

Floats

Lesson Time: 30 Minutes

Float is used for pushing images and other block elements to the left or right of their **parents**.

In this code, we have a paragraph element, with a child `` element inside of it.

```
<p>

Lorem ipsum dolor sit amet,
audiam definitionem sed in, decore primis eleifend cum ne.
Te est option perpetua similique. An ius euismod accumsan, alia purto neglegent
pro ad.

<img class='leftimg' src='https://media.istockphoto.com/photos/coconut-palm-

Utu in scaevola convenire reformidans, eu vim deterruisset conclusionemque.
An nobis suscipit constituto sea, ne omnesque suavitate est. Ne duo saepe meli
mei mundi ludus facete ut.

</p>
```

Without Float the image just appears static inside the paragraph.

Lorem ipsum dolor sit amet, audiam definitionem sed in, decore primis eleifend cum ne. Te est option perpetua similique. An ius euismod accumsan, alia purto neglegentur pro ad.



Ne duo saepe melius quaeque, mei mundi ludus facete ut.

Utu in scaevola convenire reformidans, eu vim deterruisset conclusionemque. An nobis suscipit constituto sea, ne omnesque suavitate est.

With Float, it nicely “floats” to the left of the parent.



Lorem ipsum dolor sit amet, audiam definitionem sed in, decore primis eleifend cum ne. Te est option perpetua similique. An ius euismod accumsan, alia purto neglegentur pro ad. Usu in scaevola convenire reformidans, eu vim deternuisset conclusionemque. An nobis suscipit constituto sea, ne omnesque suavitatem est. Ne duo saepe melius quaeque, mei mundi ludus facere ut.

The code is as simple as this.

```
.leftimg{  
  float:left;  
}
```

Key Terms	float
Lesson Files	
Additional Resources	https://www.quackit.com/css/tutorial/css_float.cfm
Further Learning	

Advanced CSS

Lesson Time: 60 Minutes

Media Queries

Media Queries allow you to apply a CSS rule ONLY when the browser screen is a certain size. We use Media Queries to switch to a different set of rules on smaller screens.

```
@media (min-width: 576px) {  
  .container {  
    width: 540px;  
  }  
}
```

Media Queries use the syntax `@media (condition to apply) {standard css code}`

In the example above, the CSS rule only applies when the browser window is a minimum width of 576px. If less, this CSS code will NOT apply. We use media queries to control responsive design, and they should be a part of all your CSS going forward.

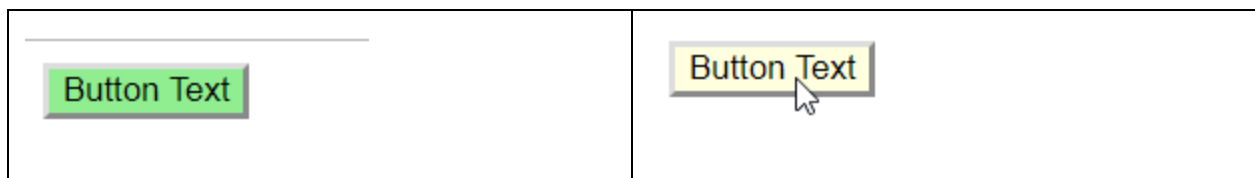
In order for media queries to work properly, we need to include this block of code in the `<head>` section of the HTML. It will never change. This tells the HTML to set all content to the size of the browser window, and set the initial scale to 1.

```
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

:Hover

Hover is one of the pseudo-classes in CSS. We can use :hover to create a hover effect on an element without using Javascript.

```
9      <style>
10
11      .button{
12      background-color: lightgreen;
13      }
14
15      .button:hover{
16      background-color: lightyellow;
17      }
18
19      </style>
20
21  </head>
22
23  <body>
24      <input class="button" type="button" name="example" value="Button Text">
25  </body>
26  </html>
```



:Focus

Focus takes effect when an input element has focus

When the user clicks the textbox, the textbox gains focus. CSS paints a red border around the textbox in this example.

<pre><input class= button type= button name= example value <input class="tbx" type="textbox" name="textinput"> </body></pre>	<pre>.tbx:focus{ border: 4px solid red; }</pre>
	

:nth-child(n)

This pseudo-element is often used in tables to select odd/even rows or to select a certain element.

<pre><table> <tr>... <td>Row 1</td> <td>SomeValue</td> <td>SomeValue</td> </tr> <tr>... <td>Row 2</td> <td>SomeValue</td> <td>SomeValue</td> </tr></pre>	<pre>tr:nth-child(even){ background-color:lightblue; } table{ border: 2px solid black; margin: 10px; }</pre>
---	--

Row 1	SomeValue	SomeValue
Row 2	SomeValue	SomeValue
Row 3	SomeValue	SomeValue
Row 4	SomeValue	SomeValue

RGBA Color

RGBA color is just like RGB color, except it adds an Alpha Transparency. Elements can now be semi-transparent on the page. We use RGBA with four parameters RGBA(red, green, blue, alpha). The Alpha level is a value between 0 (completely transparent) and 1 (completely opaque).

```
h1{
  color: white;
  background-color: rgba(0, 0, 0, 0.5);
  padding: 10px;
```

In this example, we are using black and making it 50% transparent.

The result is that we see partly through the black onto the color below.

Box Model Exercise 1

Key Terms	:Hover, Media Queries, RGBA
Lesson Files	

Additional Resources	@media Tutorial https://www.quackit.com/css/at-rules/css_media_at-rule.cfm Hover https://www.w3schools.com/cssref/sel_hover.asp Pseudo Classes https://www.w3schools.com/css/css_pseudo_classes.asp
Further Learning	