

BOOTSTRAP CAROUSEL PLUGIN

http://www.tutorialspoint.com/bootstrap/bootstrap_carousel_plugin.htm

Copyright © tutorialspoint.com

The Bootstrap carousel is a flexible, responsive way to add a slider to your site. In addition to being responsive, the content is flexible enough to allow images, iframes, videos, or just about any type of content that you might want.

*If you want to include this plugin functionality individually, then you will need the **carousel.js**. Else, as mentioned in the chapter [Bootstrap Plugins Overview](#), you can include the **bootstrap.js** or the minified **bootstrap.min.js**.*

Example

A simple slideshow below shows a generic component for cycling through the elements like a carousel, using the Bootstrap carousel plugin. To implement the carousel, you just need to add the code with the markup. There is no need for data attributes, just simple class-based development.

```
<div >
  <!-- Carousel indicators -->
  <ol >
    <li data-target="#myCarousel" data-slide-to="0" ></li>
    <li data-target="#myCarousel" data-slide-to="1"></li>
    <li data-target="#myCarousel" data-slide-to="2"></li>
  </ol>
  <!-- Carousel items -->
  <div >
    <div >
      
    </div>
    <div >
      
    </div>
    <div >
      
    </div>
  </div>
  <!-- Carousel nav -->
  <a
    data-slide="prev">&l saquo;</a>
  <a
    data-slide="next">&r saquo;</a>
</div>
```



Optional Captions

You can add captions to your slides easily with the **.carousel-caption** element within any **.item**. Place just about any optional HTML within there and it will be automatically aligned and formatted. The following example demonstrates this:

```
<div >
  <!-- Carousel indicators -->
  <ol >
    <li data-target="#myCarousel" data-slide-to="0" ></li>
    <li data-target="#myCarousel" data-slide-to="1"></li>
    <li data-target="#myCarousel" data-slide-to="2"></li>
  </ol>
  <!-- Carousel items -->
  <div >
    <div >
      
      <div >This Caption 1</div>
    </div>
    <div >
      
      <div >This Caption 2</div>
    </div>
    <div >
      
      <div >This Caption 3</div>
    </div>
  </div>
  <!-- Carousel nav -->
  <a
    data-slide="prev">&lsaquo;</a>
  <a
    data-slide="next">&rsaquo;</a>
</div>
```



Usage

- **Via data attributes** : Use data attributes to easily control the position of the carousel.
 - Attribute **data-slide** accepts the keywords *prev* or *next*, which alters the slide position relative to its current position.
 - Use **data-slide-to** to pass a raw slide index to the carousel **data-slide-to="2"**, which shifts the slide position to a particular index beginning with 0.
 - The **data-ride="carousel"** attribute is used to mark a carousel as an animation starting at page load.
- **Via JavaScript** : The carousel can be manually called with JavaScript as below:

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

Options

There are certain, options which can be passed via data attributes or JavaScript are listed in the following table:

Option Name	Type/Default Value	Data attribute name	Description
interval	number <i>Default: 5000</i>	data-interval	The amount of time to delay between automatically cycling an item. If false, carousel will not automatically cycle.
pause	string <i>Default: "hover"</i>	data-pause	Pauses the cycling of the carousel on mouseenter and resumes the cycling of the carousel on mouseleave.
wrap	boolean <i>Default: true</i>	data-wrap	Whether the carousel should cycle continuously or have hard stops.

Methods

Here is a list of useful methods that can be used with carousel code.

Method	Description	Example
<code>.carousel options</code>	Initializes the carousel with an optional options object and starts cycling through items.	<pre>\$('#identifier').carousel({ interval: 2000 })</pre>
<code>.carousel 'cycle'</code>	Cycles through the carousel items from left to right.	<pre>\$('#identifier').carousel('cycle')</pre>
<code>.carousel 'pause'</code>	Stops the carousel from cycling through items.	<pre>\$('#identifier').carousel('pause')</pre>
<code>.carousel number</code>	Cycles the carousel to a particular frame 0based, similar to an array.	<pre>\$('#identifier').carousel(number)</pre>
<code>.carousel 'prev'</code>	Cycles to the previous item.	<pre>\$('#identifier').carousel('prev')</pre>
<code>.carousel 'next'</code>	Cycles to the next item.	<pre>\$('#identifier').carousel('next')</pre>

Example

The following example demonstrates the usage of methods:

```
<div >  
  <!-- Carousel indicators -->  
  <ol >  
    <li data-target="#myCarousel" data-slide-to="0"  
      ></li>
```

```

    <li data-target="#myCarousel" data-slide-to="1"></li>
    <li data-target="#myCarousel" data-slide-to="2"></li>
</ol>
<!-- Carousel items -->
<div >
    <div >
        
    </div>
    <div >
        
    </div>
    <div >
        
    </div>
</div>
<!-- Carousel nav -->
<a
    data-slide="prev">&lsaquo;</a>
<a
    data-slide="next">&rsaquo;</a>
<!-- Controls buttons -->
<div style="text-align:center;">
    <input type="button" >
    <input type="button" >
    <input type="button" >
    <input type="button" >
    <input type="button" >
    <input type="button" >
    <input type="button" >
</div>
</div>
<script>
$(function(){
    // Initializes the carousel
    $(".start-slide").click(function(){
        $("#myCarousel").carousel('cycle');
    });
    // Stops the carousel
    $(".pause-slide").click(function(){
        $("#myCarousel").carousel('pause');
    });
    // Cycles to the previous item
    $(".prev-slide").click(function(){
        $("#myCarousel").carousel('prev');
    });
    // Cycles to the next item
    $(".next-slide").click(function(){
        $("#myCarousel").carousel('next');
    });
    // Cycles the carousel to a particular frame
    $(".slide-one").click(function(){
        $("#myCarousel").carousel(0);
    });
    $(".slide-two").click(function(){
        $("#myCarousel").carousel(1);
    });
    $(".slide-three").click(function(){
        $("#myCarousel").carousel(2);
    });
});
</script>

```





Events

Bootstrap's carousel class exposes two events for hooking into carousel functionality which are listed in the following table.

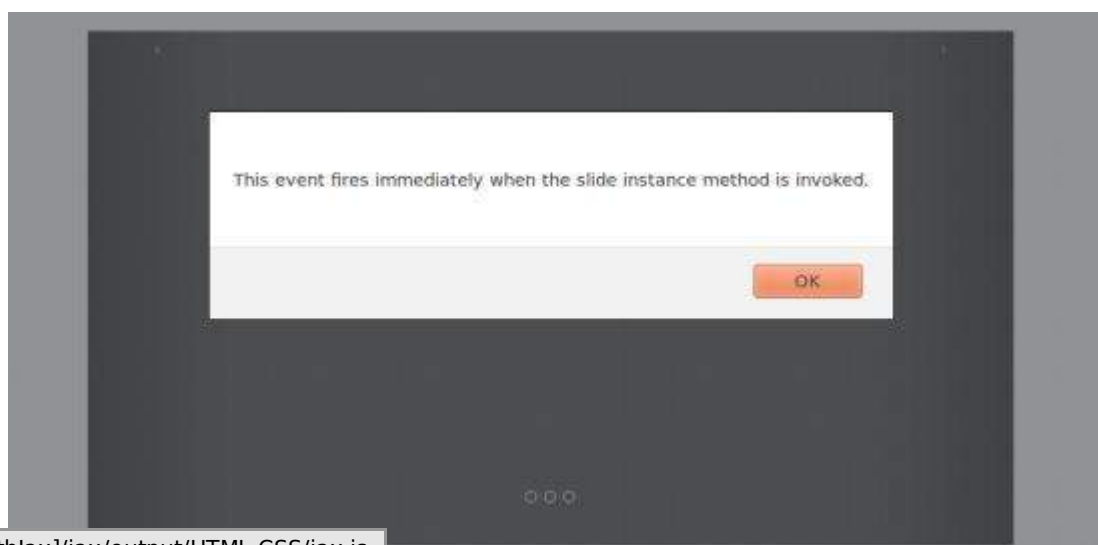
Event	Description	Example
slide.bs.carousel	This event fires immediately when the slide instance method is invoked..	<pre>\$('#identifier').on('slide.bs.carousel', function () { // do something... })</pre>
slid.bs.carousel	This event is fired when the carousel has completed its slide transition.	<pre>\$('#identifier').on('slid.bs.carousel', function () { // do something... })</pre>

Example

The following example demonstrates the usage of events:

```
<div >
  <!-- Carousel indicators -->
  <ol >
    <li data-target="#myCarousel" data-slide-to="0"
      ></li>
    <li data-target="#myCarousel" data-slide-to="1"></li>
    <li data-target="#myCarousel" data-slide-to="2"></li>
  </ol>
  <!-- Carousel items -->
  <div >
    <div >
      
    </div>
    <div >
      
    </div>
    <div >
      
    </div>
  </div>
  <!-- Carousel nav -->
  <a
    data-slide="prev">&lsaquo;</a>
  <a
    data-slide="next">&rsaquo;</a>
</div>
<script>
  $(function(){
    $('#myCarousel').on('slide.bs.carousel', function () {
```

```
    alert("This event fires immediately when the slide instance method"
        +"is invoked.");
    });
</script>
```



Loading [MathJax]/jax/output/HTML-CSS/jax.js