CSS #3 – Sectioning

## Semantics Overview

What are *semantics*? [Semantics within HTML](http://boagworld.com/dev/semantic-code-what-why-how/) is the practice of giving content on the page meaning and structure by using the proper element. Semantic code describes the value of content on a page, regardless of the style or appearance of that content. There are several benefits to using semantic elements, including enabling computers, screen readers, search engines, and other devices to adequately read and understand the content on a web page. Additionally, semantic HTML is easier to manage and work with, as it shows clearly what each piece of content is about.

Let’s look at two elements—<div>s and <span>s—that actually don’t hold any semantic value. They exist for styling purposes only.

## Identifying Divisions & Spans

Divisions, or <div>s, and <span>s are HTML elements that act as containers solely for styling purposes. As generic containers, they do not come with any overarching meaning or semantic value. Paragraphs are semantic in that content wrapped within a<p> element is known and understood as a paragraph. <div>s and <span>s do not hold any such meaning and are simply containers.

**Building Structure**

For the longest time the structure of a web page was built using divisions. The problem was that divisions provide no semantic value, and it was fairly difficult to determine the intention of these divisions. Fortunately HTML5 introduced new [structurally based elements](http://dev.opera.com/articles/new-structural-elements-in-html5/), including the <header>, <nav>, <article>, <section>, <aside>, and <footer> elements.

All of these new elements are intended to give meaning to the organization of our pages and improve our structural semantics. They are all block-level elements and do not have any implied position or style. Additionally, all of these elements may be used multiple times per page, so long as each use reflects the proper semantic meaning.

**ASSIGNMENT**

**Experiment with HTML5 structurally based elements**

1. Read through this webpage: <http://learn.shayhowe.com/html-css/getting-to-know-html/>

and this webpage: <http://htmldog.com/guides/html/intermediate/sectioning/>

1. Copy/paste the code below into Sublime Text, save it as **london.html**, and study it.

<!DOCTYPE html>

<html>

<head>

<style>

header {

background-color:black;

color:white;

text-align:center;

padding:5px;

}

nav {

line-height:30px;

background-color: #eeeeee;

height:300px;

width:100px;

float: right;

padding:5px;

}

section {

width:350px;

float:left;

padding:10px;

}

footer {

background-color:black;

color:white;

clear:both;

text-align:center;

padding:5px;

}

</style>

</head>

<body>

<header>

<h1>City Gallery</h1>

</header>

<nav>

London<br>

2nd City <br>

3rd City

</nav>

<section>

<h1>London</h1>

<p>London is the capital city of England. It is the most populous city in the United Kingdom,

with a metropolitan area of over 13 million inhabitants.</p>

</section>

<footer>

© Mr. Singer 2016

</footer>

</body>

</html>

1. Customize the above code:
   1. Increase section padding to 30px.
   2. Float the Navigation bar on the left instead of the right.
   3. Change Footer and Header background colors.
   4. Fix the Copyright logo in the Footer.
   5. Create a second paragraph inside <section> and write another couple sentences about London.
   6. Insert a picture into a new HTML element called <aside>, that relates to London.
      1. Make sure it fits appropriately. *In other words, mind the picture’s width. Also, the top of the picture needs padding so it doesn’t sit directly next to the header. To accomplish this you should create a new CSS rule for <img>.*
   7. Make additional customizations to this webpage, past what I’ve asked you.
   8. Think of two additional cities, anywhere in the world.
   9. Create two more webpages that look exactly like london.html (I’d suggest you just “save as”, but name these files by the city names you chose.
   10. Enter a couple sentences into each section on each city page.
   11. Insert a picture into each webpage’s <aside> that relates to each city, and make sure it fits appropriately.
   12. Create a folder named LastFirstCSS3Per# and place all three webpages inside of it *(do not put # in your filename or it won’t work).*
   13. Make sure the links in your Navigation Bar work on each page. That is, make sure each city link delivers you to the appropriate page, including itself!

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **RUBRIC** | **Developing (13 pts.)** | **Emerging (15 pts.)** | **Proficient (17 pts.)** | **Advanced**  **(20 pts.)** |
| **CSS #3 (Sectioning)** | You completed **little** of the required work. | You completed **most** of the required work. | You completed **all** required work.  *Or you completed most required work and customized further.* | You completed **all** required work **and customized further!** |