HYPER TEXT MARKUP LANGUAGE (HTML)

Class 08
Semantic HTML, Inline & Block Elements

Semantic HTML

Semantic HTML involves using HTML elements that convey meaning and structure, going beyond mere visual presentation. In simpler terms, it means using HTML tags that accurately describe the purpose and content they contain.

Backstory of Semantic HTML

BEFORE SEMANTIC HTML
Stuff
Stuff
Stuff
AFTER SEMANTIC HTML
Books
Box
Pens

The Internet was originally invented to share documents, programmers were not concerned about how it looked. As the information grew and so did the Internet, people wanted the web to look nicer. Think of HTML tags like labels you put on different parts of a box. Before, people were using generic labels that talk much about what was inside. For example, instead of saying "clothes" or "books," they just said "stuff."

This was a problem because when someone else looked at the box, they wouldn't know what was inside just by reading the label. Similarly, search engines and screen readers couldn't understand what the content on a webpage was about because it was labeled too generically (nothing specific).

Semantic HTML is like using labels that describe what's inside the box. So now, instead of just saying "stuff," you might use labels like "clothes," "books," or "shoes." This helps everyone - humans and computers to understand what's inside the box, or on the webpage.

HYPER TEXT MARKUP LANGUAGE (HTML)

Semantic HTML means using HTML elements that carry meaning, and structure beyond just presentation. In simpler terms, it means using HTML tags that accurately describe the content they contain. In HTML, for example, the <h1> tag is a semantic element, that tells the computer that it is the main heading of the page.

Advantages of using Semantic HTML

Accessibility: Semantic HTML helps improve accessibility by providing meaningful structure to content, making it easier for screen readers and other assistive technologies to interpret and navigate the webpage.

SEO (Search Engine Optimization): Search engines rely on semantic markup to better understand the content and context of a webpage. Using semantic HTML elements can improve your website's search engine ranking and visibility.

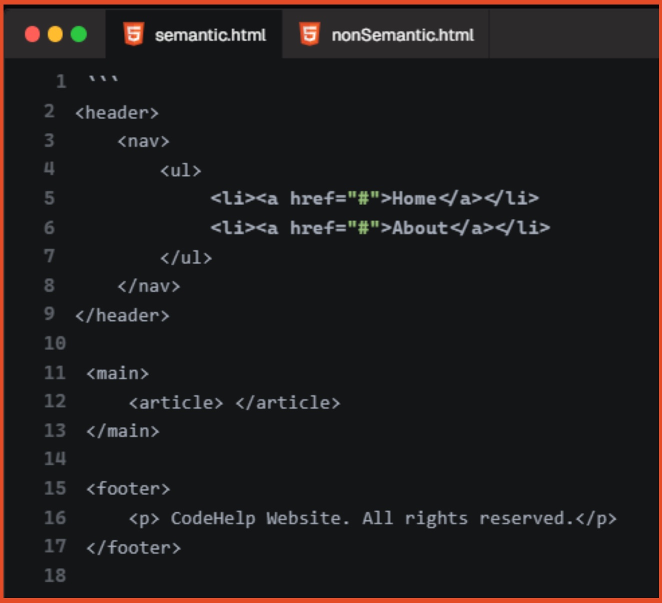
Future-proofing: Semantic HTML is designed to be forward-compatible, meaning that as new technologies and standards emerge, websites built with semantic markup are more likely to remain compatible and functional.

Maintainability: Semantic HTML promotes cleaner and more maintainable code by providing a clear and logical structure. This makes it easier for developers to understand, modify, and debug code, especially in collaborative projects or when revisiting code after some time.

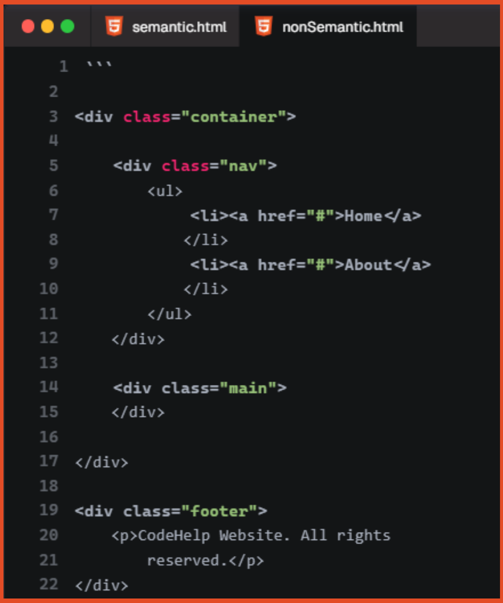
Consistency: Semantic HTML encourages consistent use of markup across different web pages and projects, leading to a more uniform and cohesive user experience.

Why use Semantic HTML?

You must be thinking why do you need to use Semantic HTML elements? Fair question. Let’s find out valid reasons for this change. The most obvious reason is semantic HTML tags are easier to read. When you look at <h1> you know it’s a main heading already. Let’s understand this more clearly. Below are two blocks of code –



~~~
```
<header>
    <nav>
        <ul>
            <li><a href="#">Home</a></li>
            <li><a href="#">About</a></li>
        </ul>
    </nav>
</header>

<main>
    <article> </article>
</main>

<footer>
    <p> CodeHelp Website. All rights reserved.</p>
</footer>
~~~

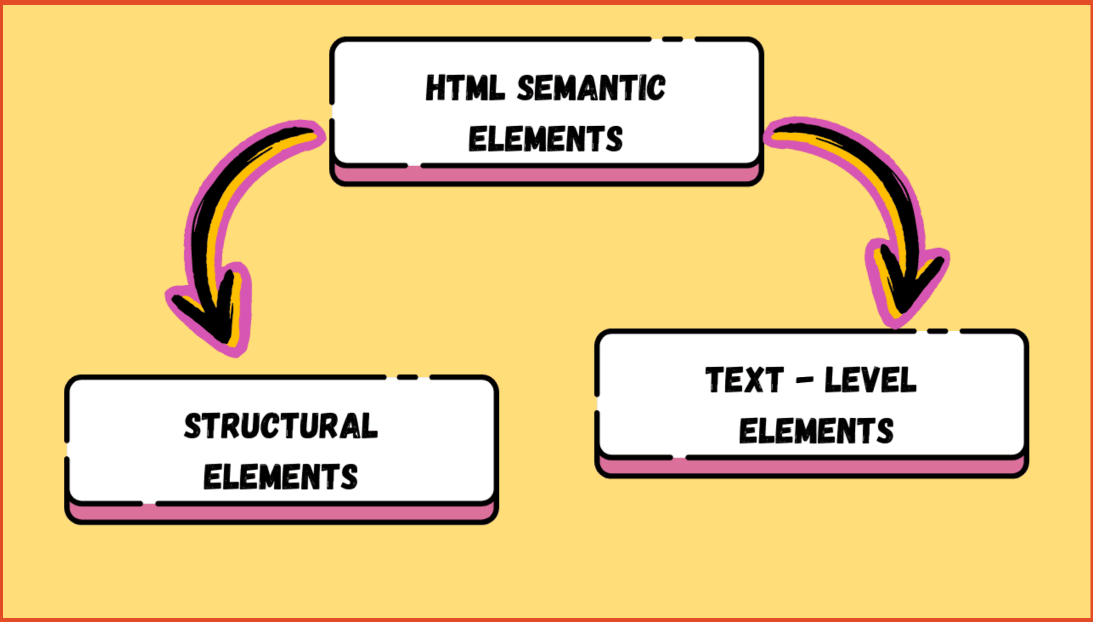


~~~
```

<div class="container">

    <div class="nav">
        <ul>
            <li><a href="#">Home</a>
            </li>
            <li><a href="#">About</a>
            </li>
        </ul>
    </div>

    <div class="main">
    </div>

</div>

<div class="footer">
    <p>CodeHelp Website. All rights
        reserved.</p>
</div>
~~~

HYPER TEXT MARKUP LANGUAGE (HTML)

HTML Semantic Elements

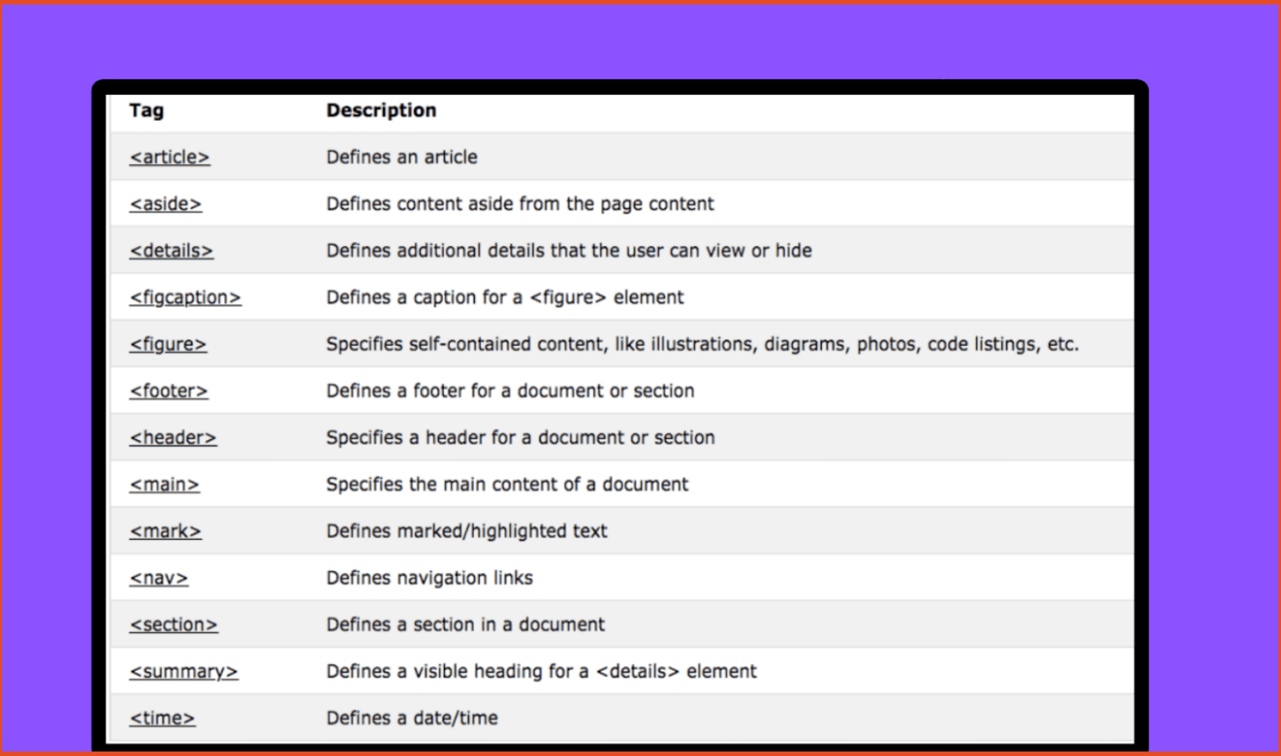
Semantic elements in HTML are those that carry meaning, providing context and structure to the content they enclose. They describe the purpose of the content rather than just its presentation, which is crucial for accessibility, search engine optimization (SEO), and overall maintainability of the code.



Structural Elements: These elements define the different parts of a web page, such as the header, navigation, main content, aside, section, and footer.

Text-level elements: These elements define the type of text content, such as headings, paragraphs, emphasis, and strong emphasis.

List of Semantic Elements in HTML



Tag	Description
<article>	Defines an article
<aside>	Defines content aside from the page content
<details>	Defines additional details that the user can view or hide
<figcaption>	Defines a caption for a <figure> element
<figure>	Specifies self-contained content, like illustrations, diagrams, photos, code listings, etc.
<footer>	Defines a footer for a document or section
<header>	Specifies a header for a document or section
<main>	Specifies the main content of a document
<mark>	Defines marked/highlighted text
<nav>	Defines navigation links
<section>	Defines a section in a document
<summary>	Defines a visible heading for a <details> element
<time>	Defines a date/time

HYPER TEXT MARKUP LANGUAGE (HTML)

Block-level Elements

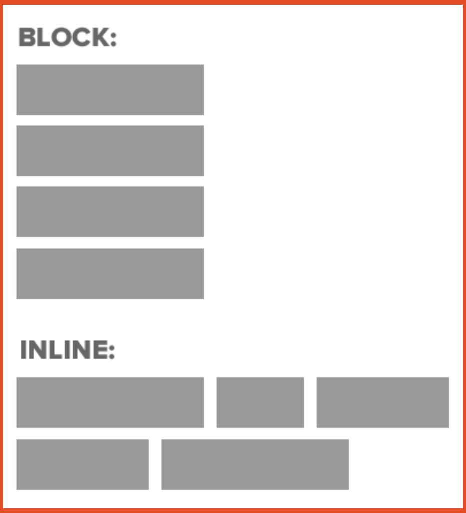
A block-level element always starts on a new line, and the browsers automatically add some space (a margin) before and after the element.

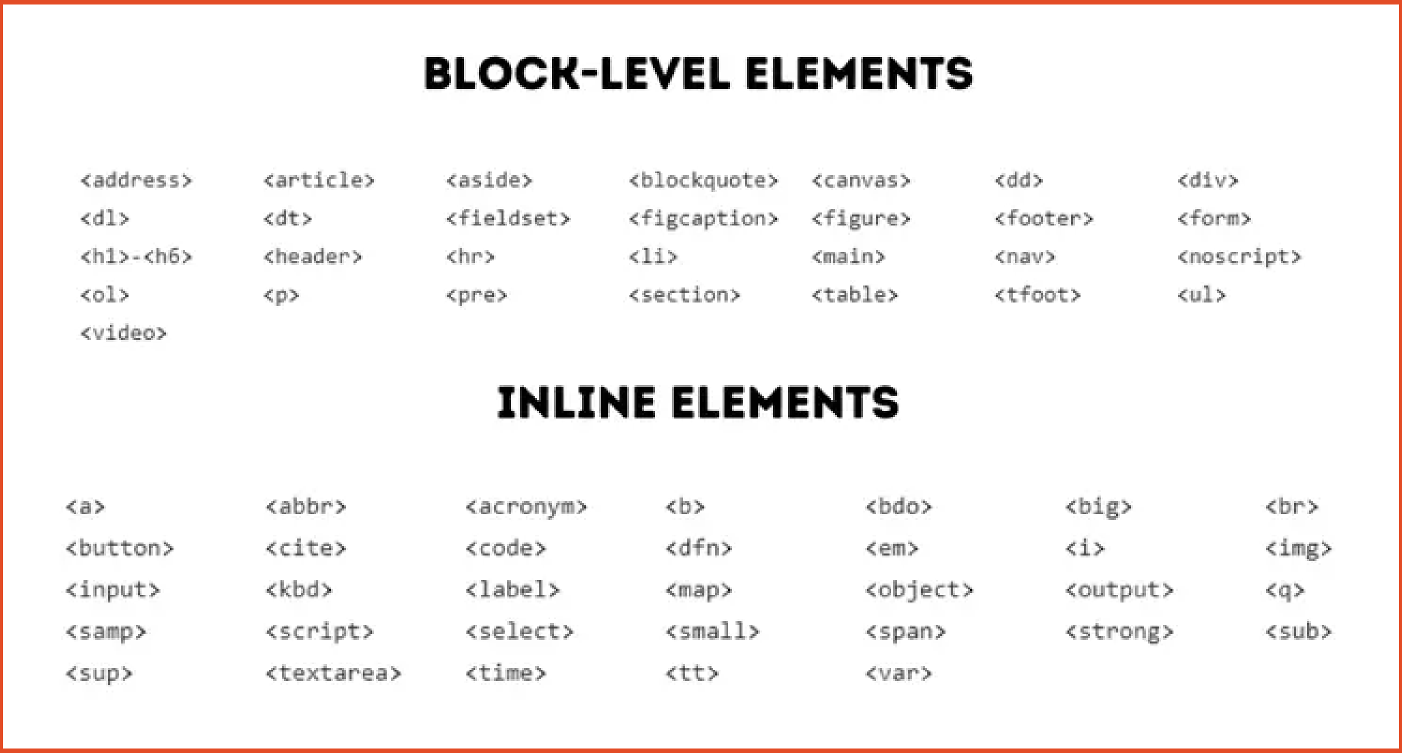
A block-level element always takes up the full width available (stretches out to the left and right as far as it can).

Inline Elements

An inline element does not start on a new line.

An inline element only takes up as much width as necessary.





BLOCK-LEVEL ELEMENTS

`<address>`	`<article>`	`<aside>`	`<blockquote>`	`<canvas>`	`<dd>`	`<div>`
`<dl>`	`<dt>`	`<fieldset>`	`<figcaption>`	`<figure>`	`<footer>`	`<form>`
`<h1>-<h6>`	`<header>`	`<hr>`	`<li>`	`<main>`	`<nav>`	`<noscript>`
`<ol>`	`<p>`	`<pre>`	`<section>`	`<table>`	`<tfoot>`	`<ul>`
`<video>`						

INLINE ELEMENTS

`<a>`	`<abbr>`	`<acronym>`	`<b>`	`<bdo>`	`<big>`	`<br>`
`<button>`	`<cite>`	`<code>`	`<dfn>`	`<em>`	`<i>`	`<img>`
`<input>`	`<kbd>`	`<label>`	`<map>`	`<object>`	`<output>`	`<q>`
`<samp>`	`<script>`	`<select>`	`<small>`	`<span>`	`<strong>`	`<sub>`
`<sup>`	`<textarea>`	`<time>`	`<tt>`	`<var>`		