**Semantic Elements**

* Semantic elements clearly describe their meaning to both the browser and the developer.
* They make your code more readable, accessible, and SEO-friendly.

**Examples:**

* <header> – Represents the header of a section or page.
* <nav> – Defines a section for navigation links.
* <main> – The main content of the document.
* <section> – A section of related content.
* <article> – An independent, self-contained piece of content.
* <footer> – Represents the footer of a section or page.

**Example:**

|  |
| --- |
| <article>  <header>  <h1>HTML Tutorial</h1>  <p>Published on April 20, 2025</p>  </header>  <section>  <p>This tutorial covers the basics of HTML...</p>  </section>  <footer>  <p>Author: Sachin</p>  </footer>  </article> |

**Non-Semantic Elements**

Non-semantic elements don’t describe their meaning. They are generic containers with no specific role.

**Examples:**

* <div> – Generic container for block-level content.
* <span> – Generic container for inline content.

|  |
| --- |
| <div class="header">  <h1>HTML Tutorial</h1>  </div>  <div class="content">  <p>This tutorial covers the basics of HTML...</p>  </div>  <div class="footer">  <p>Author: Sachin</p>  </div> |

**Key Differences:**

|  |  |  |
| --- | --- | --- |
| **Feature** | **Semantic Elements** | **Non-Semantic Elements** |
| Meaningful tags | Yes (<header>, <nav>) | No (<div>, <span>) |
| Accessibility | Better for screen readers | Less informative |
| SEO | Helps search engines understand content | Not SEO-friendly by default |
| Readability | More readable and self-explanatory | Depends on class names or comments |