**1. Setup**

Visit the [What Shoul0d We Call Me](https://s3.amazonaws.com/one-month-html/whatshouldwecall/whatshouldwecall.html) blog to follow along.

**2. Objective**

Using the Chrome Web Inspector, try to find the answers to each of the following questions. You have 6 minutes. GO!

1. What color is the date link on this site "5th Jun 2012"?
2. What heading tag (ex. h1, h2, h3, etc) is being used for the title of each blog post?
3. What is the width and height of the main GIF?
4. What is the width of the sidebar?
5. What CSS declaration would you add to the web inspector to make all the images disappear?
6. color: #C6C6C6; ♫
7. h2
8. max-width: 500px
9. 332 x 96 px
10. What CSS declaration would you add to the web inspector to make all the images disappear:

"display: none;" or "visibility: hidden;

**Scavenger hunt answers:**

1. #C6C6C6
2. h2
3. 446 x 252
4. 332px
5. display:none *removes images*

**Important alternatives**

1. visibility: hidden *space available, just transparent.*
2. To hack a link off the page
3. When to add !important;  display: none !important;

In this lesson, we start a new project—we’re going to make a cat blog (meow). By working through the project, you will learn how to lay out a page's elements with HTML and CSS. Our website will have a common layout that you might have noticed on other websites. It will have a header, footer, main section divided into columns and a sidebar.

### 1. Download the template

[Click here to download](https://s3.amazonaws.com/one-month-rails-production/assets/files/000/004/396/original/column-layout.zip?1496101204) the blank template and follow along with the video above.

### 2. Create a basic template using the <div> tag

<div id="container">

</div>

### 3. Build out the layout of your site

<div id="container">

<div id="header">

<h1>Column Layout Page</h1>

</div>

<div id="main">

<h2>Heading</h2>

</div>

<div id="footer">

</div>

</div><!-- #container -->

### 4. Add content to the main

<div id="main">

<h2>Heading</h2>

<img src="images/kitten.jpg" alt="kitten">

...

### 5. Add a paragraph

In Sublime text, type "lorem" and press the TAB on your keyboard. BOOM! It should generate this automatically:

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod

tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,

quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo

consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse

cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non

proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

### 6. Create your second blog post

<h2>Blog Post #2</h2>

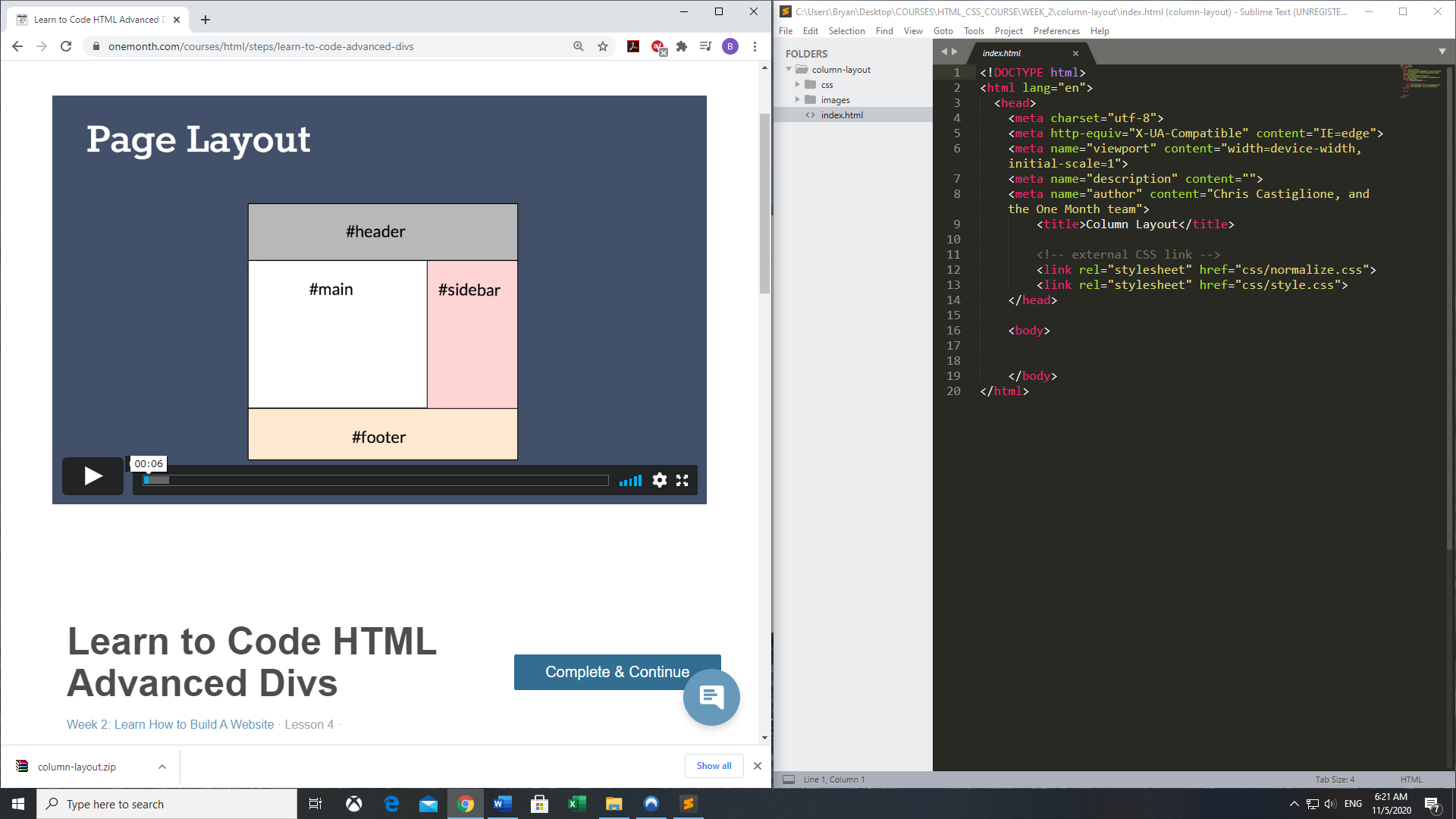
### 7. Write some footer copy

<div id="footer">

<p>Created by <a href="http://www.twitter.com/castig"

target="\_blank">@castig</a> for <a href="http://www.onemonthhtml.com" target="\_blank">OneMonthHTML</a></p>

</div><!-- #footer -->



Let’s style the different sections of the blog webpage. In this lesson, I offer some tips on using HTML and CSS comments to keep your code organized and easily readable .

### 1. Layout out your CSS selectors

...

#container {

}

#header {

}

#main {

}

#footer {

}

### 2. Hook up your sections with some color

...

#header {

background: Silver;

}

#main {

background: SkyBlue;

}

#footer {

background: PeachPuff;

}

### 3. Give sections shape

...

#container {

width: 1000px;

margin: 0 auto 0 auto;

}

...

### 4. Add padding to your layout

...

#header {

background: Silver;

padding: 50px;

}

#main {

background: SkyBlue;

padding: 50px;

}

#footer{

background: PeachPuff;

padding: 50px;

}

### 5. Add the sidebar to your HTML

...

<div id="sidebar">

<h3>Sidebar</h3>

<ul>

<li>August</li>

<li>July</li>

<li>June</li>

<li>May</li>

</ul>

</div>

### 6. Add a closing sidebar comment

Why do we do this? Because it's helpful to name your closing </div>.

Think of it this way...the larger your site gets, the more </div> tags you'll add. At some point, you may want to delete sections from your layout. Without comments, it will be difficult to remember which two <div></div> tags go together. Commenting on the closing </div> makes this much clearer.

</div><!-- #sidebar -->

**1. Commit to memory these two things**

1. IDs: You cannot reuse an ID name. Only one per page!
2. .classes: You CAN have multiple class names per page

**2. Use classes to add color to headers**

1. IDs begin with the # symbol
2. Classes begin with a dot

HTML

...

<h2 class="blue">Heading</h2>

...

<h2 class="red">Blog Post #2</h2>

...

CSS:

...

.blue {

color: blue;

}

.red {

color:red;

}

...

**3. Add alignment to images**

/css/styles.css

...

.alignleft {

float: left;

}

.alignright {

float: right;

}

/index.html

...

<img src="images/kitten.jpg" alt="kitten" class="alignright">

...

**4. Add margin to aligned images**

.alignleft {

float: left;

margin: 0 15px 15px 0;

}

.alignright {

float: right;

margin: 0 0 15px 15px;

}

**5. Tip of the Day: Understand Margin and Padding Values**

Remember: It's like a clock. As in, it goes clockwise: top, right, bottom and left

margin: 10px 20px 15px 5px;

/\*

10px = top

20px = right

15px = bottom

5px = left

\*/

# Inline vs. Block

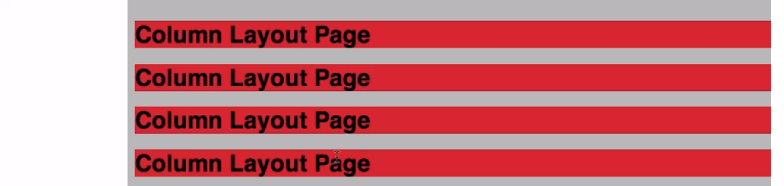
[Week 2: Learn How to Build A Website](https://onemonth.com/courses/html#learn-how-to-build-website)· Lesson 9 ·

* [Lesson Notes](https://onemonth.com/courses/html/steps/css-inline-vs-block#step-notes)
* [Discussion (1)](https://onemonth.com/courses/html/steps/css-inline-vs-block#discussion)

What’s the difference between inline and block elements in HTML? In short, block elements won’t allow any other HTML elements to appear either to the left or right, while inline elements allow text, or other HTML elements to pile up on either side. Let me show you a few examples of inline vs. block.

### 1. Understanding Block

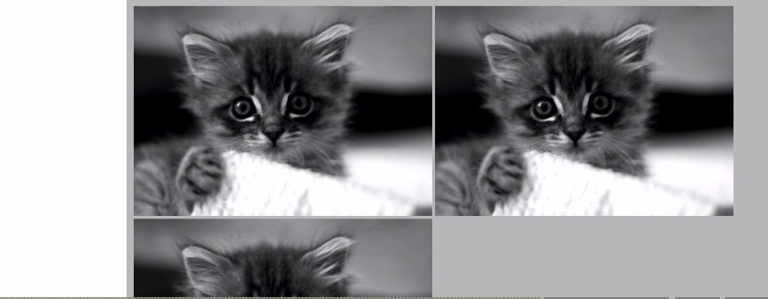
Block elements take up the full width...



Examples of block elements: <h1>, <h2>, <div>

### 2. Understanding Inline

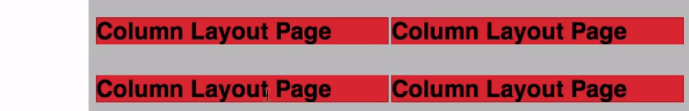
Inline elements fall side by side....



Examples of inline elements: <a>, <img>, <span>

### 3. Inline-Block!?!?! (don't stress about understanding this one right now)

Inline-block elements maintain their width (like block elements), but fall side by side (like inline elements). Just like this...



Try it yourself:

h1 {  
  
  display: inline-block;  width: 400px;  
  
}

### 4. Don't stress...

1. Again, the inline-block concept is somewhat advanced.
2. You probably won't absorb all of this right now, but it's good to be aware of for the future.

[Dark logo @2x](https://onemonth.com/)



# Extra Credit: Getting a Deeper Understanding

[Week 2: Learn How to Build A Website](https://onemonth.com/courses/html#learn-how-to-build-website)· Lesson 10 ·

* [Lesson Notes](https://onemonth.com/courses/html/steps/page-layout-going-deeper#step-notes)
* [Discussion (2)](https://onemonth.com/courses/html/steps/page-layout-going-deeper#discussion)

In this lesson we’ll go deeper into HTML and CSS, we’ll look at default styles in the browser (and how to override them), and we’ll also look at how to achieve consistency across the different browsers like Chrome, Safari, and Microsoft Edge.

### 1. Topics covered in this video

1. normalization
2. IE conditionals
3. browser consistency
4. box-model hack

### 2. Additional resources

1. Normalize: <http://necolas.github.io/normalize.css/>
2. Internet Explorer Conditionals: <http://www.quirksmode.org/css/condcom.html>