html tags

Two types: semantic and div

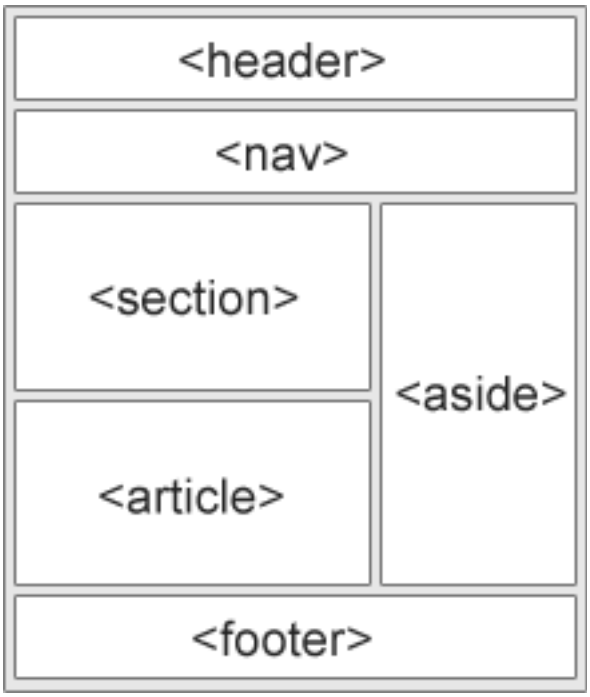
Sister A

2017

Semantic vs s

What's the Difference

When setting up a page in html, it is important to break the content up into smaller sections so that when we add css (styling) to the page, the sections can be styled or controlled. A typical web page might look something like this:



Semantic elements are tags that should always be used first when sectioning a page in html… if there is not a semantic tag… then use a div tag. Semantic tags are elements with meaning… in other words, they describe what the tag is there for such as a p tag (paragraph) or a h nav tag for navigation. Here is a list of some semantic elements we will be using:

<nav>	navigation
<p>	paragraph
<h>	headline
<section>	section
<header>	top of document

<main>	main part of document
<footer>	bottom of document
<head>	top of html document
<html>	tells browser it is an html document
<article>	a self contained content such as blog post, newspaper article
<img>	signifies an image
<a href>	signifies a link
<figure>	to group an image and caption together
<figcaption>	caption for an image in a <figure> section
<aside>	defines content aside from the page content
<details>	defines additional details the user can view or hide
<mark>	defines marked/highlighted text
<summary>	defines visible heading for a <details> element
<time>	defines a date or time
	list item
	unordered list
	ordered list
<body>	body of content
	happening for a certain distance or time

Because there might be many other items that need to be “sectioned” so they can be styled with css that do not fit into one of these semantic names, they came up with the .

A tag is classified as an HTML element, and it is, but it is used in conjunction with other elements from your stylesheet. In its basic form, the **div** tag does nothing to a page’s appearance. If you put this in your HTML code:

nothing will happen that you can see.

A doesn't convey any meaning about its contents (unlike a **p** element that signifies a paragraph, or an **h1** or **h2** element that would indicate a level 1 or level 2 heading).

This makes it easy to customize to your needs.

The **div** element is currently the most common method for identifying the structural sections of a document and for laying out a web page using CSS.

The **div** element is used to group segments of content into logical divisions. The element itself is semantically neutral, but not entirely meaningless. A div simply says "these things belong together, and are separate from those other things." The div element is a content organization tool, not a page layout tool.

The **div** elements are like bookends: what's important is what goes between them.

The **div** is an 'anything-goes' element – it can contain any inline or block-level elements you choose, so it has no typical content.

Here are the main attributes you can place inside a **div** element:

class: specifies a "classname" for an element

id: specifies a unique "id" for an element

style: specifies an inline style for an element

title: specifies extra information about an element

There are others, but these are the ones you will work with most frequently. Let's take a quick look at the two you will be using class and id. **ALWAYS** use with a class or id: **NEVER just use **

The **class** selector is a CSS selector that serves to identify an element, or group of elements. By identifying something as a *class,* you give yourself the opportunity to style it with multiple CSS elements, without styling anything else.

The **id** selector (think I.D. as in "identification," not "id" as in Freud) is a CSS selector similar to the *class* selector. It, too lets you identify an element, or group of elements, and style them. The biggest differences between them is that you can only use an *id* once in a page, whereas you can use a *class* over and over again in a page; since you can only use the *id* once, it takes precedence over the *class* in your stylesheet.