

How do I... Center a Web page layout using CSS?

Date: January 2nd, 2008

Author: John Lee

Category: How do I, Programming

Tags: Web, Pennsylvania, CSS, Nation, Layout, Web Page, Continent, Liberty, Let, John Lee

The most common LCD monitors on the market today follow the wide-screen format. That wide ratio can play havoc with Web site designers and their layout of Web pages. However, with a little bit of Cascading Style Sheet (CSS) code, you can control your Web page design by centering it on any screen despite its horizontal to vertical ratio.

This blog post is also available in PDF format as a TechRepublic download, which includes the sample code.



The situation

The next time you step into the local Best Buy store, take a look at their selection of computer monitors. You'll notice that as the screen ratio of monitors has shifted towards a more movie-theater like scale (rectangular) and away from a television ratio of 4:3 (square), the bigger a monitor gets, the wider it gets.

That's great for watching movies, playing games or running palette-heavy programs such as Photoshop, but what about surfing the Web? Long lines of text are hard and tiring on the eye, which is why you see so many Web sites that center their layouts. This allows the authors to set a fixed width for their content and present it in a dynamic fashion as the user's browser window expands and contracts in width.

You could, of course, create a fixed-width table and center it on the page, but today's Web designers and developers are trying to move away from tables altogether and instead control layout entirely with CSS markup. The small bit of CSS code that I am about to show you will get you started with using CSS to create your Web page layouts.

Open your preferred HTML editor and create a new page. Enter the code below (**Listing A**) to get started.

Listing A

```
<html>
<head>
<title>Centering a page layout with CSS</title>
</head>
<body>
</body>
</html>
```

Let's go ahead and place some formatted text in the Body of the HTML document (**Listing B**), so we can see the effects of our CSS code as we build it.

Listing B

```
<html>
<head>
<title>Centering a page layout with CSS</title>
</head>
<body>
<h1>The Gettysburg Address</h1>
<h2>Gettysburg, Pennsylvania<br>
November 19, 1863</h2>
Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty, and c
</body>
</html>
```

Save the HTML document, and then preview it in your browser. Play with the width of the browser window to see how the copy expands and shrinks to match the size of the viewing area.

Let's start building the CSS code and tame all of that copy. Let's keep this simple and create your CSS code in the head of this document (**Listing C**) instead of linking to an external style sheet. Place it right immediately after the closing </TITLE> tag.

Listing C

```
<html>
```

```
<head>
<title>Centering a page layout with CSS</title>
<style type="text/css">
<!--
-->
</style>
</head>
<body>
<h1>The Gettysburg Address</h1>
<h2>Gettysburg, Pennsylvania<br>
November 19, 1863</h2>
Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty, and c
</body>
</html>
```

Most Web Designers, when declaring a CSS class that is going to be used as a centering element, give the class the name "wrapper" or "container." I prefer "wrapper," so that's what we'll use here (**Listing D**).

Listing D

```
<html>
<head>
<title>Centering a page layout with CSS</title>
<style type="text/css">
<!--
#wrapper
-->
</style>
</head>
<body>
<h1>The Gettysburg Address</h1>
<h2>Gettysburg, Pennsylvania<br>
November 19, 1863</h2>
Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty, and c
</body>
</html>
```

Next, we'll set the width of the wrapper to 720 pixels (**Listing E**), so that it will sit nicely on an 800 x 600 monitor. Of course, you could make the width as long or as short as you like to accommodate your design.

Listing E

```
<html>
<head>
<title>Centering a page layout with CSS</title>
<style type="text/css">
<!--
#wrapper {
width: 720px;
```

```
-->
</style>
</head>
<body>
<h1>The Gettysburg Address</h1>
<h2>Gettysburg, Pennsylvania<br>
November 19, 1863</h2>
Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty, and c
</body>
</html>
```

That sets the width of our content, but how do we center it? Simple — the margin attribute shown in **Listing F**, with a setting of zero (0) and automatic alignment, places all of our content in the center of the screen:

Listing F

```
<html>
<head>
<title>Centering a page layout with CSS</title>
<style type="text/css">
<!--
#wrapper {
    width: 720px;
    margin: 0 auto;
}
-->
</style>
</head>
<body>
<h1>The Gettysburg Address</h1>
<h2>Gettysburg, Pennsylvania<br>
November 19, 1863</h2>
Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty, and c
</body>
</html>
```

Believe it or not, that small amount of code does the job... in most modern browsers. Older versions of IE and Netscape, however, have issues when complying with this CSS markup, and will center-align the text, when what we really want is the text to be left-aligned, but stay put in the center of the window. Don't worry though; there is a simple fix for these outdated browsers.

Start by defining a text-align attribute for the <BODY> tag of the HTML document, as shown in **Listing G**. You must place it before the wrapper style, or this will not work, due to the Cascading nature of Cascading Style Sheets!

Listing G

```
<html>
<head>
<title>Centering a page layout with CSS</title>
<style type="text/css">
<!--
```

```
body {
text-align: center;
}
#wrapper {
width: 720px;
margin: 0 auto;
}
-->
</style>
</head>
<body>
<h1>The Gettysburg Address</h1>
<h2>Gettysburg, Pennsylvania<br>
November 19, 1863</h2>
Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty, and c
</body>
</html>
```

Then add the text-align attribute below to the wrapper class and give it a value of left as in **Listing H**.

Listing H

```
<html>
<head>
<title>Centering a page layout with CSS</title>
<style type="text/css">
<!--
body {
text-align: center;
}
#wrapper {
width: 720px;
margin: 0 auto;
text-align: left;
}
-->
</style>
</head>
<body>
<h1>The Gettysburg Address</h1>
<h2>Gettysburg, Pennsylvania<br>
November 19, 1863</h2>
Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty, and c
</body>
</html>
```

There is one more line of code that needs to be written before we can test our work. Our wrapper will break in Netscape 6 if the browser window is shorter than the width of the wrapper. A min-width attribute (**Listing I**) in the body style will take care of that.

Listing I

```
<html>
<head>
<title>Centering a page layout with CSS</title>
<style type="text/css">
<!--
body {
text-align: center;
min-width: 760px;
}
#wrapper {
width: 720px;
margin: 0 auto;
text-align: left;
}
-->
</style>
</head>
<body>
<h1>The Gettysburg Address</h1>
<h2>Gettysburg, Pennsylvania<br>
November 19, 1863</h2>
Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty, and c
</body>
</html>
```

Finally, apply the wrapper to the content of the HTML document by placing a <DIV> tag around all of the Gettysburg Address copy as shown in **Listing J**.

Listing J

```
<html>
<head>
<title>Centering a page layout with CSS</title>
<style type="text/css">
<!--
body {
text-align: center;
min-width: 760px;
}
#wrapper {
width: 720px;
margin: 0 auto;
text-align: left;
```

```
}
-->
</style>
</head>
<body>
<div id="wrapper">
<h1>The Gettysburg Address</h1>
    <h2>Gettysburg, Pennsylvania<br>
    November 19, 1863</h2>
    Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty, and c
</div>
</body>
</html>
```

Preview the page in several browsers to see how the left-justified text stays put in the center of the browser no matter how small or narrow the browser window is.

Although we used no tables in this example, you could place whatever content you wish within the <DIV> tag — tables included — and the CSS structure would keep your layout in place in the center.

John Lee is a consultant specializing in design and illustration and a freelance technical writer. You can visit his Web site at johnleestudio.com.

Print/View all Posts

Comments on this blog

Trackbacks

The URI to TrackBack this entry is: <http://blogs.techrepublic.com.com/howdoi/?p=186>

No trackbacks yet.

Copyright © 2007 CNET Networks, Inc. All Rights Reserved.