Bootstrap Images

In this tutorial you will learn how to style images, as well as how to create responsive images and videos using Bootstrap.

Styling Images with Bootstrap

Images are very common in modern web design. So styling images and placing it properly on the web pages is very important for improving the user experience.

Using the Bootstrap built-in classes you can easily style images such as making the round cornered or circular images, or give them effect like thumbnails.

Example	Try this code »

```
1  <img src="images/avatar.svg" class="rounded" alt="Rounded Image">
2  <img src="images/avatar.svg" class="rounded-circle" alt="Circular Image">
3  <img src="images/avatar.svg" class="img-thumbnail" alt="Thumbnail Image">
```

— The output of the above example will look something like this:

Creating Responsive Images and Videos

With Bootstrap you can make the images responsive too. Just add the class `.img-fluid` to the `<img>` tag. This class mainly applies the styles `max-width: 100%;` and `height: auto;` to the image so that it scales nicely to fit the containing element — in case if the width of the image is larger than the containing element itself. Check out the following example to see how it works:

Example	Try this code »

```
1  <img src="images/kites.jpg" class="img-fluid" alt="Flying Kites">
2  <img src="images/sky.jpg" class="img-fluid" alt="Cloudy Sky">
3  <img src="images/balloons.jpg" class="img-fluid" alt="Hot Air Balloons">
```

Note: When making the responsive layouts consider making your images responsive too, otherwise if an image width is larger than the parent element's width in any case, it will overflow and may break your web page layout.

You can also make the video or slideshow embedded in a web page responsive without affecting its original aspect ratio. To do this wrap any embed like an `<iframe>`, or `<video>` in a `<div>` element

and apply the class `.embed-responsive`, and an aspect ratio class such as `.embed-responsive-16by9`.

You can optionally apply an explicit descendant class `.embed-responsive-item` on the embedded element to match the styling for other attributes. Here's is an example:

Example	Try this code »

```html
<!-- 21:9 aspect ratio -->
<div class="embed-responsive embed-responsive-21by9">
    <iframe class="embed-responsive-item" src="//www.youtube.com/embed/YE7VzlLtp-4">
</iframe>
</div>

<!-- 16:9 aspect ratio -->
<div class="embed-responsive embed-responsive-16by9">
    <iframe class="embed-responsive-item" src="//www.youtube.com/embed/YE7VzlLtp-4">
</iframe>
</div>

<!-- 4:3 aspect ratio -->
<div class="embed-responsive embed-responsive-4by3">
    <iframe class="embed-responsive-item" src="//www.youtube.com/embed/YE7VzlLtp-4">
</iframe>
</div>

<!-- 1:1 aspect ratio -->
<div class="embed-responsive embed-responsive-1by1">
    <iframe class="embed-responsive-item" src="//www.youtube.com/embed/YE7VzlLtp-4">
</iframe>
</div>
```

In the above example, we've created the 4 responsive videos with 4 different aspect ratios (**21:9**, **16:9**, **4:3**, and **1:1**) by using the classes `.embed-responsive-21by9`, `.embed-responsive-16by9`, `.embed-responsive-4by3`, and `.embed-responsive-1by1`, respectively.

Tip: The aspect ratio of an image describes the proportional relationship between its width and its height. Two common videographic aspect ratios are 16:9 and 4:3.

Horizontal Alignment of Images

You can align the images left or right within a larger box using the `.pull-left` or `.pull-right` classes. You can also use the text alignment classes such as `.text-left`, `.text-right`, and `.text-center` on the parent container to align the inline images horizontally left, right, and center.

However, to center align the block-level images you need to use the `.mx-auto` margin utility class.

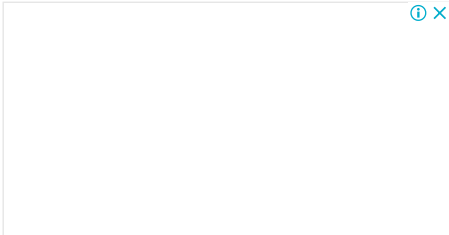Let's take a look at the following example to see how it works:

Example	Try this code »

```html
<!-- Horizontal alignment using float classes -->
<div class="box clearfix">
    <img src="images/avatar.svg" class="pull-left" alt="Sample Image">
    <img src="images/avatar.svg" class="pull-right" alt="Sample Image">
</div>

<!-- Left alignment using text alignment classes -->
<div class="box text-left">
    <img src="images/avatar.svg" alt="Sample Image">
</div>
```

```
12    <!-- Right alignment using text alignment classes -->
13    <div class="box text-right">
14        <img src="images/avatar.svg" alt="Sample Image">
15    </div>
16
17    <!-- Center alignment using text alignment classes -->
18    <div class="box text-center">
19        <img src="images/avatar.svg" alt="Sample Image">
20    </div>
21
22    <!-- Center alignment of block-level image using auto margin -->
23    <div class="box">
24        <img src="images/avatar.svg" class="d-block mx-auto" alt="Sample Image">
25    </div>
```
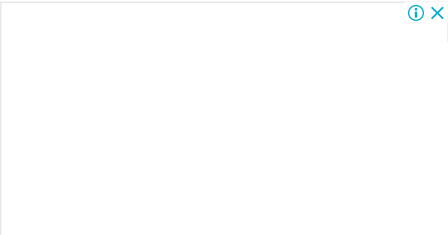
« PREVIOUS PAGE   NEXT PAGE »

Is this website helpful to you? Please give us a like, or share your feedback to help us improve. Connect with us on Facebook and Twitter for

ABOUT US

Our Story

Terms of Use

Privacy Policy

CONTACT

Contact Us

Report Error

Advertise

INTERACTIVE TOOLS

Title & Meta Length Calculator   HTML Formatter

Bootstrap Button Generator   HTML Color Picker

SQL Playground   HTML Editor