indixpert

HTML ELEMENTS

HTML ELEMENT

An HTML element is a fundamental building block of a webpage. It represents a part of the content or structure of the webpage and is defined by an opening tag, optional content, and a closing tag.

Structure: HTML Element = Start Tag + Content + End Tag

Example:

```
<p>This is a Paragraph</p>
```

In this example:

- <p> is the opening tag (for the paragraph element),
- "This is a paragraph." is the content,
- </p> is the closing tag.

TYPE OF HTML ELEMENT

Block Level Elements

These elements typically take up the full width available, with a new line before and after. Common examples include <div>, <p>, <h1>, <ul>, <section>, and <article>

Inline Elements

These elements only take up as much width as their content requires and don’t break the flow of text. Examples include <span>, <a>, <strong>, and <img>

Empty (Self-closing) Elements

Some elements don’t require a closing tag. These are often referred to as self-closing or void elements. For example, <img>, <input>, and <br>

TYPE OF HTML ELEMENT

Nested Elements

HTML elements can be nested inside each other to create a hierarchical structure. The enclosing element is often referred to as the "parent," while the enclosed element is called the "child."

Example:

```
<div>
    <h1>This is a Heading</h1>
    <p>This is a Paragraph</p>
 </div>
```

In this example:

- <div> is the parent element.
- Inside the <div>, there is a heading (<h1>) and a paragraph (<p>).

CONTAINERS IN HTML

In HTML, containers are essential for organizing and grouping content. There are two types of containers Block Containers (<div>) & Inline Container <span>.

Common Uses

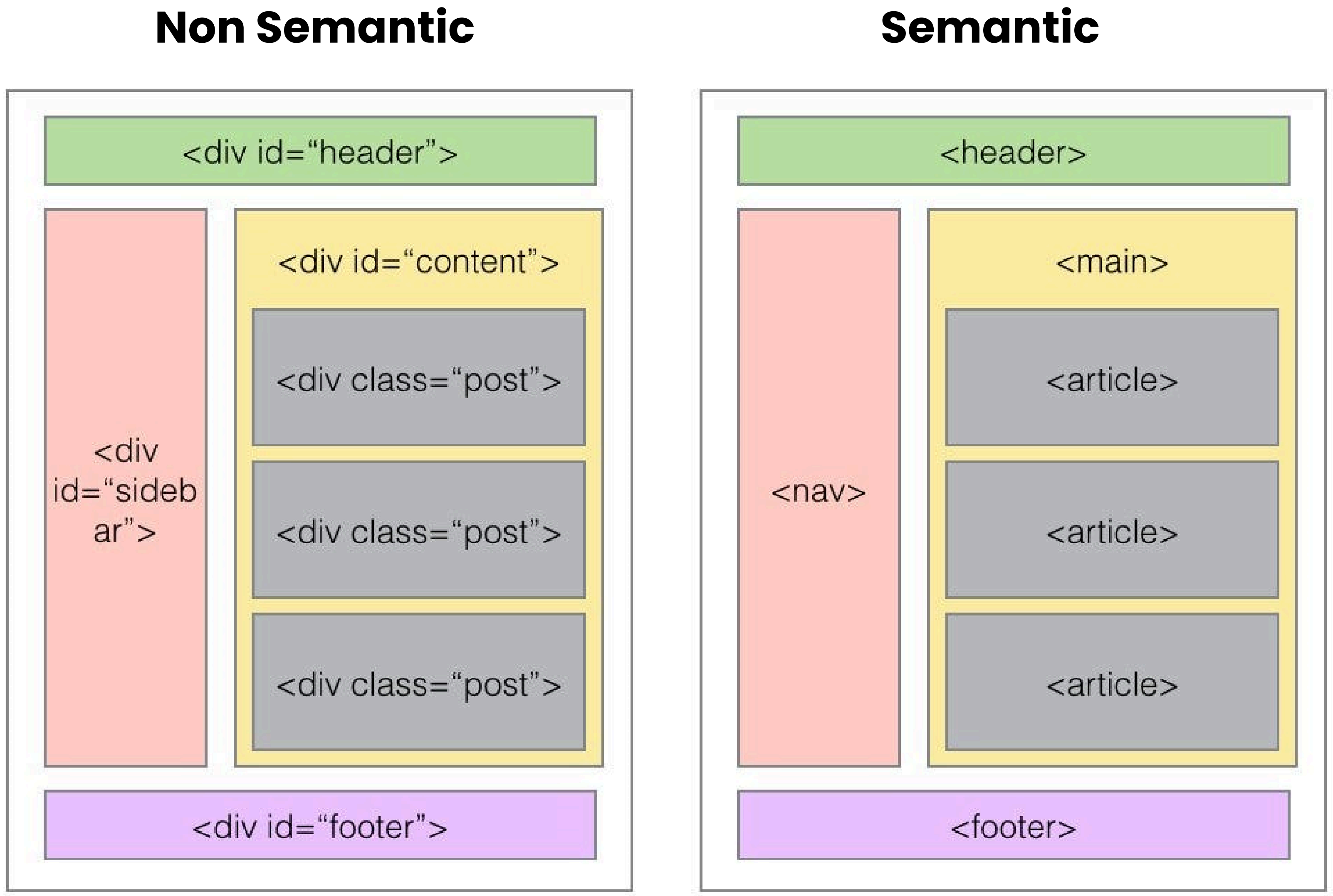
- Grouping elements for layout (like creating sections of a page).
- Applying styles or JavaScript to a group of elements.
- Organizing content into logical sections of a webpage.

Example:

```
<div>
    <h1>This is a <span>Heading</span></h1>
    <p>This is a <span>Paragraph</span></p>
 </div>
```

SEMANTIC ELEMENTS

Semantic elements are HTML elements that convey meaning about their content both to the browser and to developers. They help improve code readability and accessibility



Non Semantic

Semantic

SEMANTIC ELEMENTS

Why Use Semantic Elements?

Accessibility: Semantic elements improve accessibility because they provide clear meaning for screen readers and other assistive technologies.

SEO: Search engines can better understand the content of the page and rank it accordingly.

Maintainability: Clear, meaningful HTML code is easier to maintain, especially for teams or future developers.

Code Readability: Semantic elements make the code more readable and self-explanatory.

SEMANTIC ELEMENTS

<header>	For introductory content like logo and navigation
<nav>	To add navigation
<main>	To define the actual content of the page apart from header, sidebar and footer
<article>	To define Independent contents like blog post or news article
<section>	To group related contents
<footer>	To define footer content, typically copyright information and contact information.

SEMANTIC ELEMENTS

Best Practices

Use the right element for the right purpose, Use article and section wisely.

Use semantic tag instead of div and span whenever possible.

Always keep the structure clean and easy for better readability.

Always check for accessibility via developer tools and other accessibility tools.

indixpert

THANK YOU

PHONE NUMBER

(+91) 778 899 2897

WEBSITE

www.indixpert.com