Semantic HTML

Semantic HTML

Semantic HTML introduces meaning to the code we write. Before Semantic HTML the elements didn't have any meaning as to what it does or what content goes in it. An element such as `<div>` was used as a general-purpose element to create things from headers to footers to articles. With Semantic HTML we were introduced to elements that tell developers and browsers exactly what it does and what content should go in it.

```
<!--Non Semantic HTML-->
<div id="footer">
 <p>this is a footer</p>
</div>

<!--Semantic HTML-->
<footer>
 <p>this is a footer</p>
</footer>
```

Element Placement

Semantic HTML introduces elements that can tell developers exactly what the element does or where it's placed based on the name of that element. Some of these elements are `<header>`, `<nav>`, `<main>`, and `<footer>`. `<header>` describes the content at the top of the page `<body>`. It may include a logo, navigational links or a search bar. `<nav>` encapsulates the page's navigational links. It is often placed inside the `<header>` or `<footer>`. `<main>` encapsulates the main content of a page between the header/navigation and the footer areas. `<footer>` includes the page's footer content at the bottom of the `<body>`.

Embedding media

Semantic HTML introduces us to `<video>`, `<audio>` and `<embed>`. `<video>` allows us to add videos to our website. `<audio>` allows us to implement audio into our website. `<embed>` can be used to implement any type of media. These elements are universal in that they all use the `src` attribute to link the source of the content. `<video>` and `<audio>` requires a closing tag while `<embed>` is a self-closing tag.

```
<!--Video Tag-->
<video
src="4kvideo.mp4">video
not supported</video>

<!--Audio Tag-->
<audio
src="koreanhiphop.mp3">
</audio>

<!--Embed tag-->
<embed
src="babyyoda.gif"/>
```

<figure> and <figcaption>

The `<figure>` element is used to encapsulate media such as an image, diagram. or code snippet. The `<figcaption>` element is used to describe the media encapsulated within the `<figure>` element. Developers will normally use `<figcaption>` within the `<figure>` element to group the media and description. This way, if a developer decides to change the position of the media, the description will follow along with it.

```
<figure>
 <img src="qwerty.jpg">
 <figcaption>The image
shows the layout of
a qwerty keyboard.
</figcaption>
</figure>
```

<section> and <article>

codecademy

`<section>` defines elements in a document, such as chapters, headings, or any other area of the document with the same theme. `<article>` holds content that makes sense on its own such as articles, blogs, and comments. Generally developers will use `<section>` to define a theme for the webpage and use `<article>` to write independent content for that theme. This does not mean that `<article>` has to be used with `<section>` .

```
<section>
  <!--defines theme-->
  <h2>Top Sports league
in America</h2>
<!--writes independent
content relating to that
theme-->
  <article>
    <p>One of the top
sports league is the nba.
</p>
  </article>
</section>
```

<aside> Aside Element

The `<aside>` element is used to mark additional information that can enhance another element but isn't required in order to understand the main content. Usually, this information would be in a sidebar or a location where it doesn't obstruct the main piece of content. An example of this would be an article that discusses how to take care of a dog and next to the article an ad would appear advertising a dog grooming product.

```
<article>
<!--Main Content-->
</article>
<aside>
<!--Additional
information-->
</aside>
```