

# JavaScript

Donnez vie aux composants de Bootstrap avec plus d'une dizaine de plugins jQuery personnalisés. Ajoutez-les facilement tous à la fois, ou un par un.

## Vue d'ensemble

### Individuel ou compilé

Les plugins peuvent être inclus individuellement (à l'aide des fichiers `*.js` individuels de Bootstrap), ou d'un seul coup (à l'aide de `bootstrap.js` ou de la version minimisée `bootstrap.min.js`).

#### Utilisation du JavaScript compilé

`bootstrap.js` et `bootstrap.min.js` contiennent chacun tous les plugins dans un seul fichier. Utilisez-en un seul.

#### Attributs des données de composant

N'utilisez pas des attributs de données provenant de plusieurs plugins sur le même élément. Par exemple, un bouton ne peut pas avoir une infobulle et basculer sur une fenêtre modale. Pour réaliser ceci, utilisez un élément d'encadrement.

#### Dépendances des plugins

Some plugins and CSS components depend on other plugins. If you include plugins individually, make sure to check for these dependencies in the docs. Also note that all plugins depend on jQuery (this means jQuery must be included **before** the plugin files). Consult our `bower.json` (<https://github.com/twbs/bootstrap/blob/v3.2.0/bower.json>) to see which versions of jQuery are supported.

## Attributs de données

Vous pouvez utiliser tous les plugins de Bootstrap entièrement avec l'API de marquage sans écrire la moindre ligne de JavaScript. Il s'agit de l'API de première classe de Bootstrap et doit être votre premier choix lors de l'utilisation d'un plugin.

Cela étant, dans certaines situations, il peut s'avérer préférable de désactiver cette fonctionnalité. C'est pourquoi nous offrons la possibilité de désactiver l'API d'attributs de données en supprimant la liaison de tous les événements du document utilisant `data-api` pour l'espace de noms. Voici comment cela se présente :

```
$(document).off('.data-api')
```

Sinon, pour cibler un plugin particulier, ajoutez simplement le nom du plugin en tant qu'espace de noms ainsi que l'espace de noms `data-api` comme ceci :

```
$(document).off('.alert.data-api')
```

## API programmable

Nous pensons que vous devriez pouvoir utiliser tous les plugins Bootstrap uniquement à travers l'API JavaScript sans problèmes. Toutes les API publiques sont des méthodes simples, enchaînables, et renvoient la collection sur laquelle elles agissent.

```
$('.btn.danger').button('toggle').addClass('fat')
```

Toutes les méthodes doivent accepter un objet d'options facultatif, une chaîne qui cible une méthode particulière, ou rien (ce qui lance un plugin avec le comportement par défaut) :

```
$('#myModal').modal()           // initialized with defaults
$('#myModal').modal({ keyboard: false }) // initialized with no keyboard
$('#myModal').modal('show')     // initializes and invokes show immediately
```

Each plugin also exposes its raw constructor on a `Constructor` property: `$.fn.popover.Constructor`. If you'd like to get a particular plugin instance, retrieve it directly from an element:

```
$('#[rel="popover"%5D').data('popover').
```

## Aucun conflit

Il est parfois nécessaire d'utiliser des plugins Bootstrap avec d'autres frameworks d'interface utilisateur. Dans de tels cas, des collisions d'espaces de noms peuvent parfois se produire. Si cela arrive, vous pouvez appeler `.noConflict` sur le plugin dont vous souhaitez utiliser l'ancienne valeur.

```
var bootstrapButton = $.fn.button.noConflict() // return $.fn.button to
previously assigned value
$.fn.bootstrapBtn = bootstrapButton           // give $.fn.bootstrapBtn the
Bootstrap functionality
```

## Événements

Bootstrap provides custom events for most plugins' unique actions. Generally, these come in an infinitive and past participle form - where the infinitive (ex. `show`) is triggered at the start of an event, and its past participle form (ex. `shown`) is triggered on the completion of an action.

À partir de la version 3.0.0, tous les événements Bootstrap ont un espace de noms.

Tous les événements infinitifs proposent la fonctionnalité `preventDefault`. Ceci vous offre la possibilité d'arrêter l'exécution d'une action avant son démarrage.

```
$('#myModal').on('show.bs.modal', function (e) {  
  if (!data) return e.preventDefault() // stops modal from being shown  
})
```

### Bibliothèques tierces

**Bootstrap ne prend pas officiellement en charge les bibliothèques JavaScript tierces**, comme Prototype ou jQuery UI. Malgré `.noConflict` et les événements à espace de noms, vous pourriez avoir à résoudre des problèmes de compatibilité vous-même.

## Transitions `transition.js`

### À propos des transitions

Pour les effets de transition simples, ajoutez `transition.js` une seule fois avec les autres fichiers JS. Si vous utilisez `bootstrap.js` compilé (ou minimisé), inutile de l'inclure : il est déjà présent.

### Contenu du fichier

Transition.js est un système d'aide simple pour les événements `transitionEnd` ainsi qu'un émulateur de transitions CSS. Il est utilisé par les autres plugins pour vérifier la prise en charge des transitions CSS et détecter les transitions bloquées.

## Fenêtres modales `modal.js`

# Exemples

Les fenêtres modales sont rationalisées mais flexibles, des invites avec les fonctionnalités minimales requises et des valeurs par défaut intelligentes.

## Les fenêtres modales chevauchantes ne sont pas prises en charge

Veillez à ne pas ouvrir une fenêtre modale pendant qu'une autre est encore visible. Afficher plus d'une fenêtre modale à la fois nécessite un code personnalisé.

## Placement du marquage de fenêtre modale

Essayez toujours de place le code HTML d'une fenêtre modale dans une position supérieure de votre document pour éviter que d'autres composants n'affectent l'apparence ou le comportement de votre fenêtre modale.

## Mises en gardes pour les appareils mobiles

There are some caveats regarding using modals on mobile devices. See our browser support docs (</fr/docs/bootstrap/javascript/..../getting-started/#support-fixed-position-keyboards>) for details.

## Exemple statique

Une fenêtre modale rendue avec un en-tête, un corps et un jeu d'actions dans le pied de page.

### EXAMPLE

Titre de fenêtre modale

Un corps de rêve...

Fermer

Enregistrer les modifications

```
<div class="modal fade">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close" data-dismiss="modal"><span aria-
hidden="true">&times;</span><span class="sr-only">Close</span></button>
        <h4 class="modal-title">Modal title</h4>
      </div>
      <div class="modal-body">
        <p>One fine body&hellip;</p>
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-default" data-
dismiss="modal">Close</button>
        <button type="button" class="btn btn-primary">Save changes</button>
      </div>
    </div><!-- /.modal-content -->
  </div><!-- /.modal-dialog -->
</div><!-- /.modal -->
```

---

## Démonstration

Basculez sur une fenêtre modale avec JavaScript en cliquant sur le bouton ci-dessous. Elle défilera depuis le haut de l'écran et apparaîtra progressivement.

---

### EXAMPLE

Lancez la fenêtre modale de démonstration

```

<!-- Button trigger modal -->
<button class="btn btn-primary btn-lg" data-toggle="modal" data-
target="#myModal">
  Launch demo modal
</button>

<!-- Modal -->
<div class="modal fade" id="myModal" tabindex="-1" role="dialog" aria-
labelledby="myModalLabel" aria-hidden="true">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close" data-dismiss="modal"><span aria-
hidden="true">&times;</span><span class="sr-only">Close</span></button>
        <h4 class="modal-title" id="myModalLabel">Modal title</h4>
      </div>
      <div class="modal-body">
        ...
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-default" data-
dismiss="modal">Close</button>
        <button type="button" class="btn btn-primary">Save changes</button>
      </div>
    </div>
  </div>
</div>

```

## Rendre les fenêtres modales accessibles

Veillez à ajouter `role="dialog"` à `.modal`, et à l'attribut `aria-labelledby="myModalLabel"` pour faire référence au titre de fenêtre modale, et `aria-hidden="true"` pour indiquer aux technologies d'assistance d'ignorer les éléments DOM.

Vous pouvez de plus donner une description à votre fenêtre modale avec `aria-describedby` sur `.modal`.

## Embedding YouTube videos

Embedding YouTube videos in modals requires additional JavaScript not in Bootstrap to automatically stop playback and more. See this helpful Stack Overflow post (<http://stackoverflow.com/questions/18622508/bootstrap-3-and-youtube-in-modal>) for more information.

# Tailles optionnelles

Les fenêtres modales ont deux options de taille, disponibles via les classes de modificateur pour être placées sur un `.modal-dialog`.

## EXAMPLE

Grande fenêtre modale

Petite fenêtre modale

```
<!-- Large modal -->
<button class="btn btn-primary" data-toggle="modal" data-target=".bs-example-modal-lg">Large modal</button>
```

```
<div class="modal fade bs-example-modal-lg" tabindex="-1" role="dialog" aria-labelledby="myLargeModalLabel" aria-hidden="true">
  <div class="modal-dialog modal-lg">
    <div class="modal-content">
      ...
    </div>
  </div>
</div>
```

```
<!-- Small modal -->
<button class="btn btn-primary" data-toggle="modal" data-target=".bs-example-modal-sm">Small modal</button>
```

```
<div class="modal fade bs-example-modal-sm" tabindex="-1" role="dialog" aria-labelledby="mySmallModalLabel" aria-hidden="true">
  <div class="modal-dialog modal-sm">
    <div class="modal-content">
      ...
    </div>
  </div>
</div>
```

## Remove animation

For modals that simply appear rather than fade in to view, remove the `.fade` class from your modal markup.

```
<div class="modal" tabindex="-1" role="dialog" aria-labelledby="" aria-hidden="true">
  ...
</div>
```

# Utilisation

`.modal-open` au `<body>` pour outrepasser le comportement de défilement par défaut et génère un `.modal-backdrop` pour fournir une zone cliquable qui sert à révoquer les fenêtres modales affichées lorsqu'on clique à l'extérieur de la fenêtre modale.

## Via les attributs de données

Activez une fenêtre modale sans écrire de JavaScript. Définissez `data-toggle="modal"` sur un élément de contrôle, comme un bouton, avec un `data-target="#foo"` ou `href="#foo"` pour cibler une fenêtre modale spécifique sur laquelle basculer.

```
<button type="button" data-toggle="modal" data-target="#myModal">Launch modal</button>
```

## Via JavaScript

Appelez une fenêtre modale avec l'id `myModal` en une seule ligne de JavaScript :

```
$('#myModal').modal(options)
```

## Options

Vous pouvez passer des options via les attributs de données ou JavaScript. Pour les attributs de données, ajoutez le nom de l'option à `data-`, comme dans `data-backdrop=""`.

Nom	type	par défaut	description
backdrop	booléen ou la chaîne 'static'	true	Comprend un élément de fond de fenêtre modale. Vous pouvez aussi spécifier <code>static</code> pour un fond qui ne ferme pas la fenêtre modale lorsque vous cliquez dessus.
keyboard	booléen	true	Ferme la fenêtre modale si l'utilisateur appuie sur la touche Échap.
show	booléen	true	Affiche la fenêtre modale lors de l'initialisation.
remote	chemin	false	Si une URL distante est fournie, <b>le contenu sera chargé une fois</b> par la méthode <code>load</code> de jQuery et injectée dans le div <code>.modal-content</code> . Si vous utilisez l'API de données, vous pouvez utiliser l'attribut <code>href</code> de façon alternative pour spécifier la source distante. Un exemple de ceci est présenté ci-dessous : <div data-bbox="651 2024 1359 2089"><pre>&lt;a data-toggle="modal" href="remote.html" data-target="#myModal"&gt;Click me&lt;/a&gt;</pre></div>



[illegible]

## Méthodes

```
.modal(options)
```

Active votre contenu en tant que fenêtre modale. Accepte un `object` facultatif d'options.

```
$('#myModal').modal({
  keyboard: false
})
```

```
.modal('toggle')
```

Bascule manuellement une fenêtre modale. **Renvoie vers l'appelant avant que la fenêtre modale ne soit affichée ou cachée** (c'est à dire avant que l'événement `shown.bs.modal` ou `hidden.bs.modal` n'ait lieu).

```
$('#myModal').modal('toggle')
```

```
.modal('show')
```

Ouvre manuellement une fenêtre modale. **Renvoie vers l'appelant avant que la fenêtre modale ne soit affichée** (c'est à dire avant que l'événement `shown.bs.modal` n'ait lieu).

```
$('#myModal').modal('show')
```

```
.modal('hide')
```

Cache manuellement une fenêtre modale. **Renvoie vers l'appelant avant que la fenêtre modale ne soit cachée** (c'est à dire avant que l'événement `hidden.bs.modal` n'ait lieu).

```
$('#myModal').modal('hide')
```

## Événements

La classe « modal » de Bootstrap présente quelques événements pour aborder les fonctionnalités des fenêtres modales.

Type d'événement	Description
show.bs.modal	Cet événement se déclenche immédiatement lorsque la méthode d'instance <code>show</code> est appelée. Si c'est dû à un clic, l'élément cliqué est disponible en tant que propriété <code>relatedTarget</code> de l'événement.
shown.bs.modal	Cet événement est déclenché lorsque la fenêtre modale a été rendue visible

	pour l'utilisateur (il attend la fin des transitions CSS). Si c'est dû à un clic, l'élément cliqué est disponible en tant que propriété <code>relatedTarget</code> de l'événement.
<code>hide.bs.modal</code>	Cet événement est déclenché immédiatement lorsque la méthode d'instance <code>hide</code> a été appelée.
<code>hidden.bs.modal</code>	Cet événement est déclenché lorsque la fenêtre modale a été masquée pour l'utilisateur (il attend la fin des transitions CSS).
<code>loaded.bs.modal</code>	Cet événement est déclenché lorsque la fenêtre modale a chargé du contenu en utilisant l'option distante.

```
$('#myModal').on('hidden.bs.modal', function (e) {  
  // do something...  
})
```

## Menus déroulants dropdown.js

### Exemples

Ajoutez avec ce plugin simple des menus déroulants à quasiment tout élément, dont les barres de navigation, les onglets et les pilules,

#### Dans une barre de navigation

##### EXAMPLE

#### Dans des pilules

##### EXAMPLE

Lien normal    Menu déroulant ▼    Dropdown 2 ▼    Dropdown 3 ▼

# Utilisation

Le plugin dropdown bascule le contenu caché (menus déroulants) via des attributs de données ou Javascript en basculant la classe `.open` sur l'élément de liste parent. Lorsqu'il est ouvert, le plugin ajoute également `.dropdown-backdrop` comme zone cliquable pour fermer les menus déroulants lorsque l'utilisateur clique en dehors du menu. Notez bien que l'attribut `data-toggle=dropdown` est utilisé pour fermer les menus déroulants au niveau des applications, ce n'est donc pas une bonne idée de toujours l'utiliser.

## Via les attributs de données

Ajoutez `data-toggle="dropdown"` à un lien ou un bouton pour faire apparaître ou disparaître un menu déroulant.

```
<div class="dropdown">
  <a data-toggle="dropdown" href="#">Dropdown trigger</a>
  <ul class="dropdown-menu" role="menu" aria-labelledby="dLabel">
    ...
  </ul>
</div>
```

Pour conserver les URL intactes, utilisez l'attribut `data-target` au lieu de `href="#"`.

```
<div class="dropdown">
  <a id="dLabel" role="button" data-toggle="dropdown" data-target="#"
  href="/page.html">
    Dropdown <span class="caret"></span>
  </a>

  <ul class="dropdown-menu" role="menu" aria-labelledby="dLabel">
    ...
  </ul>
</div>
```

## Via JavaScript

Appeler les menus déroulants via JavaScript :

```
$( '.dropdown-toggle' ).dropdown()
```

`data-toggle="dropdown"` still required

Peu importe que vous invoquiez votre menu déroulant via JavaScript ou que vous utilisiez la data-api, `data-toggle="dropdown"` doit toujours être présent sur l'élément de déclenchement du menu déroulant.

## Options

*Aucun*

## Méthodes

`$.dropdown('toggle')`

Bascule le menu déroulant d'une barre de navigation ou d'une navigation à onglets particulière.

## Événements

Tous les événements de menu déroulant sont déclenchés sur l'élément parent du `.dropdown-menu`.

Type d'événement	Description
show.bs.dropdown	Cet événement se déclenche immédiatement lorsque la méthode d'instance <code>show</code> est appelée. L'élément ancre de basculement est disponible en tant que propriété <code>relatedTarget</code> de l'événement.
shown.bs.dropdown	Cet événement est déclenché lorsque le menu déroulant a été rendu visible pour l'utilisateur (il attend la fin des transitions CSS). L'élément ancre de basculement est disponible en tant que propriété <code>relatedTarget</code> de l'événement.
hide.bs.dropdown	Cet événement est déclenché lorsque la méthode d'instance <code>hide</code> a été appelée. L'élément ancre de basculement est disponible en tant que propriété <code>relatedTarget</code> de l'événement.
hidden.bs.dropdown	Cet événement est déclenché lorsque le menu déroulant a été complètement masqué pour l'utilisateur (il attend la fin des transitions CSS). L'élément ancre de basculement est disponible en tant que propriété <code>relatedTarget</code> de l'événement.

```
$('#myDropdown').on('show.bs.dropdown', function () {  
    // do something...  
})
```

# ScrollSpy `scrollspy.js`

## Exemple dans une barre de navigation

Le plugin ScrollSpy sert à actualiser automatiquement les cibles de navigation en fonction de la position de défilement. Défilez sur la zone en dessous de la barre de navigation et observez la modification de la classe active. Les sous-éléments du menu déroulant seront également mis en valeur.

### EXAMPLE

@fat

Ad leggings keytar, brunch id art party dolor labore. Pitchfork yr enim lo-fi before they sold out qui. Tumblr farm-to-table bicycle rights whatever. Anim keffiyeh carles cardigan. Velit seitan mcsweeney's photo booth 3 wolf moon irure. Cosby sweater lomo jean shorts, williamsburg hoodie minim qui you probably haven't heard of them et cardigan trust fund culpa biodiesel wes anderson aesthetic. Nihil tattooed accusamus, cred irony biodiesel keffiyeh artisan ullamco consequat.

@mdo

Veniam marfa mustache skateboard, adipisicing fugiat velit pitchfork beard. Freegan beard aliqua

## Utilisation

### Requires relative positioning

No matter the implementation method, scrollspy requires the use of `position: relative;` on the element you're spying on. In most cases this is the `<body>`.

### Via les attributs de données

Pour ajouter facilement un comportement scrollspy à votre barre de navigation supérieure, ajoutez `data-spy="scroll"` à l'élément que vous souhaitez surveiller (en général `<body>`), puis ajoutez l'attribut `data-target` avec l'ID ou la classe de l'élément parent à tout composant `.nav` de Bootstrap.

```
body {  
  position: relative;  
}
```

```
<body data-spy="scroll" data-target=".navbar-example">
  ...
  <div class="navbar-example">
    <ul class="nav nav-tabs" role="tablist">
      ...
    </ul>
  </div>
  ...
</body>
```

## Via JavaScript

After adding `position: relative;` in your CSS, call the scrollspy via JavaScript:

```
$('.body').scrollspy({ target: '.navbar-example' })
```

### Cibles d'ID résolubles requises

Les liens de la barre de navigation doivent avoir des cibles d'ID résolubles. Par exemple, un `<a href="#home">home</a>` doit correspondre à quelque chose dans le DOM comme `<div id="home"></div>`.

### Non-`:visible` target elements ignored

Target elements that are not `:visible` according to jQuery (<http://api.jquery.com/visible-selector/>) will be ignored and their corresponding nav items will never be highlighted.

## Méthodes

### `.scrollspy('refresh')`

Lors de l'utilisation d'un scrollspy avec l'ajout ou la suppression d'éléments du DOM, vous devrez appeler la méthode `refresh` comme ceci :

```
$('.[data-spy="scroll"]').each(function () {
  var $spy = $(this).scrollspy('refresh')
})
```

## Options

Vous pouvez passer des options via les attributs de données ou JavaScript. Pour les attributs de données, ajoutez le nom de l'option à `data-`, comme dans `data-offset=""`.

Nom	type	par défaut	description
offset	nombre	10	Pixels à décaler par rapport au sommet lors du calcul de la position d'un défilement.

## Événements

Type d'événement	Description
activate.bs.scrollspy	Cet événement se déclenche lorsqu'un nouvel élément est activé par le scrollspy.

```
$('#myScrollspy').on('activate.bs.scrollspy', function () {  
  // do something...  
})
```

# Onglets pouvant être basculés tab.js

## Exemples d'onglets

Ajoutez des fonctionnalités d'onglet dynamiques et rapides pour effectuer la transition entre des volets de contenu local, y compris avec les menus déroulants.

### EXAMPLE

Accueil	Profil	Menu déroulant ▼
---------	--------	------------------

Raw denim you probably haven't heard of them jean shorts Austin. Nesciunt tofu stumptown aliqua, retro synth master cleanse. Mustache cliche tempor, williamsburg carles vegan helvetica. Reprehenderit butcher retro keffiyeh dreamcatcher synth. Cosby sweater eu banh mi, qui irure terry richardson ex squid. Aliquip placeat salvia cillum iphone. Seitan aliquip quis cardigan american apparel, butcher voluptate nisi qui.

### Étend la navigation par onglets

This plugin extends the tabbed navigation component (</fr/docs/bootstrap/javascript/./components/#nav-tabs>) to add tabbable areas.

# Utilisation

Activer les onglets déclenchables via JavaScript (chaque onglet doit être activé séparément) :

```
$('#myTab a').click(function (e) {
  e.preventDefault()
  $(this).tab('show')
})
```

Vous pouvez activer des onglets individuels manuellement de plusieurs façons :

```
$('#myTab a[href="#profile"]').tab('show') // Select tab by name
$('#myTab a:first').tab('show') // Select first tab
$('#myTab a:last').tab('show') // Select last tab
$('#myTab li:eq(2) a').tab('show') // Select third tab (0-indexed)
```

## Marquage

You can activate a tab or pill navigation without writing any JavaScript by simply specifying `data-toggle="tab"` or `data-toggle="pill"` on an element. Adding the `nav` and `nav-tabs` classes to the `tab ul` will apply the Bootstrap tab styling (</fr/docs/bootstrap/javascript/..components/#nav-tabs>), while adding the `nav` and `nav-pills` classes will apply pill styling (</fr/docs/bootstrap/javascript/..components/#nav-pills>).

```
<!-- Nav tabs -->
<ul class="nav nav-tabs" role="tablist">
  <li class="active"><a href="#home" role="tab" data-toggle="tab">Home</a>
</li>
  <li><a href="#profile" role="tab" data-toggle="tab">Profile</a></li>
  <li><a href="#messages" role="tab" data-toggle="tab">Messages</a></li>
  <li><a href="#settings" role="tab" data-toggle="tab">Settings</a></li>
</ul>

<!-- Tab panes -->
<div class="tab-content">
  <div class="tab-pane active" id="home">...</div>
  <div class="tab-pane" id="profile">...</div>
  <div class="tab-pane" id="messages">...</div>
  <div class="tab-pane" id="settings">...</div>
</div>
```

## Effet de fondu

Pour faire apparaître les onglets en fondu, ajoutez `.fade` à chaque `.tab-pane`. Le premier carreau de l'onglet doit inclure `.in` pour faire apparaître correctement le contenu initial en fondu.



```
<div class="tab-content">
  <div class="tab-pane fade in active" id="home">...</div>
  <div class="tab-pane fade" id="profile">...</div>
  <div class="tab-pane fade" id="messages">...</div>
  <div class="tab-pane fade" id="settings">...</div>
</div>
```

## Méthodes

### `$.tab`

Active un élément d'onglet et le conteneur du contenu. L'onglet doit posséder un `data-target` ou un `href` ciblant un nœud du conteneur dans le DOM.

```
<ul class="nav nav-tabs" role="tablist" id="myTab">
  <li class="active"><a href="#home" role="tab" data-toggle="tab">Home</a>
</li>
  <li><a href="#profile" role="tab" data-toggle="tab">Profile</a></li>
  <li><a href="#messages" role="tab" data-toggle="tab">Messages</a></li>
  <li><a href="#settings" role="tab" data-toggle="tab">Settings</a></li>
</ul>

<div class="tab-content">
  <div class="tab-pane active" id="home">...</div>
  <div class="tab-pane" id="profile">...</div>
  <div class="tab-pane" id="messages">...</div>
  <div class="tab-pane" id="settings">...</div>
</div>

<script>
  $(function () {
    $('#myTab a:last').tab('show')
  })
</script>
```

## Événements

Type d'événement	Description
show.bs.tab	Cet événement se déclenche lors de l'affichage d'un onglet, mais avant l'affichage de celui-ci. Utilisez <code>event.target</code> et <code>event.relatedTarget</code> pour cibler l'onglets actif et l'onglet précédemment actif (si disponible), respectivement.
shown.bs.tab	Cet événement se déclenche lors de l'affichage d'un onglet, une fois que celui-ci est affiché. Utilisez <code>event.target</code> et <code>event.relatedTarget</code> pour cibler l'onglet actif et l'onglet précédemment actif (si disponible), respectivement.

```
$('#a[data-toggle="tab"]').on('shown.bs.tab', function (e) {  
  e.target // activated tab  
  e.relatedTarget // previous tab  
})
```

# Infobulles tooltip.js

## Exemples

Inspirées de l'excellent plugin jQuery.tipsy écrit par Jason Frame, les infobulles sont une version mise à jour qui ne se base pas sur des images, utilise CSS3 pour les animations et des attributs de données pour le stockage local du titre.

Survolez les liens ci-dessous pour voir des infobulles :

### EXAMPLE

Tight pants next level keffiyeh you probably haven't heard of them. Photo booth beard raw denim letterpress vegan messenger bag stumptown. Farm-to-table seitan, mcsweeney's fixie sustainable quinoa 8-bit american apparel have a terry richardson vinyl chambray. Beard stumptown, cardigans banh mi lomo thundercats. Tofu biodiesel williamsburg marfa, four loko mcsweeney's cleanse vegan chambray. A really ironic artisan whatever keytar, scenester farm-to-table banksy Austin twitter handle freegan cred raw denim single-origin coffee viral.

## Quatre directions

### EXAMPLE

Infobulle à gauche

Infobulle en haut

Infobulle en bas

Infobulle à droite

```
<button type="button" class="btn btn-default" data-toggle="tooltip" data-placement="left" title="Tooltip on left">Tooltip on left</button>
```

```
<button type="button" class="btn btn-default" data-toggle="tooltip" data-placement="top" title="Tooltip on top">Tooltip on top</button>
```

```
<button type="button" class="btn btn-default" data-toggle="tooltip" data-placement="bottom" title="Tooltip on bottom">Tooltip on bottom</button>
```

```
<button type="button" class="btn btn-default" data-toggle="tooltip" data-placement="right" title="Tooltip on right">Tooltip on right</button>
```

## Fonctionnalité opt-in

Pour des raisons de performance, les data-api des infobulles et des popovers sont définies en opt-in, ce qui veut dire que **vous devez les initialiser vous même** .

Les infobulles dans les groupes de boutons et les groupes de saisie requièrent des paramètres particuliers.

Lorsque vous utilisez des infobulles sur des éléments dans un `.btn-group` , ou un `.input-group` vous devez spécifier l'option `container: 'body'` (documentée ci-dessous) pour éviter les effets secondaires indésirables (élargissement de l'élément et/ou perte des coins arrondis lorsque l'infobulle est déclenchée).

## Don't try to show tooltips on hidden elements

Invoking `$(...).tooltip('show')` when the target element is `display: none;` will cause the tooltip to be incorrectly positioned.

## Les infobulles sur les éléments désactivés requièrent des éléments d'encadrement

Pour ajouter une infobulle à un élément `disabled` ou `.disabled` , insérez l'élément dans un `<div>` et appliquez plutôt l'infobulle à ce `<div>` .

# Utilisation

Le plugin d'infobulle génère du contenu et du marquage à la demande, et par défaut, il place les infobulles après leur élément déclencheur.

Déclenchez l'infobulle via JavaScript :

```
$('#example').tooltip(options)
```

## Marquage

Le marque requis pour une infobulle est seulement un attribut `data` et un `title` sur l'élément HTML auquel vous souhaitez ajouter une infobulle. Le marque généré d'une infobulle est plutôt simple, même s'il requiert une position (définie par défaut sur `top` par le plugin).

### Liens sur plusieurs lignes

Parfois vous souhaitez ajoutez une infobulle à un hyperlien qui englobe plusieurs lignes. Le comportement par défaut du plugin d'infobulle est de le centrer horizontalement et verticalement. Ajoutez `white-space: nowrap;` à vos ancres pour éviter cela.

```
<!-- HTML to write -->
<a href="#" data-toggle="tooltip" title="Some tooltip text!">Hover over me</a>

<!-- Generated markup by the plugin -->
<div class="tooltip top" role="tooltip">
  <div class="tooltip-arrow"></div>
  <div class="tooltip-inner">
    Some tooltip text!
  </div>
</div>
```

## Options

Vous pouvez passer des options via les attributs de données ou JavaScript. Pour les attributs de données, ajoutez le nom de l'option à `data-`, comme dans `data-animation=""`.

Nom	Type	Par défaut	Description
animation	booléen	true	Apply a CSS fade transition to the tooltip
container	chaîne   faux	false	Appends the tooltip to a specific element. Example: container: 'body'. This option is particularly useful in that it allows you to position the tooltip in the flow of the document near the triggering element - which will prevent the tooltip from floating away from the triggering element during a window resize.

delay	nombre   objet	0	<p>Delay showing and hiding the tooltip (ms) - does not apply to manual trigger type</p> <p>Si un nombre est fourni, le retard est appliqué à l'affichage et au masquage</p> <p>La structure de l'objet est : <code>delay: { show: 500, hide: 100 }</code></p>
html	booléen	false	<p>Insérez du HTML dans l'infobulle. Si la valeur est false, la méthode <code>text</code> de jQuery sera utilisée pour insérer le contenu dans le DOM. Utilisez du texte si vous craignez les attaques XSS.</p>
placement	chaîne   fonction	'top'	<p>How to position the tooltip - top   bottom   left   right   auto.</p> <p>When "auto" is specified, it will dynamically reorient the tooltip. For example, if placement is "auto left", the tooltip will display to the left when possible, otherwise it will display right.</p>
selector	chaîne	false	<p>If a selector is provided, tooltip objects will be delegated to the specified targets. In practice, this is used to enable dynamic HTML content to have tooltips added. See this (<a href="https://github.com/twbs/bootstrap/issues/4215">https://github.com/twbs/bootstrap/issues/4215</a>) and an informative example (<a href="http://jsfiddle.net/fScua/">http://jsfiddle.net/fScua/</a>).</p>
template	chaîne	'<div class="tooltip" role="tooltip"><div class="tooltip-arrow"></div><div class="tooltip-inner"></div></div>'	<p>Base HTML to use when creating the tooltip.</p> <p>The tooltip's <code>title</code> will be injected into the <code>.tooltip-inner</code>.</p> <p><code>.tooltip-arrow</code> will become the tooltip's arrow.</p> <p>The outermost wrapper element should have the <code>.tooltip</code> class.</p>
title	chaîne   fonction	"	<p>Default title value if <code>title</code> attribute isn't present</p>
trigger	chaîne	'hover focus'	<p>How tooltip is triggered - click   hover   focus   manual. You may pass multiple triggers; separate them with a space.</p>

viewport	string   object	{ selector: 'body', padding: 0 }	Keeps the tooltip within the bounds of this element. Example: viewport: '#viewport' or { selector: '#viewport', padding: 0 }
----------	-----------------	----------------------------------	---

## Attributs de données pour des infobulles individuelles

Des options pour les infobulles individuelles peuvent être spécifiées, de façon alternative, à travers l'utilisation d'attributs de données, comme expliqué ci-dessus.

## Méthodes

### \$.tooltip(options)

Ajoute un gestionnaire d'infobulle à une collection d'éléments.

### .tooltip('show')

Révèle l'infobulle d'un élément.

```
$('#element').tooltip('show')
```

### .tooltip('hide')

Masque l'infobulle d'un élément.

```
$('#element').tooltip('hide')
```

### .tooltip('toggle')

Bascule sur l'infobulle d'un élément.

```
$('#element').tooltip('toggle')
```

### .tooltip('destroy')

Masque et détruit l'infobulle d'un élément.

```
$('#element').tooltip('destroy')
```

## Événements

Type d'événement	Description
show.bs.tooltip	Cet événement se déclenche immédiatement lorsque la méthode d'instance <code>show</code> est appelée.

shown.bs.tooltip	Cet événement est déclenché lorsque l'infobulle a été rendue visible pour l'utilisateur (il attend la fin des transitions CSS).
hide.bs.tooltip	Cet événement est déclenché immédiatement lorsque la méthode d'instance <code>hide</code> a été appelée.
hidden.bs.tooltip	Cet événement est déclenché lorsque l'infobulle a été masquée pour l'utilisateur (il attend la fin des transitions CSS).

```
$('#myTooltip').on('hidden.bs.tooltip', function () {  
  // do something...  
})
```

# Popovers `popover.js`

## Exemples

Ajoutez des petites couches de contenu, comme sur iPad, à tout élément pour présenter des informations secondaires.

### Dépendance des plugins

Les popovers requièrent que le plugin d'infobulle soit inclus dans votre version de Bootstrap.

### Fonctionnalité opt-in

Pour des raisons de performance, les data-api des infobulles et des popovers sont définies en opt-in, ce qui veut dire que **vous devez les initialiser vous même** .

### Les popovers dans des groupes de boutons et groupes de saisie requièrent des paramètres particuliers

Lorsque vous utilisez des popovers sur des éléments dans un `.btn-group` , ou un `.input-group` vous devrez spécifier l'option `container: 'body'` (documentée ci-dessous) pour éviter les effets secondaires indésirables (élargissement de l'élément et/ou perte des coins arrondis lorsque le popover est déclenché).

## Don't try to show popovers on hidden elements

Invoking `$(...).popover('show')` when the target element is `display: none;` will cause the popover to be incorrectly positioned.

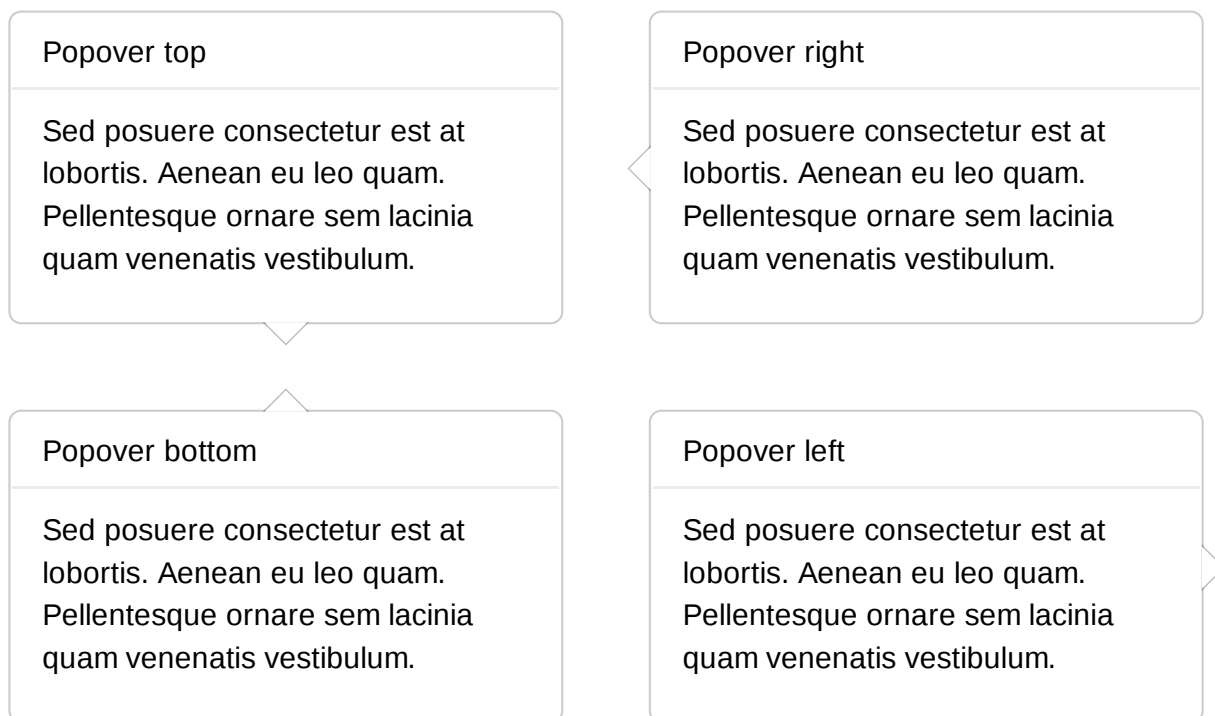
## Les popovers sur les éléments désactivés requièrent des éléments d'encadrement

Pour ajouter un popover à un élément `disabled` ou `.disabled`, insérez l'élément dans un `<div>` et appliquez plutôt le popover à ce `<div>`.

## Popover statique

Quatre options sont disponibles pour l'alignement : `top`, `right`, `bottom`, et `left`.

### EXAMPLE



## Démonstration



---

**EXAMPLE**

Cliquez pour basculer le popover

```
<button type="button" class="btn btn-lg btn-danger" data-toggle="popover"
title="Popover title" data-content="And here's some amazing content. It's very
engaging. Right?">Click to toggle popover</button>
```

---

## Quatre directions

---

**EXAMPLE**

Popover à gauche

Popover en haut

Popover en bas

Popover à droite

```
<button type="button" class="btn btn-default" data-container="body" data-
toggle="popover" data-placement="left" data-content="Vivamus sagittis lacus vel
augue laoreet rutrum faucibus.">
```

Popover on left

```
</button>
```

```
<button type="button" class="btn btn-default" data-container="body" data-
toggle="popover" data-placement="top" data-content="Vivamus sagittis lacus vel
augue laoreet rutrum faucibus.">
```

Popover on top

```
</button>
```

```
<button type="button" class="btn btn-default" data-container="body" data-
toggle="popover" data-placement="bottom" data-content="Vivamus
sagittis lacus vel augue laoreet rutrum faucibus.">
```

Popover on bottom

```
</button>
```

```
<button type="button" class="btn btn-default" data-container="body" data-
toggle="popover" data-placement="right" data-content="Vivamus sagittis lacus
vel augue laoreet rutrum faucibus.">
```

Popover on right

```
</button>
```

---

## Dismiss on next click

Use the `focus` trigger to dismiss popovers on the next click that the user makes.

## EXAMPLE

## Dismissible popover

```
<button type="button" class="btn btn-lg btn-danger popover-dismiss" data-  
toggle="popover" title="Dismissible popover" data-content="And here's some  
amazing content. It's very engaging. Right?">Dismissible popover</button>
```

```
$('.popover-dismiss').popover({  
  trigger: 'focus'  
})
```

## Liens sur plusieurs lignes

Parfois vous souhaitez ajouter un popover à un hyperlien qui englobe plusieurs lignes. Le comportement par défaut du plugin popover est de le centrer horizontalement et verticalement. Ajoutez `white-space: nowrap;` à vos ancres pour éviter ceci.

## Utilisation

Activez les popovers via JavaScript :

```
$('#example').popover(options)
```

## Options

Vous pouvez passer des options via les attributs de données ou JavaScript. Pour les attributs de données, ajoutez le nom de l'option à `data-`, comme dans `data-animation=""`.

Nom	Type	Par défaut	Description
animation	booléen	true	Apply a CSS fade transition to the popover
container	chaîne   faux	false	Ajoute le popover à un élément spécifique. Exemple : <code>container: 'body'</code> . Cette option est particulièrement utile car elle vous permet de positionner le popover dans le flux de documents à proximité de l'élément déclencheur, ce qui

			empêchera le popover de flotter à l'écart de l'élément déclencheur lors d'un redimensionnement de fenêtre.
content	chaîne   fonction	"	<p>Default content value if <code>data-content</code> attribute isn't present.</p> <p>If a function is given, it will be called with 1 argument, which is the element that the popover is attached to.</p>
delay	nombre   objet	0	<p>Delay showing and hiding the popover (ms) - does not apply to manual trigger type</p> <p>Si un nombre est fourni, le retard est appliqué à l'affichage et au masquage</p> <p>La structure de l'objet est : <code>delay: { show: 500, hide: 100 }</code></p>
html	booléen	false	Insérez du HTML dans le popover. Si la valeur est false, la méthode <code>text</code> de jQuery sera utilisée pour insérer du contenu dans le DOM. Utilisez du texte si vous craignez les attaques XSS.
placement	chaîne   fonction	'right'	<p>How to position the popover - top   bottom   left   right   auto.</p> <p>When "auto" is specified, it will dynamically reorient the popover. For example, if placement is "auto left", the popover will display to the left when possible, otherwise it will display right.</p>
selector	chaîne	false	If a selector is provided, popover objects will be delegated to the specified targets. In practice, this is used to enable dynamic HTML content to have popovers added. See this ( <a href="https://github.com/twbs/bootstrap/issues/4215">https://github.com/twbs/bootstrap/issues/4215</a> ) and an informative example ( <a href="http://jsfiddle.net/fScua/">http://jsfiddle.net/fScua/</a> ).
template	chaîne	'<div class="popover"'	Base HTML to use when creating the popover.

		<pre>role="tooltip"&gt; &lt;div class="arrow"&gt; &lt;/div&gt;&lt;h3 class="popover- title"&gt;&lt;/h3&gt; &lt;div class="popover- content"&gt;&lt;/div&gt; &lt;/div&gt;'</pre>	<p>The popover's <code>title</code> will be injected into the <code>.popover-title</code>.</p> <p>The popover's <code>content</code> will be injected into the <code>.popover-content</code>.</p> <p><code>.arrow</code> will become the popover's arrow.</p> <p>The outermost wrapper element should have the <code>.popover</code> class.</p>
title	chaîne   fonction	"	Default title value if <code>title</code> attribute isn't present
trigger	chaîne	'click'	How popover is triggered - click   hover   focus   manual. You may pass multiple triggers; separate them with a space.
viewport	string   object	{ selector: 'body', padding: 0 }	Keeps the popover within the bounds of this element. Example: <code>viewport: '#viewport'</code> or <code>{ selector: '#viewport', padding: 0 }</code>

## Attributs de données pour les popovers individuels

Des options peuvent être spécifiées de façon alternative pour des popovers individuels à travers l'utilisation d'attributs de données, comme expliqué ci-dessus.

## Méthodes

### `$.popover(options)`

Initialise les popovers pour une collection d'éléments.

### `.popover('show')`

Révèle le popover d'un élément.

```
$('#element').popover('show')
```

### `.popover('hide')`

Masque le popover d'un élément.

```
$('#element').popover('hide')
```

## .popover('toggle')

Bascule le popover d'un élément.

```
$('#element').popover('toggle')
```

## .popover('destroy')

Masque et détruit le popover d'un élément.

```
$('#element').popover('destroy')
```

## Événements

Type d'événement	Description
show.bs.popover	Cet événement se déclenche immédiatement lorsque la méthode d'instance <code>show</code> est appelée.
shown.bs.popover	Cet événement est déclenché lorsque le popover a été rendu visible pour l'utilisateur (il attendra la fin des transitions CSS).
hide.bs.popover	Cet événement est déclenché immédiatement lorsque la méthode d'instance <code>hide</code> a été appelée.
hidden.bs.popover	Cet événement est déclenché lorsque le popover a fini d'être masqué pour l'utilisateur (il attendra la fin des transitions CSS).

```
$('#myPopover').on('hidden.bs.popover', function () {  
    // do something...  
})
```

## Messages d'alerte `alert.js`

## Exemples d'alertes

Ajoutez la fonctionnalité de révocation à tous les messages d'alerte avec ce plugin.

### EXAMPLE

**Holy guacamole!** Best check yo self, you're not looking too good.



#### EXAMPLE

Oups ! Vous avez rencontré un problème !



Changez ceci et cela et réessayez. Duis mollis, est non commodo luctus, nisi erat porttitor ligula, eget lacinia odio sem nec elit. Cras mattis consectetur purus sit amet fermentum.

Prendre cette mesure

Ou faire ceci

## Utilisation

Activez la révocation d'une alerte via JavaScript :

```
$(".alert").alert()
```

## Marquage

Ajoutez simplement `data-dismiss="alert"` à votre bouton de fermeture pour ajouter automatiquement une fonctionnalité de fermeture d'alerte.

```
<button type="button" class="close" data-dismiss="alert"><span aria-hidden="true">&times;</span><span class="sr-only">Close</span></button>
```

## Méthodes

### `$.alert()`

Englobe toutes les alertes avec la fonctionnalité de fermeture. Pour que vos alertes s'animent lors de leur fermeture, vérifiez que les classes `.fade` et `.in` leur ont déjà été appliquées.

### `.alert('close')`

Ferme une alerte.

```
$(".alert").alert('close')
```

## Événements

La classe d'alerte de Bootstrap expose quelques événements pour aborder les fonctionnalités d'alerte.

Type d'événement	Description
close.bs.alert	Cet événement se déclenche immédiatement lorsque la méthode d'instance <code>close</code> est appelée.
closed.bs.alert	Cet événement est déclenché lorsque l'alerte a été fermée (il attendra la fin des transitions CSS).

```
$('#my-alert').on('closed.bs.alert', function () {  
  // do something...  
})
```

## Boutons `button.js`

### Exemples d'utilisation

Faites-en davantage avec des boutons. Les boutons de commande déclarent ou créent des groupes de boutons pour plus de composants, comme les barres d'outils.

#### À états

Ajoutez `data-loading-text="Loading..."` pour utiliser un état de chargement sur un bouton.

#### EXAMPLE

État de chargement

```
<button type="button" id="loading-example-btn" data-loading-text="Loading..."  
class="btn btn-primary">  
  Loading state  
</button>  
<script>  
  $('#loading-example-btn').click(function () {  
    var btn = $(this)  
    btn.button('loading')  
    $.ajax(...).always(function () {  
      btn.button('reset')  
    });  
  });  
</script>
```

## Basculement simple

Ajoutez `data-toggle="button"` pour activer le basculement sur un seul bouton.

### EXAMPLE

Basculement simple

```
<button type="button" class="btn btn-primary" data-toggle="button">Single toggle</button>
```

## Case à cocher

Ajoutez `data-toggle="buttons"` à un groupe de cases à cocher pour un basculement de style case à cocher sur `btn-group`.

### Pre-checked options need `.active`

For pre-checked options, you must add the `.active` class to the input's `label` yourself.

### EXAMPLE

☒ Option 1 (pre-checked) ☐ Option 2 ☐ Option 3

```
<div class="btn-group" data-toggle="buttons">
  <label class="btn btn-primary active">
    <input type="checkbox" checked> Option 1 (pre-checked)
  </label>
  <label class="btn btn-primary">
    <input type="checkbox"> Option 2
  </label>
  <label class="btn btn-primary">
    <input type="checkbox"> Option 3
  </label>
</div>
```

## Bouton radio

Ajoutez `data-toggle="buttons"` à un groupe de boutons radio pour un basculement de style bouton radio sur `btn-group`.



## Preselected options need `.active`

For preselected options, you must add the `.active` class to the input's `label` yourself.

### EXAMPLE

☒ Option 1 (preselected)

☐ Option 2

☐ Option 3

```
<div class="btn-group" data-toggle="buttons">
  <label class="btn btn-primary active">
    <input type="radio" name="options" id="option1" checked> Option 1
  (preselected)
  </label>
  <label class="btn btn-primary">
    <input type="radio" name="options" id="option2"> Option 2
  </label>
  <label class="btn btn-primary">
    <input type="radio" name="options" id="option3"> Option 3
  </label>
</div>
```

## Utilisation

Activer des boutons via JavaScript :

```
$('.btn').button()
```

## Marquage

Les attributs de données font partie intégrante du plugin de boutons. Consultez l'exemple de code suivant pour différents types de marquage.

## Options

*Aucun*

## Méthodes

`$.button('toggle')`

Bascule l'état de pression. Donne au bouton l'aspect d'un bouton activé.

## Basculement automatique

Vous pouvez activer le basculement automatique d'un bouton en utilisant l'attribut `data-toggle` .

```
<button type="button" class="btn btn-primary" data-toggle="button">...</button>
```

## `$.button('loading')`

Définit l'état du bouton comme celui de chargement en cours. Désactive le bouton et remplace son texte par celui du chargement. Le texte du chargement doit être défini dans l'élément de bouton avec l'attribut de données `data-loading-text` .

```
<button id="loading-example-btn" type="button" class="btn btn-primary" data-loading-text="loading stuff...">...</button>
<script>
  $('#loading-example-btn').click(function () {
    var btn = $(this)
    btn.button('loading')
    $.ajax(...).always(function () {
      btn.button('reset')
    });
  });
</script>
```

## Compatibilité sur les différents navigateurs

Firefox persists form control states across page loads (<https://github.com/twbs/bootstrap/issues/793>). A workaround for this is to use `autocomplete="off"` .

## `$.button('reset')`

Réinitialise l'état du bouton : remplace le texte par celui d'origine.

## `$.button(string)`

Réinitialise l'état du bouton : remplace le texte par tout texte d'état défini dans les données.

```
<button type="button" class="btn btn-primary" data-complete-text="finished!" >...</button>
<script>
  $('.btn').button('complete')
</script>
```

# Réduction collapse.js

## À propos

Obtenez des styles de base et une prise en charge flexible des composants réductibles comme les accordéons et la navigation.

### Dépendance des plugins

Le réduction requiert l'inclusion du plugin de transitions dans votre version de Bootstrap.

## Exemple d'accordéon

Avec le plugin de réduction, nous avons construit un accordéon simple en étendant le composant du panneau.

### EXAMPLE

#### Élément de groupe réductible #1

Anim pariatur cliche reprehenderit, enim eiusmod high life accusamus terry richardson ad squid. 3 wolf moon officia aute, non cupidatat skateboard dolor brunch. Food truck quinoa nesciunt laborum eiusmod. Brunch 3 wolf moon tempor, sunt aliqua put a bird on it squid single-origin coffee nulla assumenda shoreditch et. Nihil anim keffiyeh helvetica, craft beer labore wes anderson cred nesciunt sapiente ea proident. Ad vegan excepteur butcher vice lomo. Leggings occaecat craft beer farm-to-table, raw denim aesthetic synth nesciunt you probably haven't heard of them accusamus labore sustainable VHS.

#### Collapsible Group Item #2

#### Collapsible Group Item #3

```
<div class="panel-group" id="accordion">
  <div class="panel panel-default">
    <div class="panel-heading">
      <h4 class="panel-title">
        <a data-toggle="collapse" data-parent="#accordion" href="#collapseOne">
          Collapsible Group Item #1
        </a>
      </h4>
    </div>
  </div>
</div>
```

```

</div>
<div id="collapseOne" class="panel-collapse collapse in">
  <div class="panel-body">
    Anim pariatur cliche reprehenderit, enim eiusmod high life accusamus
    terry richardson ad squid. 3 wolf moon officia aute, non cupidatat skateboard
    dolor brunch. Food truck quinoa nesciunt laborum eiusmod. Brunch 3 wolf moon
    tempor, sunt aliqua put a bird on it squid single-origin coffee nulla assumenda
    shoreditch et. Nihil anim keffiyeh helvetica, craft beer labore wes anderson
    cred nesciunt sapiente ea proident. Ad vegan excepteur butcher vice lomo.
    Leggings occaecat craft beer farm-to-table, raw denim aesthetic synth nesciunt
    you probably haven't heard of them accusamus labore sustainable VHS.
  </div>
</div>
</div>
<div class="panel panel-default">
  <div class="panel-heading">
    <h4 class="panel-title">
      <a data-toggle="collapse" data-parent="#accordion" href="#collapseTwo">
        Collapsible Group Item #2
      </a>
    </h4>
  </div>
  <div id="collapseTwo" class="panel-collapse collapse">
    <div class="panel-body">
      Anim pariatur cliche reprehenderit, enim eiusmod high life accusamus
      terry richardson ad squid. 3 wolf moon officia aute, non cupidatat skateboard
      dolor brunch. Food truck quinoa nesciunt laborum eiusmod. Brunch 3 wolf moon
      tempor, sunt aliqua put a bird on it squid single-origin coffee nulla assumenda
      shoreditch et. Nihil anim keffiyeh helvetica, craft beer labore wes anderson
      cred nesciunt sapiente ea proident. Ad vegan excepteur butcher vice lomo.
      Leggings occaecat craft beer farm-to-table, raw denim aesthetic synth nesciunt
      you probably haven't heard of them accusamus labore sustainable VHS.
    </div>
  </div>
</div>
<div class="panel panel-default">
  <div class="panel-heading">
    <h4 class="panel-title">
      <a data-toggle="collapse" data-parent="#accordion"
href="#collapseThree">
        Collapsible Group Item #3
      </a>
    </h4>
  </div>
  <div id="collapseThree" class="panel-collapse collapse">
    <div class="panel-body">
      Anim pariatur cliche reprehenderit, enim eiusmod high life accusamus
      terry richardson ad squid. 3 wolf moon officia aute, non cupidatat skateboard
      dolor brunch. Food truck quinoa nesciunt laborum eiusmod. Brunch 3 wolf moon
      tempor, sunt aliqua put a bird on it squid single-origin coffee nulla assumenda

```

```
shoreditch et. Nihil anim keffiyeh helvetica, craft beer labore wes anderson  
cred nesciunt sapiente ea proident. Ad vegan excepteur butcher vice lomo.  
Leggings occaecat craft beer farm-to-table, raw denim aesthetic synth nesciunt  
you probably haven't heard of them accusamus labore sustainable VHS.  
    </div>  
  </div>  
</div>  
</div>
```

Vous pouvez également utiliser le plugin sans marquage d'accordéon. Utilisez un bouton pour étendre et réduire un autre élément.

```
<button type="button" class="btn btn-danger" data-toggle="collapse" data-  
target="#demo">  
  simple collapsible  
</button>  
  
<div id="demo" class="collapse in">...</div>
```

## Utilisation

Le plugin de réduction utilise quelques classes pour se charger des tâches les plus difficiles :

- `.collapse` hides the content
- `.collapse.in` shows the content
- `.collapsing` is added when the transition starts, and removed when it finishes

Ces classes peuvent être trouvées dans `component-animations.less`.

## Via les attributs de données

Ajoutez simplement `data-toggle="collapse"` et un `data-target` à l'élément pour affecter automatiquement le contrôle d'un élément réductible. L'attribut `data-target` accepte un sélecteur CSS auquel appliquer la réduction. Veillez à ajouter la classe `collapse` à l'élément réductible. Si vous souhaitez l'ouvrir par défaut, ajoutez la classe supplémentaire `in`.

Pour ajouter une gestion de groupe de type accordéon à une commande réductible, ajoutez l'attribut de données `data-parent="#selector"`. Consultez la démonstration pour voir cela en action.

## Via JavaScript

Activer manuellement avec :

```
$('.collapse').collapse()
```

## Options

Vous pouvez passer des options via les attributs de données ou JavaScript. Pour les attributs de données, ajoutez le nom de l'option à `data-`, comme dans `data-parent=""`.

Nom	type	par défaut	description
parent	selector	false	S'il s'agit d'un sélecteur, alors tous les éléments réductibles sous le parent spécifié seront fermés lorsque cet élément réductible est affiché (comme avec le comportement classique d'accordéon, cela est dépendant de la classe <code>panel</code> )
toggle	booléen	true	Bascule l'élément réductible lors de l'invocation

## Méthodes

### `.collapse(options)`

Active votre contenu en tant qu'élément réductible. Accepte un `object` d'options facultatif.

```
$('#myCollapsible').collapse({  
  toggle: false  
})
```

### `.collapse('toggle')`

Bascule un élément réductible sur affiché ou masqué.

### `.collapse('show')`

Affiche un élément réductible.

### `.collapse('hide')`

Masque un élément réductible.

## Événements

La classe de réduction de Bootstrap présente quelques événements pour aborder la fonctionnalité de réduction.

Type d'événement	Description
show.bs.collapse	Cet événement se déclenche immédiatement lorsque la méthode d'instance <code>show</code> est appelée.
shown.bs.collapse	Cet événement est déclenché lorsqu'un élément de réduction est rendu visible pour l'utilisateur (il attendra la fin des transitions CSS).
hide.bs.collapse	Cet événement est déclenché immédiatement lorsque la méthode <code>hide</code> a été

	appelée.
hidden.bs.collapse	Cet événement est déclenché lorsqu'un élément de réduction est masqué pour l'utilisateur (il attendra la fin des transitions CSS).

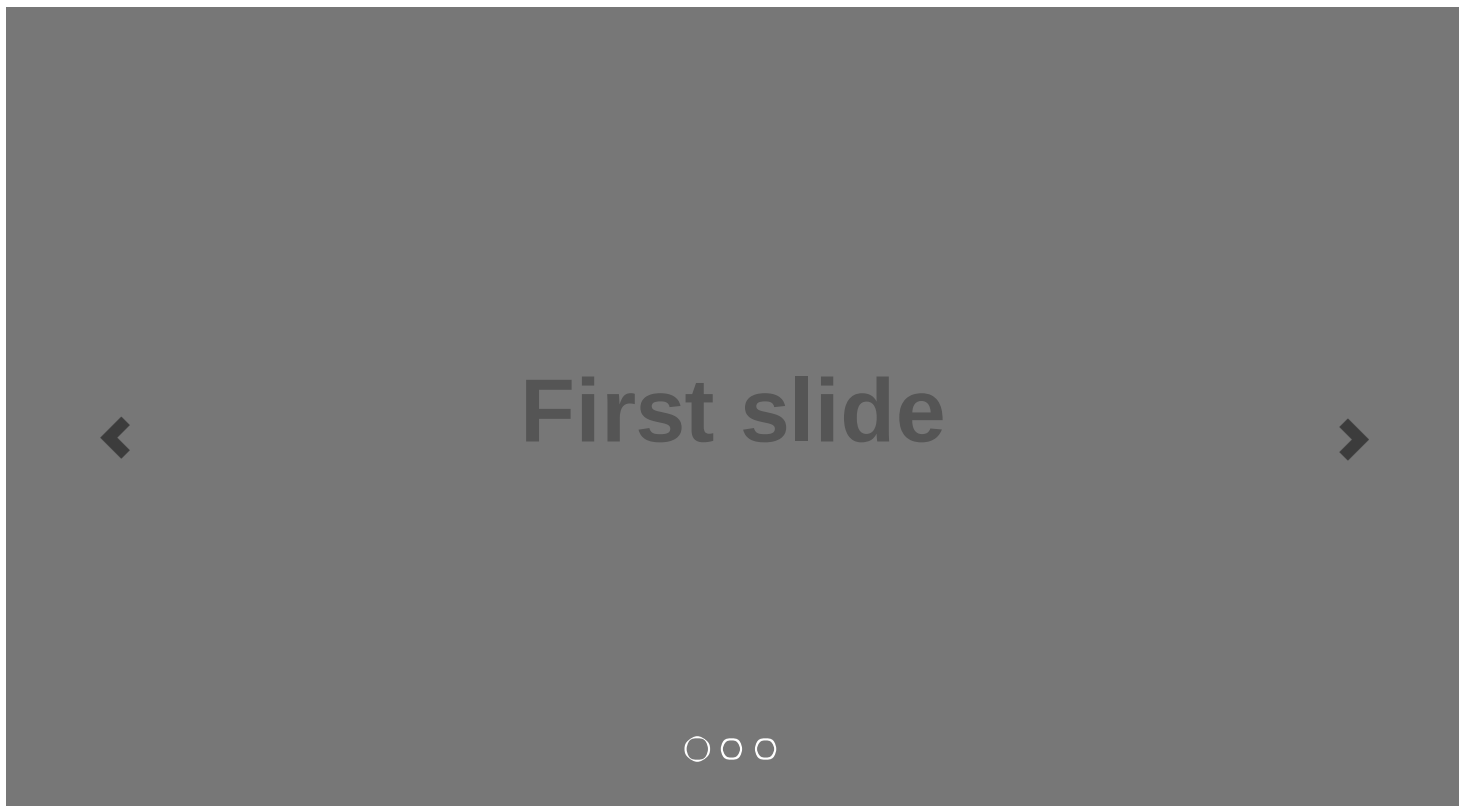
```
$('#myCollapsible').on('hidden.bs.collapse', function () {  
  // do something...  
})
```

# Carrousel `carousel.js`

## Exemples

Le diaporama ci-dessous présente un plugin et un composant génériques pour afficher des éléments en boucle, comme un carrousel.

### EXAMPLE



```

<div id="carousel-example-generic" class="carousel slide" data-ride="carousel">
  <!-- Indicators -->
  <ol class="carousel-indicators">
    <li data-target="#carousel-example-generic" data-slide-to="0"
class="active"></li>
    <li data-target="#carousel-example-generic" data-slide-to="1"></li>
    <li data-target="#carousel-example-generic" data-slide-to="2"></li>
  </ol>

  <!-- Wrapper for slides -->
  <div class="carousel-inner">
    <div class="item active">
      
      <div class="carousel-caption">
        ...
      </div>
    </div>
    <div class="item">
      
      <div class="carousel-caption">
        ...
      </div>
    </div>
    ...
  </div>

  <!-- Controls -->
  <a class="left carousel-control" href="#carousel-example-generic"
role="button" data-slide="prev">
    <span class="glyphicon glyphicon-chevron-left"></span>
  </a>
  <a class="right carousel-control" href="#carousel-example-generic"
role="button" data-slide="next">
    <span class="glyphicon glyphicon-chevron-right"></span>
  </a>
</div>

```

## Les animations de transition ne sont pas prises en charge dans Internet Explorer 8 et 9

Bootstrap utilise exclusivement CSS3 pour ses animations, mais Internet Explorer 8 et 9 ne prennent pas en charge les propriété CSS nécessaires. Ainsi, il n'y a pas d'animation de transition des diapos lors de l'utilisation de ces navigateurs. Nous avons délibérément décidé de ne pas inclure de fallbacks jQuery pour les transitions.

## Légendes optionnelles



Ajoutez facilement des légendes à vos diapositives avec l'élément `.carousel-caption` dans tout `.item`. Placez le code HTML optionnel de votre choix ici et celui-ci sera automatiquement aligné et formaté.

#### EXAMPLE



```
<div class="item">
  
  <div class="carousel-caption">
    <h3>...</h3>
    <p>...</p>
  </div>
</div>
```

#### Problème d'accessibilité

Le composant carrousel n'est pas généralement conforme aux normes d'accessibilité. Si vous devez être conforme, veuillez considérer d'autres options pour présenter votre contenu.

## Utilisation

### Carrousels multiples

Carousels require the use of an `id` on the outermost container (the `.carousel`) for carousel controls to function properly. When adding multiple carousels, or when changing a carousel's `id`, be sure to update the relevant controls.

## Via les attributs de données

Utilisez les attributs de données pour contrôler facilement la position du carrousel. `data-slide` accepte les mots-clés `prev` ou `next`, ce qui modifie la position de la diapositive par rapport à sa position actuelle. Vous pouvez également utiliser `data-slide-to` pour passer un index de diapositive brut au carrousel `data-slide-to="2"`, ce qui décale la position de la diapositive sur un index particulier commençant par `0`.

The `data-ride="carousel"` attribute is used to mark a carousel as animating starting at page load. **It cannot be used in combination with (redundant and unnecessary) explicit JavaScript initialization of the same carousel.**

## Via JavaScript

Appeler le carrousel manuellement avec :

```
$('.carousel').carousel()
```

## Options

Vous pouvez passer des options via les attributs de données ou JavaScript. Pour les attributs de données, ajoutez le nom de l'option à `data-`, comme dans `data-interval=""`.

Nom	type	par défaut	description
interval	nombre	5000	La durée de décalage entre les défilements automatiques d'éléments. Si la valeur est false, le carrousel ne défile pas automatiquement.
pause	chaîne	hover	Met en pause le cycle du carrousel lorsque le curseur entre dedans et le reprend lorsque la souris en sort.
wrap	booléen	true	Indique si le carrousel doit défiler en continu ou marquer des arrêts.

## Méthodes

### .carousel(options)

Initialise le carrousel avec un `object` d'options optionnel et commence à afficher les éléments en boucle.

```
$('.carousel').carousel({  
  interval: 2000  
})
```

### .carousel('cycle')

Affiche les éléments du carrousel en boucle, de gauche à droite.

### .carousel('pause')

Arrête la lecture des éléments du carrousel.

### .carousel(number)

Fait défiler le carrousel jusqu'à une trame particulière (en partant de 0, comme pour les vecteurs).

### .carousel('prev')

Affiche l'élément précédent.

### .carousel('next')

Affiche l'élément suivant.

## Événements

La classe « carousel » de Bootstrap présente deux événements pour aborder les fonctionnalités du carrousel.

Both events have the following additional properties:

- `direction` : The direction in which the carousel is sliding (either "left" or "right").
- `relatedTarget` : The DOM element that is being slid into place as the active item.

Type d'événement	Description
slide.bs.carousel	Cet événement se déclenche immédiatement lorsque l'instance de la méthode <code>slide</code> est invoquée.
slid.bs.carousel	Cet événement est déclenché lorsque le carrousel a terminé sa transition de diapositive.

```
$('#myCarousel').on('slide.bs.carousel', function () {  
  // do something..  
})
```

# Affixe affix.js

## Exemple

La sous-navigation sur la gauche est une démonstration du plugin d'affixation.

## Utilisation

Use the affix plugin via data attributes or manually with your own JavaScript. **In both situations, you must provide CSS for the positioning and width of your affixed content.**

## Positionnement via CSS

Le plugin d'affixation bascule entre trois classes, chacune représente un état particulier : `.affix`, `.affix-top` et `.affix-bottom`. Vous devez fournir les styles pour ces trois classes vous-même (indépendants de ce plugin) pour gérer les positions réelles.

Voici comment fonctionne le plugin « affix » :

1. Pour commencer, le plugin ajoute `.affix-top` pour indiquer que l'élément est dans sa position la plus élevée. A ce niveau, aucun positionnement CSS n'est requis.
2. Défiler au-delà de l'élément que vous voulez affixer devrait déclencher l'affixation. C'est ici que `.affix` remplace `.affix-top` et définit `position: fixed;` (fournie par le code CSS de Bootstrap).
3. Si un décalage de fond est défini, défiler au-delà devrait remplacer `.affix` par `.affix-bottom`. Vu que les décalages sont facultatifs, en définir un requiert que vous définissiez le CSS adapté. Dans ce cas, ajoutez `position: absolute;` lorsque c'est nécessaire. Le plugin utilise l'attribut de données ou l'option JavaScript pour déterminer où positionner l'élément depuis là-bas.

Suivez les étapes ci-dessus pour définir votre CSS pour l'une des options d'utilisation listées ci-dessous.

## Via les attributs de données

Pour ajouter facilement un comportement d'affixation à un élément, ajoutez simplement `data-spy="affix"` à l'élément que vous souhaitez surveiller. Puis utilisez les décalages pour définir quand activer ou désactiver la fixation d'un élément.

```
<div data-spy="affix" data-offset-top="60" data-offset-bottom="200">
  ...
</div>
```

## Via JavaScript

Appeler le plugin d'affixation via JavaScript :

```
$('#my-affix').affix({  
  offset: {  
    top: 100  
  }, bottom: function () {  
    return (this.bottom = $('.footer').outerHeight(true))  
  }  
})
```

## Options

Vous pouvez passer des options via les attributs de données ou JavaScript. Pour les attributs de données, ajoutez le nom de l'option à `data-`, comme dans `data-offset-top="200"`.

Nom	type	par défaut	description
offset	nombre   fonction   objet	10	Le nombre de pixels de décalage par rapport à l'écran lors du calcul de la position de défilement. Si un seul nombre est fourni, le décalage sera appliqué aux directions supérieure et inférieure. Pour fournir des décalages supérieur et inférieur différents, fournissez simplement un objet <code>offset: { top: 10 }</code> ou <code>offset: { top: 10, bottom: 5 }</code> . Utilisez une fonction lorsque vous avez besoin de calculer automatiquement un décalage.
target	selector   node   jQuery element	the window object	Specifies the target element of the affix.

## Événements

La classe « affix » de Bootstrap présente quelques événements pour aborder les fonctionnalités d'affixation.

Type d'événement	Description
affix.bs.affix	Cet événement se déclenche immédiatement avant que l'élément n'ait été affixé.
affixed.bs.affix	Cet événement se déclenche après que l'élément a été affixé.
affix-top.bs.affix	Cet événement se déclenche immédiatement avant que l'élément n'ait été affixé en haut.

affixed-top.bs.affix	Cet événement se déclenche après que l'élément a été affixé en haut.
affix-bottom.bs.affix	Cet événement se déclenche immédiatement avant que l'élément n'ait été affixé en bas.
affixed-bottom.bs.affix	Cet événement se déclenche après que l'élément a été affixé en bas.

Star

69,795

Fork

25,836

Suivre @twbootstrap

185K abonnés

Tweeter

4 76

Créé avec amour par @mdo (<http://twitter.com/mdo>) et @fat (<http://twitter.com/fat>).

Maintenu par l'équipe principale (<https://github.com/twbs?tab=members>) avec l'aide de nos contributeurs (<https://github.com/twbs/bootstrap/graphs/contributors>).

Code sous licence MIT (<https://github.com/twbs/bootstrap/blob/master/LICENSE>), documentation sous CC BY 3.0 (<http://creativecommons.org/licenses/by/3.0/>).

Currently v3.2.0 · GitHub (<https://github.com/twbs/bootstrap>) · Exemples (</fr/docs/bootstrap/javascript/..../getting-started/#examples>) · Documents de la version 2.3.2 (</fr/docs/bootstrap/javascript/..../2.3.2/>) · À propos (</fr/docs/bootstrap/javascript/..../about/>) · Exposition (<http://expo.getbootstrap.com>) · Blog (<http://blog.getbootstrap.com>) · Problèmes (<https://github.com/twbs/bootstrap/issues?state=open>) · Versions (<https://github.com/twbs/bootstrap/releases>)