

## ReadMe

Thank you for using BootstrapGallery! This is a quick documentation for how to use this Gallery the most easiest and quickest way.

## Method 1

Before we are going to create a Gallery, im going to explain you some basic understandings for the use of the gallery. Lets start with the image, what does it need.

There are a few options we can give to the image:

- Image Caption
- Image Title
- Image ID

### Image title:

Render your images the way you would like it. If you have a image title, appoint it to the data attribute data-image-title. For example:

```

```

### Image Caption:

Create a data-attribute for the image-caption like data-image-caption. For example:

```

```

### Image ID:

We can do this on 2 ways, render it by yourself, or let the plugin render it for you. The most easiest way is to let the plugin render it for you. Isn't that great?! If you render it by yourself, add to your img attribute the data attribute data-image-id. For example:

```

```

Make sure that it is a digit! If its not, there is a change that the plugin isn't working well.

### Rendering the modal for showing the image:

Now that we rendered the images, we need a bootstrap Modal. Again, 2 options: let the plugin do his work, or by ourself, lets start!

### Manually:

Create a modal with a ID you preffer. In this example: #myModal (its also the default). Now we need something to call the modal: add a data-target and data-toggle to your <img>. Use for the data-toggle the value "modal" and for data-target the ID of the Bootstrap Modal dialogue. Don't forget the hashtag! For example:

```

```

Alright, we now know how to render the images with the attributes we would like. Lets start adding the script.

We have the following code:

```

<div class="item active">
  
</div>
<div class="item">
  
</div>
<div class="item">
  
</div>

```

I have wrapped my image into a class called “item”. We need the image, so we are going to get that image attribute on a easy way. Lets use the BootstrapGallery function. Put this code in your script:

```
$(“).BootstrapGallery();
```

We have now the basic. Lets point to our image, in my case: .item > img:

```
$('.item > img').BootstrapGallery();
```

We have some settings we can use, ill explain them for you.

To add settings:

```

$('.item > img').BootstrapGallery({
  imageIDs: false,
  nextButton: 'Volgende', (Volgende is Dutch for “next”)
  addWhatYouWantAndCheckTheListBelow: true,
});

```

Setting	Default value	Explanation
imageIDs	True (Boolean)	Defines if the data-image-id needs to be made by the plugin with a increment for it
nextButton	Next (string)	Text value for the next button
prevButton	Previous (string)	Text value for the previous button
nextButtonID	show-next-image (string)	ID for the next image trigger
prevButtonID	show-previous-image (string)	ID for the previous image trigger

setCaptionTo	#image-gallery-caption (string)	ID for the element where the caption needs to be appended
setTitleTo	#image-gallery-title (string)	ID for the element where the image title needs to be appended
setImageTo	#image-gallery-image (string)	ID where the image must show up. Its not required to use a Bootstrap Modal
buttonDataTarget	myModal (string)	Use here the ID you used for the Bootstrap Modal
createModalAttr	True (Boolean)	Adds all elements to the images to trigger the modal
createBootstrapModal	True (Boolean)	Generates a Bootstrap Modal by the plugin
developmentMode	False (Boolean)	Displays errors and checks in a alert box

## Method 2

We are now going for the easy way.

Lets take a look at my code:

```
<div class="item active">
  
</div>
<div class="item">
  
</div>
<div class="item">
  
</div>
```

As you see, I have rendered some images. What I'm missing, is the data-toggle and data-target attributes. Don't worry! Let the plugin do that for you!

To render the Bootstrap Modal Gallery nice and easy and most of all: quickly, we need only 1 line of code. Isn't that cool?!

So if I take a look at the code, my image is in my class="item". That will make my code like:

```
$('.item > img').BootstrapGallery();
```

VOILA! You're done! You have now a working gallery! You can use it also for sliders!

## Method 3

Maybe you don't use Bootstrap, so what! We can generate also images for other purposes or HTML frameworks! You can use the options for it in the list. Let me show you a example, then ill explain what it does.

```
<div class="item active">
  
</div>
<div class="item">
  
</div>
<div class="item">
  
</div>

<div id="myImageLoader">
  <h1 id="image-gallery-title">Here comes my image |title</h1>
  <img src="" id="image-gallery-image">
  <h3 id="image-gallery-caption">This place is where my image caption comes</h3>
</div>
```

I created a DIV with 3 attributes: a h1, img and h3. I gave some ID's to it. I used the plugin's default. In the options we can change that. When I click now on a image, then the image will be loaded into the myImageLoader DIV.