

12 – Colors and Image Manipulation using Bootstrap 4

Colors

Bootstrap has a number of colors that you can access through contextual color names. They're used everywhere, including in buttons, background colors, as well as texts, so you'll hear some of these names in different places, but let's take a look at the basic options available in texts, links, and backgrounds. So the way that these work is by using the prefix text, and then using one of the contextual color names.

The classes for text colors are: **text-muted**, **text-primary**, **text-success**, **text-info**, **text-warning**, and **text-danger**

html

```
<p class="text-muted">The text is muted</p>
<p class="text-primary">The text is important</p>
<p class="text-success">The text indicates success</p>
<p class="text-info">The text represents some information</p>
<p class="text-warning">The text represents a warning</p>
<p class="text-danger">The text represents danger</p>
```

The text is muted

The text is important

The text indicates success

The text represents some information

The text represents a warning

The text represents danger

The classes for background colors are: **bg-primary**, **bg-success**, **bg-info**, **bg-warning**, **bg-danger**

html

```
<p class="bg-primary">The text with primary color background</p>
<p class="bg-success">The text with success color background</p>
<p class="bg-info">The text with info color background</p>
<p class="bg-warning">The text with warning color background</p>
<p class="bg-danger">The text with danger color background</p>
```

The text with primary color background

The text with success color background

The text with info color background

The text with warning color background

The text with danger color background

Working with images

Bootstrap also has some very useful classes that help you work with images, as well as with figures.

Border images

The **rounded** class adds rounded corners to an image. You can also apply direction of which edge the image do you want it to be rounded.

html

```

```



The **rounded-circle** → class shapes the image to a circle

html

```

```



The **img-thumbnail** → class shapes the image to a thumbnail (bordered)

html

```

```



Aligning images

Float an image to the right with the **float-right** class or to the left with **float-left**

Center an image by adding the utility classes **mx-auto** (margin:auto) and **d-block** (display:block) to the image.

Responsive images

Responsive images automatically adjust to fit the size of the screen.

- **img-fluid:** Create responsive images by adding an **img-fluid** class to the **** tag. The image will then scale nicely to the parent element.

The **img-fluid** class applies **max-width: 100%;** and **height: auto;**

- **figure-img** class makes the text with a smaller font and lighter color. The **<figure>** tag in HTML specifies self-contained content, like illustrations, diagrams, photos, code listings, etc.

While the content of the **<figure>** element is related to the main flow, its position is independent of the main flow, and if removed it should not affect the flow of the document.

- **figure-caption** class makes the text in **<figcaption>** with a smaller font and lighter color. The HTML **<figcaption>** element represents a caption or legend for the rest of the contents its parent figure element, if any.

```



<figure class="figure-img">
  
  
  
  <figcaption class="figure-caption">New York City</figcaption>
</figure>

```

CSS variables

Bootstrap has some features that allows you to work with a new feature of CSS called CSS Variables. Now CCS Variables are a newer feature within CSS that allows you to store a set of property values that you can reuse within your CSS. Now this version of Bootstrap provides pre-written CSS variables that can be used in developing your projects and customizing your colors. CSS Variables are so new that browser support is an issue. They're not supported in any version of Internet Explorer, and only on very new versions of the other browsers.

Syntax

```

var(custom-name, value)
<style>
  :root{
    --hello:#25FF64;
  }
</style>
<h1 style="color:var(--hello);">Our Commitment</h1>

```