Images

CSS3

Image Element Size Considerations

By default an image will take up its natural size based on the pixel dimensions of the image. In Windows, you can right-click an image file on your computer, click “Properties,” and look in the “Details” tab to discover its width and height in pixels. On Macintosh, the process would be similar but you would click “Get Info.”

The next things the browser checks before sizing an image in the web page are the width and height attributes of the image. If there are width and height values assigned on the element, these will override its default sizing.

Finally, we can interject with CSS, which supersedes those other two measures. It is important to consider the following:

- The larger the file size, the longer it will take for the user’s web browser to download.
- Using a width and height greater than the natural size will stretch the image, which will appear as pixelation.
- To avoid vertical or horizontal stretching and deformation, it is important to maintain the image’s aspect ratio (this is the relation between width and height—how many pixels high per pixels wide.)

Basic (Responsive) Image Sizing

Now that we have some requisite knowledge down, let's consider one of the most common image-sizing techniques used across the web. Consider the following CSS snippet:

```
img {
    width: auto; /* This defaults to the natural width of the image. */
    max-width: 100%; /* Ensures the width of the image never exceeds the available space of its parent. By default it would have overflown. */
    height: auto; /* A height of auto will keep the height in sync with the width, as per the image's natural aspect ratio. */
}
```

The above is just a suggestion and is suitable for most standard web articles.

For other popular image stylings, see the CSS3 Images article from W3Schools. They cover techniques for everything from rounded corners to image filters.

Aligning Images in Text

If, for example, you have an image inline with some text in a paragraph, you can use float to position it within the text either to the left or to the right. See the following...

img { float: none; }

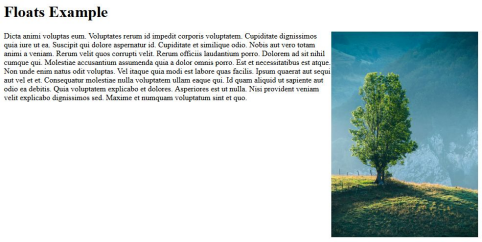
By default, the image occupies space in the paragraph as if it were a word.

img { float: left; }

Floating left will position the image to the left of the text, much like in a magazine!

img { float: right; }

Floating right will position the image to the right of the text.



Background Images

We've had a look at a few styling considerations for the img HTML element, but there is a way we can display images in a page without outright including one within the HTML code.

For most any element in the page we can add a background-image, which will appear behind any content or text within the element. This image occupies the same space that a background-color might. Note that background-color appears behind any images, and images featuring transparency will show the background colour in transparent spots.

Let's have a look at how we can use a background-image, and what we can manipulate once we've applied one to an element.

Setting a Background-Image

To set a background image, you must specify a URL as the background-image value. In your CSS, this will look like so…

```
.test-background {
    background-image: url( “../img/tree.jpg” );
}
```

Careful with the path—ensure it is correct relative to your current file. It can help to have your Web Console open when testing background images as it can let you know if the image file was found, or not, at the provided URL.

Depending on the size of the background, it may be apparent that it is repeating like a tile all the way down the page.

Notice too that the image retains its natural dimensions by default! This can be adjusted by use of more background-image properties.

Background Repeat

By default an image will repeat both horizontally and vertically to fill the space it is offered. This behaviour can be adjusted via the background-repeat property…

- repeat (default, repeats on both axes)
- repeat-x (repeats only horizontally)
- repeat-y (repeats only vertically)
- no-repeat (only appears once in the background)
- space (repeats on both axes, leaving space between each image)
- round (centres the image, repeating 50% of a copy in each direction to the edges of the available area)

The most common of these are no-repeat when you just want a single image in the background, or repeat if you have an image that tiles nicely.

Background-Size

We can adjust the size of the image using background-size. There are a few different values and formats you can assign here…

- 2 unit-specified values separated by space (first is width, second is height)
- 1 unit-specified value (will apply to both the width and height)
- contain (the image's size will be adjusted so that the image matches the width or height of the element with no overflow)
- cover (the image's size will be adjusted so that the entire visible space is covered by it, even if it results in overflow)

To clarify on the first bullet, check out the following example:

```
p {
   background-image: url( "../img/tree.jpg" );
   background-size: 150px 350px;
}
```

Background Position

Is the image not quite where you'd like it, even though the size is just right? CSS affords us the background-position property for adjusting where the image will display to us.

There are a variety of ways and syntaxes to use in positioning the background image, but some will take a bit of practice. To get the basics, it is easiest to start with the keywords and basic coordinates.

Keywords can be used alone, or in pairs, to achieve the desired effect:

- top
- right
- bottom
- left
- center

An alternative is to provide one or two unit-specified values.

1. A single value will be the distance to the right and downward that the image should appear.
2. Two values represent, respectively, the distance to the right, and the distance downward. Use negative values to move the image in the opposite direction(s).

Additional Reading

If you read enough into CSS techniques, you're likely to eventually come across the term "sprites" in regards to background images. This is the practice of having the browser load one large background image at once (instead of many smaller ones separately.)

Once the image is loaded, you can use a combination of background-size and background-position to show small pieces of that large image. These pieces, if coherent, would be referred to as sprites in a larger sprite sheet.

Learn more in CSS Tricks' CSS Sprites article.

Recommended Reading

Look into more image-related content here:

- Meyer, E. A; Weyl, E. (October 2017). CSS: The Definitive Guide, 4th Edition. O'Reilly Media, Inc.
  - Chapter 3. Values and Units
    - Images
  - Chapter 9. Colors, Backgrounds, and Gradients
    - Clipping the Background
    - Background Images
    - Background Positioning
    - Changing the Positioning Box
    - Background Repeating (or Lack Thereof)
    - Getting Attached
    - Sizing Background Images
    - Bringing It All Together
    - Multiple Backgrounds