>.visible-*inline</code>	<code>display: inline;</code>
<code>.visible-*inline-block</code>	<code>display: inline-block;</code>

Donc, pour les petits supplémentaires (`xs`) écrans, par exemple, les disponibles `.visible-*-*` classes sont: `.visible-xs-block`, `.visible-xs-inline`, et `.visible-xs-inline-block`.

Les classes `.visible-xs`, `.visible-sm`, `.visible-md` et `.visible-lg` existent aussi, mais sont **dépréciées à partir de v3.2.0**. Ils sont à peu près équivalents à l'`.visible-*block` exception des cas spéciaux supplémentaires pour basculer les `<table>` éléments concernant la PI.

cours d'impression

Comme pour les classes sensibles régulières, les utiliser pour basculer le contenu pour l'impression.

Des cours	Navigateur	Impression
<code>.visible-print-block</code> <code>.visible-print-inline</code> <code>.visible-print-inline-block</code>	Caché	Visible
<code>.hidden-print</code>	Visible	Caché

La classe `.visible-print` existe aussi mais est **obsolète** à partir de v3.2.0. Il est à peu près équivalent à l'`.visible-print-block` exception des cas spéciaux supplémentaires pour les `<table>` éléments concernant la PI.

Les cas de test

Redimensionner votre navigateur ou la charge sur différents appareils pour tester les classes utilitaires sensibles.

Visible sur ...

Coches vertes indiquent l'élément **est visible** dans votre fenêtre actuelle.

✓ Visible sur x-small

Petit

Moyen

grand

✓ Visible sur x-petits et petits	Moyennes et grandes
✓ Visible sur x-petites et moyennes	Petits et grands
✓ Visible sur x-petits et grands	Les petites et moyennes

Caché sur ...

Ici, des coches vertes indiquent également l'élément **est caché** dans votre fenêtre actuelle.

✓ Caché sur x-small	Petit
Moyen	grand
✓ Caché sur x-petits et petits	Moyennes et grandes
✓ Caché sur x-petites et moyennes	Petits et grands
✓ Caché sur x-petits et grands	Les petites et moyennes

avec moins

Le CSS de Bootstrap est construit sur Less, un préprocesseur avec des fonctionnalités supplémentaires comme des variables, mixins, et les fonctions de compilation CSS. Ceux qui cherchent à utiliser la source Moins fichiers au lieu de nos fichiers CSS compilés peuvent utiliser les nombreuses variables et mixins que nous utilisons dans l'ensemble du cadre.

Les variables de la grille et mixins sont couverts dans la section de système de grille .

Compiler Bootstrap

Bootstrap peut être utilisé dans au moins deux façons: avec le CSS compilé ou moins avec la source des fichiers. Pour compiler les fichiers Moins, consultez la section Mise en route pour savoir comment configurer votre environnement de développement pour exécuter les commandes nécessaires.

Troisième outils de compilation du parti peuvent travailler avec Bootstrap, mais ils ne sont pas pris en charge par notre équipe de base.

Variables

Les variables sont utilisées tout au long de l'ensemble du projet comme un moyen de centraliser et de partager des valeurs couramment utilisées comme les couleurs, l'espacement, ou des piles de police. Pour plus de détails, s'il vous plaît voir le Customizer .

Les couleurs

Faites facilement utiliser deux schémas de couleurs: en niveaux de gris et sémantique. Couleurs Grayscale fournissent un accès rapide aux nuances couramment utilisées noir alors que sémantique comprennent différentes couleurs attribuées aux valeurs contextuelles significatives.

EXAMPLE

```
@gray-darker: lighten(#000, 13.5%); // #222
@gray-dark:   lighten(#000, 20%);   // #333
@gray:       lighten(#000, 33.5%); // #555
@gray-light:  lighten(#000, 46.7%); // #777
@gray-lighter: lighten(#000, 93.5%); // #eee
```

EXAMPLE

```
@brand-primary: darken(#428bca, 6.5%); // #337ab7
@brand-success: #5cb85c;
@brand-info:    #5bc0de;
@brand-warning: #f0ad4e;
@brand-danger:  #d9534f;
```

Utilisez l'une de ces variables de couleur comme ils sont ou les réaffecter à des variables plus significatives pour votre projet.

```
// Use as-is
.masthead {
  background-color: @brand-primary;
}

// Reassigned variables in Less
@alert-message-background: @brand-info;
.alert {
  background-color: @alert-message-background;
}
```

Échafaudage

Une poignée de variables permettant de personnaliser rapidement les éléments clés du squelette de votre site.

```
// Scaffolding
@body-bg:  #fff;
@text-color: @black-50;
```

Des liens

styliser facilement vos liens avec la bonne couleur avec une seule valeur.

```
// Variables
@link-color:      @brand-primary;
@link-hover-color: darken(@link-color, 15%);

// Usage
a {
  color: @link-color;
  text-decoration: none;

  &:hover {
    color: @link-hover-color;
    text-decoration: underline;
  }
}
```

Notez que l' `@link-hover-color` utilise une fonction, un autre outil génial de moins, pour créer automagically la couleur hover droite. Vous pouvez utiliser `darken`, `lighten`, `saturate` et `desaturate`.

Typographie

Configurez facilement votre police de caractères, la taille du texte, ce qui conduit, et plus encore avec quelques variables rapides. Bootstrap fait usage de ceux-ci ainsi de fournir mixins typographiques faciles.

```
@font-family-sans-serif: "Helvetica Neue", Helvetica, Arial, sans-serif;
@font-family-serif: Georgia, "Times New Roman", Times, serif;
@font-family-monospace: Menlo, Monaco, Consolas, "Courier New", monospace;
@font-family-base: @font-family-sans-serif;

@font-size-base: 14px;
@font-size-large: ceil(@font-size-base * 1.25)); // ~18px
@font-size-small: ceil(@font-size-base * 0.85)); // ~12px

@font-size-h1: floor(@font-size-base * 2.6)); // ~36px
@font-size-h2: floor(@font-size-base * 2.15)); // ~30px
@font-size-h3: ceil(@font-size-base * 1.7)); // ~24px
@font-size-h4: ceil(@font-size-base * 1.25)); // ~18px
@font-size-h5: @font-size-base;
@font-size-h6: ceil(@font-size-base * 0.85)); // ~12px

@line-height-base: 1.428571429; // 20/14
@line-height-computed: floor(@font-size-base * @line-height-base)); // ~20px

@headings-font-family: inherit;
@headings-font-weight: 500;
@headings-line-height: 1.1;
@headings-color: inherit;
```

Icônes

Deux variables rapides pour la personnalisation de l'emplacement et le nom de vos icônes.

```
@icon-font-path: "../fonts/";
@icon-font-name: "glyphicon-halflings-regular";
```

Composants

Composants à travers Bootstrap font usage de certaines variables par défaut pour le réglage des valeurs communes. Ici sont le plus souvent utilisé le.

```
@padding-base-vertical: 6px;
@padding-base-horizontal: 12px;

@padding-large-vertical: 10px;
@padding-large-horizontal: 16px;

@padding-small-vertical: 5px;
@padding-small-horizontal: 10px;

@padding-xs-vertical: 1px;
@padding-xs-horizontal: 5px;

@line-height-large: 1.33;
@line-height-small: 1.5;

@border-radius-base: 4px;
@border-radius-large: 6px;
@border-radius-small: 3px;

@component-active-color: #fff;
@component-active-bg: @brand-primary;

@caret-width-base: 4px;
@caret-width-large: 5px;
```

mixins Vendor

mixins Vendor sont mixins pour aider à soutenir plusieurs navigateurs en incluant tous les préfixes des fournisseurs pertinents dans votre CSS compilé.

Box-dimensionnement

Réinitialiser le modèle de boîte de vos composants avec un seul mixin. Pour le contexte, consultez cet article utile de Mozilla .

Le mixin est **dépréciée** à partir de v3.2.0, avec l'introduction de Autoprefixer. Pour préserver la compatibilité ascendante, Bootstrap continuera à utiliser le mixin interne jusqu'à Bootstrap v4.

```
.box-sizing(@box-model) {
  -webkit-box-sizing: @box-model; // Safari <= 5
  -moz-box-sizing: @box-model; // Firefox <= 19
  box-sizing: @box-model;
}
```

Coups arrondis

Aujourd'hui , tous les navigateurs modernes prennent en charge la non-préfixée border-radius propriété. En tant que tel, il n'y a pas .border-radius() mixin, mais Bootstrap inclut des raccourcis pour arrondir rapidement deux coins d'un côté particulier d'un objet.

```
.border-top-radius(@radius) {
  border-top-right-radius: @radius;
  border-top-left-radius: @radius;
}
.border-right-radius(@radius) {
  border-bottom-right-radius: @radius;
  border-top-right-radius: @radius;
}
.border-bottom-radius(@radius) {
  border-bottom-right-radius: @radius;
  border-bottom-left-radius: @radius;
}
.border-left-radius(@radius) {
  border-bottom-left-radius: @radius;
  border-top-left-radius: @radius;
}
```

Box (drop) ombres

Si votre public cible utilise les navigateurs et appareils plus récents et les plus grands, assurez - vous d'utiliser seulement la box-shadow propriété sur son propre. Si vous avez besoin de soutien pour Android plus (pré-v4) et les appareils iOS (pré-iOS 5), utilisez le **dépréciée** mixin pour ramasser le nécessaire -webkit préfixe.

Le mixin est **dépréciée** à partir de v3.1.0, depuis Bootstrap ne supporte pas officiellement les plates - formes obsolètes qui ne supportent pas la propriété standard. Pour préserver la compatibilité ascendante, Bootstrap continuera à utiliser le mixin interne jusqu'à Bootstrap v4.

Assurez - vous d'utiliser des rgba() couleurs dans vos ombres de la boîte afin qu'ils se fondent de manière aussi transparente que possible avec les milieux.

```
.box-shadow(@shadow: 0 1px 3px rgba(0,0,0,.25)) {
  -webkit-box-shadow: @shadow; // iOS <4.3 & Android <4.1
  box-shadow: @shadow;
}
```

Transitions

mixins multiples pour une flexibilité. Définissez toutes les informations de transition avec un, ou spécifier un retard et la durée distincte au besoin.

Les mixins sont **déconseillés** comme des v3.2.0, avec l'introduction de Autoprefixer. Pour préserver la compatibilité ascendante, Bootstrap continuera à utiliser les mixins interne jusqu'à Bootstrap v4.

```
.transition(@transition) {
  -webkit-transition: @transition;
  transition: @transition;
}
.transition-property(@transition-property) {
  -webkit-transition-property: @transition-property;
  transition-property: @transition-property;
}
.transition-delay(@transition-delay) {
  -webkit-transition-delay: @transition-delay;
  transition-delay: @transition-delay;
}
.transition-duration(@transition-duration) {
  -webkit-transition-duration: @transition-duration;
  transition-duration: @transition-duration;
}
.transition-timing-function(@timing-function) {
  -webkit-transition-timing-function: @timing-function;
  transition-timing-function: @timing-function;
}
.transition-transform(@transition) {
  -webkit-transition: -webkit-transform @transition;
  -moz-transition: -moz-transform @transition;
  -o-transition: -o-transform @transition;
  transition: transform @transition;
}
```

Transformations

Rotation, échelle, traduire (déplacer) ou incliner un objet quelconque.

Les mixins sont **déconseillés** comme des v3.2.0, avec l'introduction de Autoprefixer. Pour préserver la compatibilité ascendante, Bootstrap continuera à utiliser les mixins interne jusqu'à Bootstrap v4.

```
.rotate(@degrees) {
  -webkit-transform: rotate(@degrees);
  -ms-transform: rotate(@degrees); // IE9 only
  transform: rotate(@degrees);
}

.scale(@ratio; @ratio-y...) {
  -webkit-transform: scale(@ratio, @ratio-y);
  -ms-transform: scale(@ratio, @ratio-y); // IE9 only
  transform: scale(@ratio, @ratio-y);
}

.translate(@x; @y) {
  -webkit-transform: translate(@x, @y);
  -ms-transform: translate(@x, @y); // IE9 only
  transform: translate(@x, @y);
}

.skew(@x; @y) {
  -webkit-transform: skew(@x, @y);
  -ms-transform: skewX(@x) skewY(@y); // See https://github.com/twbs/bootstrap/issues/4885; IE9+
  transform: skew(@x, @y);
}

.translate3d(@x; @y; @z) {
  -webkit-transform: translate3d(@x, @y, @z);
  transform: translate3d(@x, @y, @z);
}

.rotateX(@degrees) {
  -webkit-transform: rotateX(@degrees);
  -ms-transform: rotateX(@degrees); // IE9 only
  transform: rotateX(@degrees);
}

.rotateY(@degrees) {
  -webkit-transform: rotateY(@degrees);
  -ms-transform: rotateY(@degrees); // IE9 only
  transform: rotateY(@degrees);
}

.perspective(@perspective) {
  -webkit-perspective: @perspective;
  -moz-perspective: @perspective;
  perspective: @perspective;
}

.perspective-origin(@perspective) {
  -webkit-perspective-origin: @perspective;
  -moz-perspective-origin: @perspective;
  perspective-origin: @perspective;
}

.transform-origin(@origin) {
  -webkit-transform-origin: @origin;
  -moz-transform-origin: @origin;
  -ms-transform-origin: @origin; // IE9 only
  transform-origin: @origin;
}
```

Animations

Un mixin unique pour l'utilisation de toutes les propriétés d'animation de CSS3 dans une déclaration et d'autres mixins pour les propriétés individuelles.

Les mixins sont **déconseillés** comme des v3.2.0, avec l'introduction de Autoprefixer. Pour préserver la compatibilité ascendante, Bootstrap continuera à utiliser les mixins interne jusqu'à Bootstrap v4.

```
.animation(@animation) {
  -webkit-animation: @animation;
  animation: @animation;
}
.animation-name(@name) {
  -webkit-animation-name: @name;
  animation-name: @name;
}
.animation-duration(@duration) {
  -webkit-animation-duration: @duration;
  animation-duration: @duration;
}
.animation-timing-function(@timing-function) {
  -webkit-animation-timing-function: @timing-function;
  animation-timing-function: @timing-function;
}
.animation-delay(@delay) {
  -webkit-animation-delay: @delay;
  animation-delay: @delay;
}
.animation-iteration-count(@iteration-count) {
  -webkit-animation-iteration-count: @iteration-count;
  animation-iteration-count: @iteration-count;
}
.animation-direction(@direction) {
  -webkit-animation-direction: @direction;
  animation-direction: @direction;
}
```

Opacité

Réglez l'opacité pour tous les navigateurs et de fournir une `filter` solution de repli pour IE8.

```
.opacity(@opacity) {
  opacity: @opacity;
  // IE8 filter
  @opacity-ie: (@opacity * 100);
  filter: ~"alpha(opacity=@{opacity-ie})";
}
```

Espace réservé

Fournir un contexte pour les contrôles de formulaire dans chaque domaine.

```
.placeholder(@color: @input-color-placeholder) {
  &:::moz-placeholder { color: @color; } // Firefox
  &:-ms-input-placeholder { color: @color; } // Internet Explorer 10+
  &:::webkit-input-placeholder { color: @color; } // Safari and Chrome
}
```

Colonnes

Générer des colonnes via CSS dans un seul élément.

```
.content-columns(@width; @count; @gap) {
  -webkit-column-width: @width;
  -moz-column-width: @width;
  column-width: @width;
  -webkit-column-count: @count;
  -moz-column-count: @count;
  column-count: @count;
  -webkit-column-gap: @gap;
  -moz-column-gap: @gap;
  column-gap: @gap;
}
```

Dégradés

Transformez facilement deux couleurs dans un dégradé de fond. Obtenez plus avancée et définir une direction, utiliser trois couleurs, ou d'utiliser un dégradé radial. Avec une seule mixin vous obtenez tous les syntaxes préfixés dont vous aurez besoin.

```
#gradient > .vertical(#333; #000);
#gradient > .horizontal(#333; #000);
#gradient > .radial(#333; #000);
```

Vous pouvez également spécifier l'angle de deux couleurs, gradient linéaire standard:

```
#gradient > .directional(#333; #000; 45deg);
```

Si vous avez besoin d'un gradient de style barber-bande, qui est facile, aussi. Il suffit de spécifier une seule couleur et nous allons superposer une bande blanche translucide.

```
#gradient > .striped(#333; 45deg);
```

Jusqu'à l'ante et utiliser trois couleurs à la place. Définissez la première couleur, la deuxième couleur, l'arrêt de la couleur de la deuxième couleur (une valeur de pourcentage, comme 25%), et la troisième couleur avec ces mixins:

```
#gradient > .vertical-three-colors(#777; #333; 25%; #000);
#gradient > .horizontal-three-colors(#777; #333; 25%; #000);
```

Heads up! Si jamais vous avez besoin de supprimer un gradient, assurez - vous d'enlever tout IE spécifique que filter vous avez peut - être ajouté. Vous pouvez le faire en utilisant le .reset-filter() mixin à côté background-image: none; .

mixins utilitaires

mixins utilitaires sont mixins qui combinent des propriétés CSS autrement indépendants pour atteindre un objectif ou d'une tâche spécifique.

clearfix

Oubliez l' ajout class="clearfix" d'un élément quelconque et au lieu d' ajouter le .clearfix() mixin le cas échéant. Utilise le micro clearfix de Nicolas Gallagher .

```
// Mixin
.clearfix() {
  &:before,
  &:after {
    content: " ";
    display: table;
  }
  &:after {
    clear: both;
  }
}

// Usage
.container {
  .clearfix();
}
```

centrage horizontal

Centrer rapidement tout élément dans son parent. **Nécessite width ou max-width à régler.**

```
// Mixin
.center-block() {
  display: block;
  margin-left: auto;
  margin-right: auto;
}

// Usage
.container {
  width: 940px;
  .center-block();
}
```

Dimensionnement aides

Spécifiez les dimensions d'un objet plus facilement.

```
// Mixins
.size(@width; @height) {
  width: @width;
  height: @height;
}
.square(@size) {
  .size(@size; @size);
}

// Usage
.image { .size(400px; 300px); }
.avatar { .square(48px); }
```

textareas redimensionnables

Configurez facilement les options de redimensionnement pour tout textarea, ou tout autre élément. Par défaut , le comportement du navigateur normal (both).

```
.resizable(@direction: both) {
  // Options: horizontal, vertical,
  both
  resize: @direction;
  // Safari fix
  overflow: auto;
}
```

texte troncature

Tronquer facilement le texte avec une ellipse avec un seul mixin. **Nécessite élément à être block ou inline-block niveau.**

```
// Mixin
.text-overflow() {
  overflow: hidden;
  text-overflow: ellipsis;
  white-space: nowrap;
}

// Usage
.branch-name {
  display: inline-block;
  max-width: 200px;
  .text-overflow();
}
```

images Retina

Spécifiez deux chemins d'image et les @ 1x dimensions de l' image, et Bootstrap fournira une requête des médias @ 2x. **Si vous avez beaucoup d' images pour servir, envisager d' écrire votre rétine l' image CSS manuellement dans une seule requête des médias.**

Chemin	La description
	<pre>img-retina(@file-1x; @file-2x; @width-1x; @height-1x) { background-image: url("@{file-1x}"); @media only screen and (-webkit-min-device-pixel-ratio: 2), only screen and (min--moz-device-pixel-ratio: 2), only screen and (-o-min-device-pixel-ratio: 2/1), only screen and (min-device-pixel-ratio: 2), only screen and (min-resolution: 192dpi), only screen and (min-resolution: 2dppx) { background-image: url("@{file-2x}"); background-size: @width-1x @height-1x; } } // Usage .jumbotron { .img-retina("/img/bg-1x.png", "/img/bg-2x.png", 100px, 100px); }</pre>

Utilisation de Sass

Alors que Bootstrap est construit sur moins, il dispose également d' un port officiel Sass . Nous maintenons dans un référentiel GitHub séparé et traitons les mises à jour avec un script de conversion.

Ce qui est inclus

Depuis le port Sass a un repo séparé et sert un public légèrement différent, le contenu du projet très différent du projet Bootstrap principal. Cela garantit le port Sass est aussi compatible avec autant de systèmes basés sur Sass que possible.

Chemin	La description
lib/	Ruby code gem (configuration Sass, Rails et intégrations Compass)
tasks/	scripts Converter (en tournant amont Moins de Sass)
test/	des tests de compilation
templates/	package Compass manifeste
vendor/assets/	Sass, JavaScript et fichiers de police
Rakefile	tâches internes, tels que râteau et convert

Visitez le GitHub référentiel de ports Sass pour voir ces fichiers dans l' action.

Installation

Pour plus d' informations sur la façon d' installer et utiliser Bootstrap pour Sass, consulter le référentiel readme GitHub . Il est le plus à la source de la date et inclut des informations pour une utilisation avec Rails, Compass, et les projets Sass standard.

Bootstrap pour Sass

[GitHub](#) [Gazouillement](#) [Exemples](#) [Sur](#)

Conçu et construit avec tout l'amour du monde par [@mdo](#) et [@fat](#). Maintenu par l'équipe de base avec l'aide de nos collaborateurs.

Code de licence MIT, docs CC BY 3.0.