COESIONE ITALIA 21-27 PIEMONTE

Cofinanziato dall'Unione europea

REGIONE PIEMONTE

Web Developer

HTML, CSS e Strumenti di Digital Marketing (SEO, SEM, SEA)

Docente: Shadi Lahham

Intervento realizzato da ITS TECNOLOGIE DELL'INFORMAZIONE E DELLA COMUNICAZIONE

Semantic coding

On page seo

Shadi Lahham - Web development

Semantic coding

Semantic coding

Semantic coding is essential for on-page SEO, as it helps search engines understand the content on a web page. This involves using HTML tags like <header>, <article>, and <footer> to define the structure of the content. These tags provide context, making it easier for search engines to interpret the page's relevance and improve its ranking

Implementing semantic code enhances accessibility and user experience by creating a logical structure that's easier to navigate. It also ensures that content is more easily interpreted by screen readers, benefiting users with disabilities and leading to longer site engagement

Benefits for SEO

Semantic coding offers several SEO benefits, including clearer differentiation between content types, such as distinguishing navigation from main text. This helps search engines index pages more accurately, improving relevance in search results

Moreover, semantic code is future-proof, ensuring compatibility with evolving web standards and search engine algorithms. As search engines become more sophisticated, the use of semantic HTML positions websites for sustained SEO success

Key semantic elements

<header>
introductory content, typically a group of introductory or navigational aids

<nav>
defines a set of navigation links

<main>
the dominant content of the **<body>** of a document; there is only one **<main>** element in a document

<section>
defines a section in a document, typically with a thematic grouping of content

<article>
a self-contained piece of content that could be distributed independently such as a blog post or a news article

Key semantic elements

<aside>
a portion of content indirectly related to the main content such as sidebars or quotes

<footer>
a closing element for a section or the entire document, providing additional information or navigation

<figure>
specifies content, like illustrations, diagrams, or photos, that is referenced from the main content, along with a **<figcaption>** to provide a caption

<picture>
defines a container for multiple image sources

Additional semantic elements

<audio>
shows an embedded sound or audio stream

<video>
embeds video content in an HTML document without requiring additional plugins

<embed>
embeds multimedia content like videos, sounds, or external apps

<details>
describes a widget from which the user can obtain additional information or controls on-demand

<summary>
provides a summary visible to the user, used along with the **<details>** element

Additional semantic elements

<dialog>
defines a dialog box or a subwindow for user interaction as needed

<svg>
embeds an SVG file in an HTML document

<time>
encodes dates and times in a machine-readable format

Non-semantic HTML

```
<div id="header">
  <h1>My Website</h1>
  <div id="nav">
    <ul>
      <li><a href="home.html">Home</a></li>
      <li><a href="about.html">About</a></li>
    </ul>
  </div>
</div>
<div id="content">
  <h2>Welcome</h2>
  <p>This is my website where I share my projects and blog posts.</p>
</div>
<div id="footer">
  <p>&copy; 2024 My Website</p>
</div>
```

Semantic HTML

```
<header>
  <h1>My Website</h1>
  <nav>
    <ul>
      <li><a href="home.html">Home</a></li>
      <li><a href="about.html">About</a></li>
    </ul>
  </nav>
</header>
<main>
  <section>
    <h2>Welcome</h2>
    <p>This is my website where I share my projects and blog posts.</p>
  </section>
</main>
<footer>
  <p>&copy; 2024 My Website</p>
</footer>
```

Coherent content outline

Structuring HTML5 effectively creates a coherent content outline that enhances visual layout, SEO, and accessibility, making navigation and understanding easier

1. Use semantic markup
use meaningful tags like <**article**>, <**section**>, and <**header**> to describe the content's purpose

2. Consistent use
apply semantic tags consistently throughout your site pages to maintain a clear and uniform structure

3. Logical sections
divide content using <**section**> to create distinct areas within a page

4. Navigational aids
include navigational aids such as a table of contents or breadcrumbs to improve user experience using the <**nav**> element to wrap these aids

Coherent content outline

5. Content hierarchy
maintain a logical content hierarchy using headings <h1> to <h6> which helps both users and search engines understand the structure and importance of your content

6. Accessibility features
incorporate accessibility features to ensure all users can navigate your site effectively

Content hierarchy

<h1>
should be used for the main heading of the page, typically containing the primary keyword or a key phrase that summarizes the page's topic

<h2>
should be used for subheadings that support the H1, typically including secondary keywords and breaking down the content into main sections

<h3>, <h4>, <h5>, <h6>
should be used to further organize content under H2 headings to help create a clear hierarchy and improve readability, making it easier for users and search engines to understand the structure of the content

this approach provides a well-structured and comprehensive content layout that benefits SEO, readability, navigation and user experience

Content hierarchy

```
<h1>Understanding Web Development</h1>

<h2>Introduction</h2>
<p>An overview of web development, focusing on core concepts</p>

<h2>Key Concepts</h2>
<h3>Frontend</h3>
<p>Deals with user-facing parts using HTML, CSS, and JavaScript</p>
<h3>Backend</h3>
<p>Handles server-side tasks like databases and logic</p>

<h2>Tools</h2>
<h3>Version Control</h3>
<p>Git and similar tools manage code changes</p>

<h2>Conclusion</h2>
<p>Web development requires knowledge of both frontend and backend</p>
```

Semantic structure outliner tools

HTML Outliner
A tool to check the outline of an HTML document to ensure proper semantic structure

The W3 validator outliner
Nu Html Checker
provides an outline when the checkbox is selected

Chrome plugin
HTML5 Outliner - Chrome Web Store

exercise: try them on the following page Examples of semantic SEO and then other pages

guide: refer to this generic guide on outliners

Semantic code & rich snippets

Rich snippets, like ratings, reviews, or product details, are enhanced search listings that provide additional information at a glance

Semantic html, combined with structured data, improves these snippets by helping search engines understand and display structured content thus increasing the chances of rich snippets, making search results more engaging

Accessibility features

Accessibility features

Semantic coding is crucial for accessibility because screen readers and assistive technologies depend on semantic tags to understand and navigate web pages, with the **<nav>** tag indicating navigational links and allowing users to skip to relevant content

Additionally, semantic HTML improves keyboard navigation by clearly defining page sections, enabling users to jump between **<header>**, **<main>**, and **<footer>** sections and enhancing overall website accessibility for a broader audience, including those with disabilities

Accessibility features

Image descriptions
Use alt attributes to provide concise and descriptive text for all images. This helps visually impaired users understand the content

Form labels
Employ <label> elements to associate clear and informative text with form inputs. This makes it easier for users to understand the purpose of each field

ARIA attributes
Accessible Rich Internet Applications attributes provide additional semantic information to assistive technologies, such as screen readers, to help users with disabilities understand and interact with web content. They can be used to define the role of an element, its state, and its properties

ARIA roles
Utilize ARIA attributes to convey additional information and context to assistive technologies, such as screen readers. This helps users with disabilities navigate your website more

ARIA attributes

```html
<!-- button with an icon has an aria-label "search" to indicate its purpose to screen reader users -->
<button aria-label="search">
  <i class="icon-search"></i>
</button>

<!-- image with an alt attribute describes its content for screen readers -->
<img src="a-beautiful-sunset.jpg" alt="A beautiful sunset">

<!-- accessible form with label associations and aria-labels for screen readers -->
<form>
  <label for="email">Email:</label>
  <input type="email" id="email" name="email" aria-label="Email address">
  <button type="submit" aria-label="Submit Form">Submit</button>
</form>
```

aria-label

provides a textual label for an element that is not visible to users which is particularly useful for elements like buttons or icons that may not have a visible text label

ARIA roles

use role="navigation" if the div represents a navigation bar

use role="alert" if the span indicates an error message

use role="complementary" if the div provides additional information related to the main content

use role="banner" if the div serves as a header of a page or section
use role="contentinfo" if the div serves the footer of a page or section

role attribute
specifies the user interface element type for assistive technologies, aiding screen readers in understanding its function, and is typically used on non-semantic elements to provide additional context or meaning, requiring careful use to avoid confusion

ARIA roles

```
<!-- using role="button" to explicitly define an element as a button -->
<div role="button" aria-label="search">
  <i class="icon-search"></i>
</div>

<!-- <header> already identifies the site's header region; no need for `role="banner"` -->
<header role="banner">
  <h1>Website Header</h1>
  <nav>
    <ul>
      <li><a href="#home">Home</a></li>
    </ul>
  </nav>
</header>
```

ARIA roles

```html
<!-- using role="dialog" to define a modal window -->
<div role="dialog" aria-labelledby="dialog-title" aria-modal="true">
  <h2 id="dialog-title">Important Message</h2>
  <p>This is a modal dialog with some important information.</p>
  <button aria-label="Close">X</button>
</div>
```

aria-labelledby

The aria-labelledby attribute links to elements that serve as the label for the current element, aiding assistive technologies in describing it

aria-modal

The aria-modal attribute indicates if an element is modal, helping assistive technologies manage focus by signaling that the user must interact with the modal before accessing other content

Try

try the Nu Html Checker with the outline option as well as the HTML5 Outliner Chrome plugin on the following page Examples of semantic SEO and see how the page outline is structured

try the tools on some different types of online pages