### Day 8 -HTML Semantic Elements

Semantic Elements  =  Elements with meaning

A semantic element clearly describe its meaning to both the browser and the developer

When building web pages, we use a combination of non-semantic HTML and *Semantic HTML*. The word semantic means “relating to meaning,” so semantic elements provide information about the content between the opening and closing tags. By using Semantic HTML, we select HTML elements based on their meaning, not on how they are presented. Elements such as <div> and <span> are not semantic elements since they provide no context as to what is inside of those tags.



Semantic Elements refer to HTML elements that have a specific meaning. For example <h1> is a semantic element. It tells browser that the content within the tag is the most significant header contained in the HTML document. <div> on the other hand, is a non-semantic element as it only indicates a division in the HTML document and provide no information on what goes before, after or within the tag.

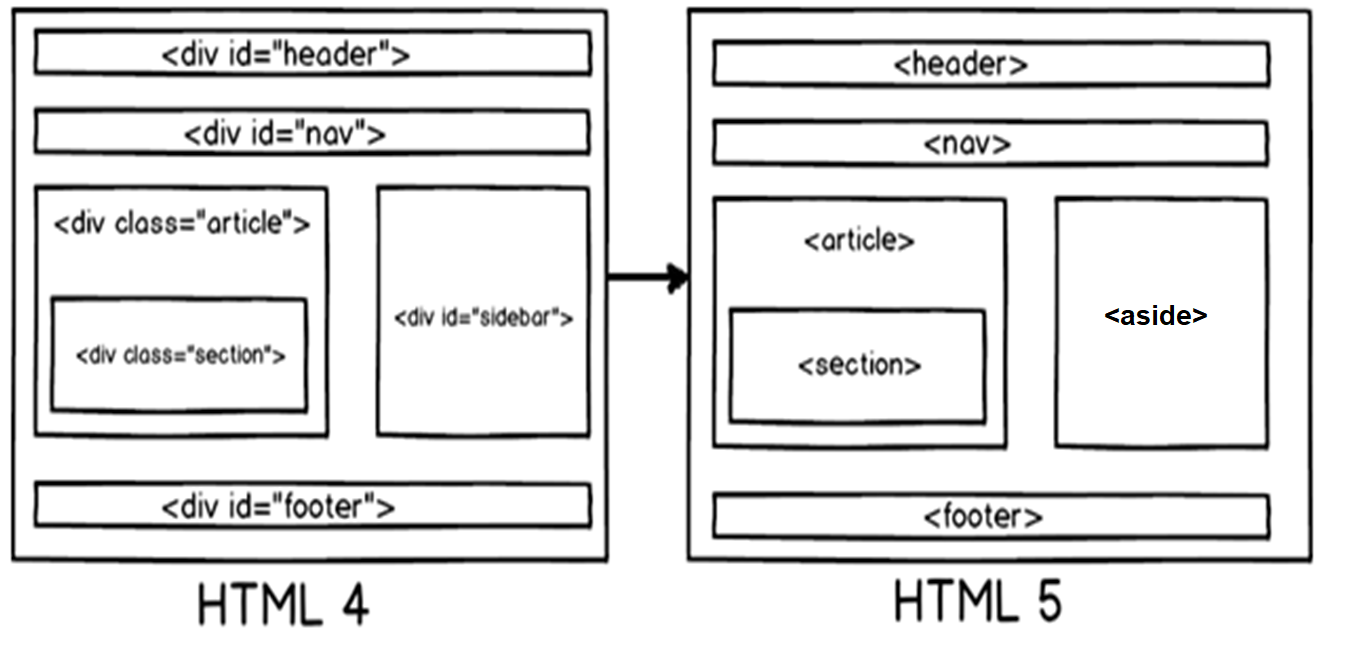
**Why use Semantic HTML?**

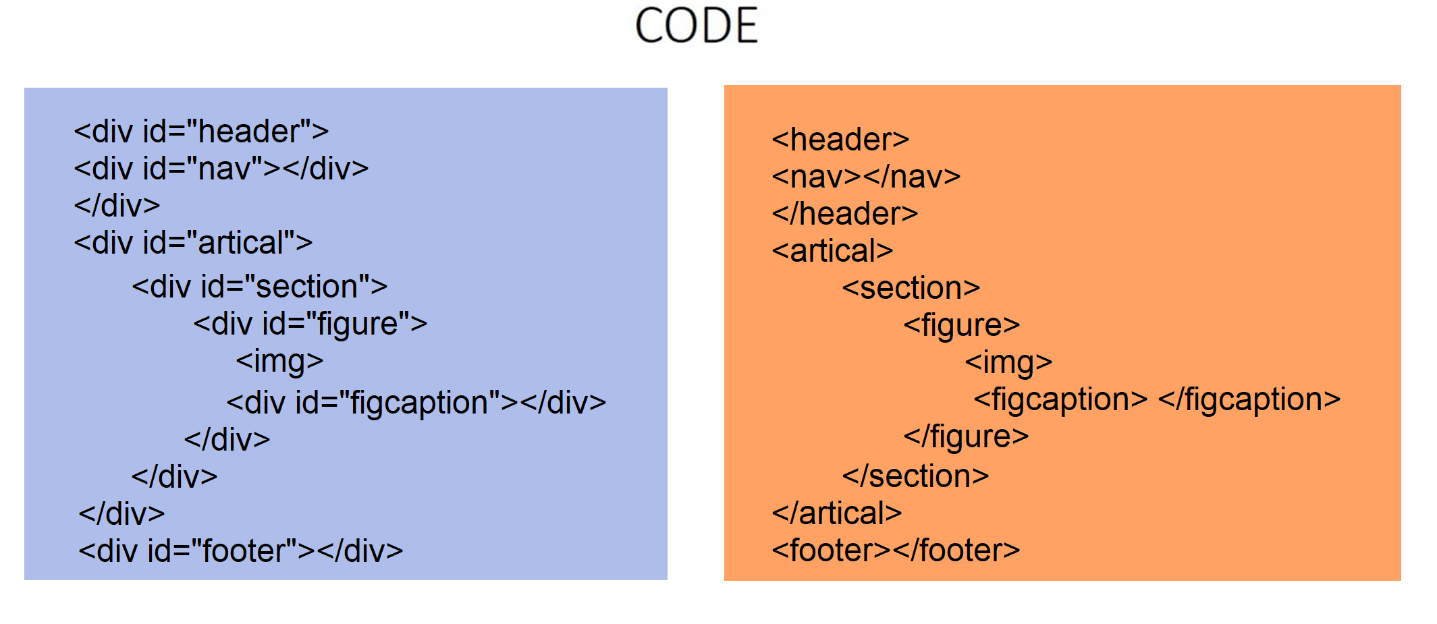
* **Accessibility:** Semantic HTML makes webpages accessible for mobile devices and for people with disabilities as well. This is because screen readers and browsers are able to interpret the code better.
* **SEO:** It improves the website SEO, or *Search Engine Optimization*, which is the process of increasing the number of people that visit your webpage. With better SEO, search engines are better able to identify the content of your website and weight the most important content appropriately.
* **Easy to Understand:** Semantic HTML also makes the website’s source code easier to read for other web developers.

**How is the page structure of HTML 5 different from HTML 4 or previous HTML?**

A typical web page has headers, footers, navigation, central area, and sidebars. Now if we want to represent the same in HTML 4 with proper names to the HTML section we would probably use a DIV tag.

But in HTML 5 they have made it more clearly by creating element names for those sections which makes your HTML more readable.





**New Semantic Tags in HTML5**

The release of HTML5 has seen the introduction of several new semantic tags that provide even richer information to search engines. Here is an overview of several HTML5 elements and what they mean for WordPress on-page SEO:

**<main>**: The <main> tag encloses the dominant content of the blog including all article content and other related sections that extend the central theme of the page, such as the <article> tag and supporting <section> tags.

**<header>**: The <header> tag is similar to the <h1> tag in that it can be used to specify the header of a page. But it can also be used to indicate the header section of a page and can even contain navigation links and other relevant text.

**<nav>**: Navigation is undoubtedly one of the most important aspects of a site. The <nav> tag can be used to specify links on a page such in the main site navigation or for pagination.

**<article>**: The <article> tag makes it easy to mark new blog posts or article entries in an online publication. Search engines can put more weight on any content wrapped with this tag. It also helps to clean up the HTML code by reducing the use of <div> tags.

**<section>**: Blog posts are typically broken into different sections to make it easier for users to find what they are looking for. The <section> tag can be used to specify these subsections of your content, each with their own separate HTML heading.

**<aside>**: An <aside> tag can be like a section tag, but one that focuses on secondary content such as  sidebar, or a post-article call to action might be a good place to use <aside> tags.

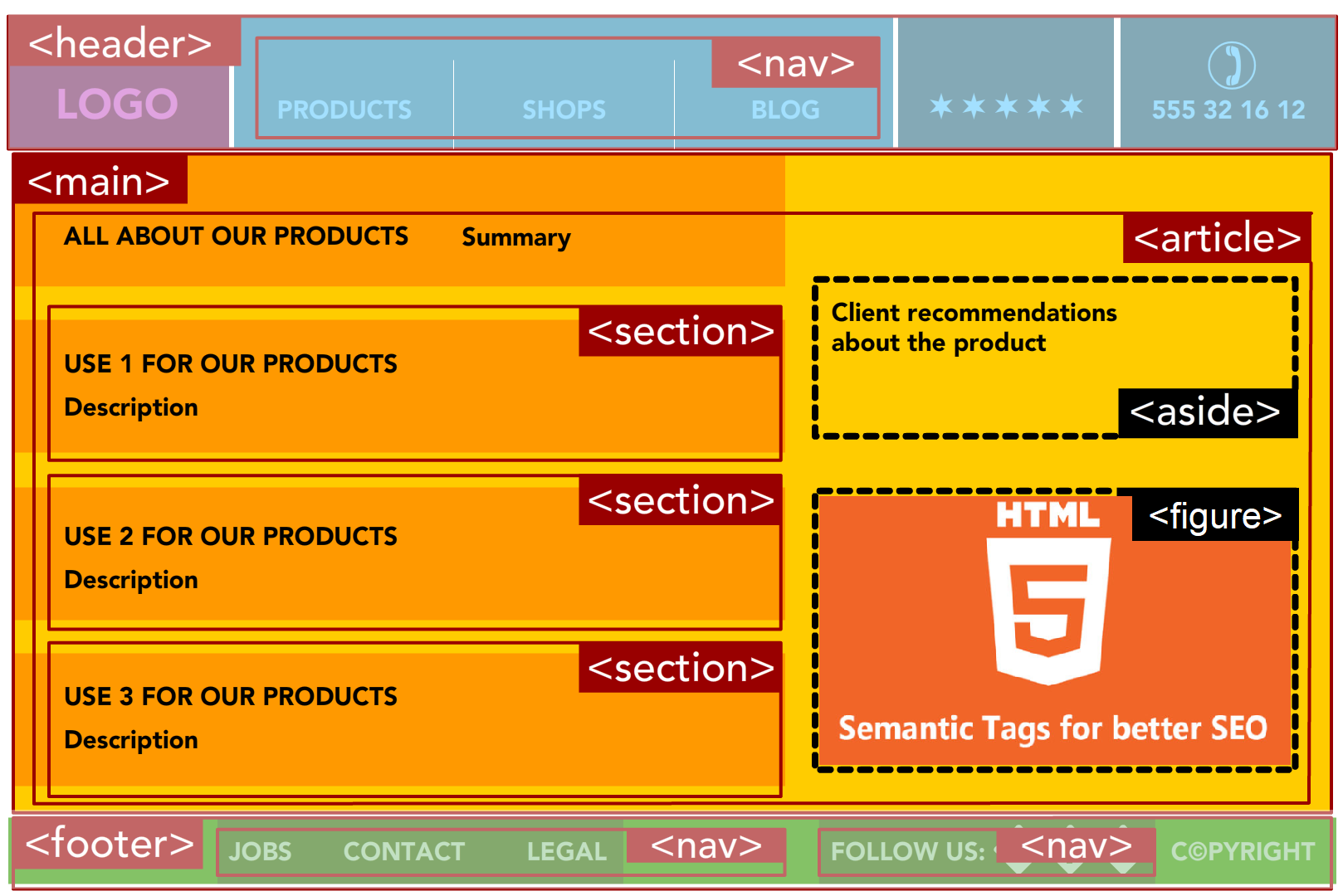
**<footer>**: While not as useful as the <header> tag, the <footer> tag still offers SEO benefits as it can be used to specify content in the footer section of a website such as company information and other useful links. Each page can even have its own footer section.

**<video>**: The <video> tag is easily one of the most useful tags as it allows for cross-browser compatibility to display videos without having to use Flash. HTML5 also makes it possible to include additional information about the video such as captions and subtitles.

**<figure>:** The <figure> element depicts space for **separated content**, such as photos, diagrams, etc. To provide a **caption** for this element, use the <figcaption> tags.

**<details>:** The <details> tag is used for the content/information which is initially hidden but could be displayed if the user wishes to see it. This tag is used to create an interactive widget that the user can open or close it. The content of the details tag is visible when open the set attributes.

**<summary>:** The <summary> tag in HTML is used to define a summary for the <details> element. When the summary is clicked by the user, the content placed inside the <details> element becomes visible which was previously hidden.



**Web-Layout Using Semantic Elements**

|  |  |
| --- | --- |
| <!DOCTYPE html>  <html lang="en">  <head>  <meta charset="UTF-8" />  <meta name="viewport" content="width=device-width, initial-scale=1.0" />  <title>Semantic Tags</title>  <style>  \* {  padding: 0;  margin: 0;  box-sizing: border-box;  font-family: Arial, Helvetica, sans-serif;  }  .container {  width: 100%;  padding: 15px;  }  header {  width: 100%;  background-color: grey;  height: 70px;  display: flex;  justify-content: center;  align-items: center;  }  nav ul {  list-style: none;  }  nav ul li {  display: inline-block;  }  nav ul li a {  display: block;  text-decoration: none;  color: white;  padding: 10px 15px;  font-size: 18px;  text-transform: uppercase;  }  main {  width: 100%;  }  section.banner {  margin-top: 15px;  width: 100%;  height: 80vh;  background-color: rgb(196, 226, 240);  display: flex;  justify-content: center;  align-items: center;  font-size: 40px;  }  section.sec-1 {  display: flex;  justify-content: space-evenly;  align-items: center;  margin-top: 15px;  }  section.sec-1 .circle {  width: 200px;  height: 200px;  background-color: rgb(221, 178, 59);  border-radius: 50%;  }  section.sec-2 {  width: 100%;  margin-top: 15px;  }  section.sec-2::after {  content: '';  display: block;  clear: both;  }  section.sec-2 article {  float: left;  width: 50%;  height: 500px;  background-color: rgb(128, 116, 97);  }  section.sec-2 aside {  width: 45%;  float: right;  height: 500px;  /\* background-color: greenyellow; \*/  display: flex;  flex-direction: column;  }  section.sec-2 aside .rightsidebar {  width: 100%;  background-color: rgba(125, 201, 122, 0.753);  height: 30%;  margin-bottom: 5%;  }  section.sec-2 aside #lastsidebar {  margin-bottom: 0;  }  section.sec-3 {  width: 100%;  height: 300px;  display: flex;  justify-content: space-evenly;  align-content: center;  margin-top: 15px;  }  section.sec-3 .grid {  width: 20%;  height: 100%;  background-color: rgb(238, 60, 60);  }  footer {  margin-top: 15px;  background-color: pink;  width: 100%;  height: 70px;  display: flex;  justify-content: center;  align-items: center;  font-size: 40px;  }  h1 {  text-align: center;  padding: 25px;  }  </style>  </head>  <body>  <div class="container">  <header>  <nav>  <ul>  <li><a href="#">Home</a></li>  <li><a href="#">About</a></li>  <li><a href="#">Services</a></li>  <li><a href="#">Blog</a></li>  <li><a href="#">Contact</a></li>  </ul>  </nav>  </header>  <main>  <section class="banner">BANNER</section>  <h1>Section-1</h1>  <section class="sec-1">  <div class="circle"></div>  <div class="circle"></div>  <div class="circle"></div>  <div class="circle"></div>  </section>  <h1>Section-2</h1>  <section class="sec-2">  <article></article>  <aside>  <div class="rightsidebar"></div>  <div class="rightsidebar"></div>  <div class="rightsidebar" id="lastsidebar"></div>  </aside>  </section>  <h1>Section-3</h1>  <section class="sec-3">  <div class="grid"></div>  <div class="grid"></div>  <div class="grid"></div>  <div class="grid"></div>  </section>  </main>  <footer>Footer</footer>  </div>  </body>  </html> |  |