Web development

lesson 3

Sep 15, 2024

HTML Content Elements

Heading Elements: Importance of Headings

- HTML provides six levels of headings, from `<h1>` to `<h6>`.

- Headings help structure the content and improve accessibility.

- Example:

```
<h1>Main Heading</h1>
<h2>Subheading</h2>
```

Heading Elements: Best Practices

- Use `<h1>` for the main title of the page.

- Subheadings should follow a hierarchical order (`<h2>`, `<h3>`, etc.).

- Avoid skipping heading levels to maintain a clear structure.

Paragraphs & Text: The Paragraph Tag

- The `<p>` tag defines a paragraph of text.

- Example:

```
<p>This is a paragraph.</p>
```

- Paragraphs provide a way to group sentences and create separation in text.

Paragraphs & Text: Line Breaks and Emphasis

- The `<br>` tag inserts a line break without starting a new paragraph.

- Example:

```
<p>Line one.<br>Line two.</p>
```

- The `<strong>` and `<em>` tags are used for bold and italic text, respectively.

Paragraphs & Text: Inline vs. Block Elements

- Paragraphs are block-level elements, meaning they take up the full width available.

- Inline elements (e.g., `<strong>`, `<em>`) only take up as much space as necessary.

- Example:

```
<p>This is a <strong>strong</strong> word in a paragraph.</p>
```

List Elements: Unordered Lists

- The `<ul>` tag creates an unordered list with bullet points.

- Each list item is defined with the `<li>` tag.

- Example:

```
<ul>
  <li>First item</li>
  <li>Second item</li>
</ul>
```

List Elements: Ordered Lists

- The `<ol>` tag creates an ordered list with numbered items.

- Like unordered lists, items are defined with the `<li>` tag.

- Example:

```
<ol>
  <li>First item</li>
  <li>Second item</li>
</ol>
```

List Elements: Nested Lists

- Lists can be nested inside other lists to create subcategories.

- Example:

```
<ul>
  <li>Item 1
    <ul>
      <li>Subitem 1</li>
    </ul>
  </li>
</ul>
```

Anchor Element: Creating Links

- The `<a>` tag creates hyperlinks to other documents or web pages.

- The `href` attribute specifies the link's destination.

- Example:

```html
<a href="https://example.com">Visit Example</a>
```

Anchor Element: Link Attributes

- The `target="_blank"` attribute opens the link in a new tab.

- Example:

```
<a href="https://example.com" target="_blank">Open in New Tab</a>
```

- The `title` attribute provides additional information when the mouse hovers over the link.

Link to document fragments in a page

It's possible to link to a specific part of an HTML document, known as a document fragment, rather than just to the top of the document. To do this you first have to assign an id attribute to the element you want to link to. It normally makes sense to link to a specific heading, so this would look something like the following:

```html
<h2 id="Mailing_address">Mailing address</h2>
```

Then to link to that specific id, you'd include it at the end of the URL, preceded by a hash/pound symbol (#), for example:

```html
<p>
  Want to write us a letter? Use our
  <a href="contacts.html#Mailing_address">mailing address</a>.
</p>
```

You can even use the document fragment reference on its own to link to another part of the current document:

```html
<p>
  The <a href="#Mailing_address">company mailing address</a> can be found at the
  bottom of this page.
</p>
```

Button Element: Basic Button

- The `<button>` element represents a clickable button.

- Example:

```html
<button>Click Me</button>
```

- Buttons can trigger actions, submit forms, or be styled as links.

Button Element: Customizing Buttons

- Buttons can be customized with CSS for different styles.

- Example:

```
<button style="background-color: blue; color: white;">Styled Button</button>
```

- Buttons can also contain icons or images.

Image Element: Adding Images

- The `<img>` tag is used to embed images in an HTML document.

- The `src` attribute specifies the image file's location.

- Example:

```
<img src="image.jpg" alt="Description">
```

Image Element: Alt Text and Accessibility

- The `alt` attribute provides alternative text for the image.

- This is important for accessibility and for cases when the image cannot be displayed.

- Example:

```
<img src="image.jpg" alt="A scenic view">
```

Image Element: Sizing and Styling

- Images can be resized using the `width` and `height` attributes.

- Example:

```
<img src="image.jpg" alt="Description" width="200" height="100">
```

- CSS can also be used to style images (e.g., adding borders or shadows).

- Link external images, eg: PixaBay

Conclusion

- We've explored HTML syntax, page structure, head information, and key content elements.

- Mastery of these fundamentals is crucial for building well-structured and accessible web pages.

- Keep practicing by creating your own HTML documents and experimenting with different elements.

Webdev video courses

NEW

HTML + CSS Basics

59 lessons • 5 hours

How many users? ⓘ

−   1   +

$129

JavaScript Basics

66 lessons • 9 hours

$129

by Todd Motto

NEW

JavaScript DOM

47 lessons • 7 hours

$129

by Todd Motto

NEW

JavaScript HTML5 APIs

72 lessons • 7 hours

$129

by Todd Motto

NEW

JavaScript Masterclass

23 lessons • 4 hours

$129

by Todd Motto

References and Guides on MDN

Learn web development

HTML section elements

Header and Footer

The `<header>` HTML element represents introductory content, typically a group of introductory or navigational aids. It may contain some heading elements but also a logo, a search form, an author name, and other elements.

```html
<header>
  <a class="logo" href="#">Cute Puppies Express!</a>
</header>
```

The `<footer>` HTML element represents a footer for its nearest ancestor sectioning content or sectioning root element. A `<footer>` typically contains information about the author of the section, copyright data or links to related documents.

```html
<footer>
    <p>This is updated at XXXX by XXX.</p>
</footer>
```

Navigation element

The `<nav>` HTML element represents a section of a page whose purpose is to provide navigation links, either within the current document or to other documents. Common examples of navigation sections are menus, tables of contents, and indexes.

```html
<nav class="menu">
  <ul>
    <li><a href="#">Home</a></li>
    <li><a href="#">About</a></li>
    <li><a href="#">Contact</a></li>
  </ul>
</nav>
```

Article element

The `<article>` HTML element represents a self-contained composition in a document, page, application, or site, which is intended to be independently distributable or reusable (e.g., in syndication). Examples include: a forum post, a magazine or newspaper article, or a blog entry, a product card, a user-submitted comment, an interactive widget or gadget, or any other independent item of content.

```html
<article class="film_review">
  <h2>Jurassic Park</h2>
  <section class="main_review">
    <h3>Review</h3>
    <p>Dinos were great!</p>
  </section>
  <section class="user_reviews">
    <h3>User reviews</h3>
    <article class="user_review">
      <h4>Too scary!</h4>
      <p>Way too scary for me.</p>
      <footer>
        <p>
          Posted on
          <time datetime="2015-05-16 19:00">May 16</time>
          by Lisa.
        </p>
      </footer>
```

Section and div element

The `<section>` HTML element represents a generic standalone section of a document, which doesn't have a more specific semantic element to represent it. Sections should always have a heading, with very few exceptions.

```
<section>
  <h2>Heading</h2>
  <p>Bunch of awesome content</p>
</section>
```

The `<div>` HTML element is the generic container for flow content. It has no effect on the content or layout until styled in some way using CSS (e.g. styling is directly applied to it, or some kind of layout model like Flexbox is applied to its parent element)

```
<div>
  <p>
    Any kind of content here. Such as &lt;p&gt;, &lt;table&gt;. You name it!
  </p>
</div>
```

main and aside element

The `<main>` HTML element represents the dominant content of the `<body>` of a document. The main content area consists of content that is directly related to or expands upon the central topic of a document, or the central functionality of an application

```html
<body>
  <a href="#main-content">Skip to main content</a>
  <!-- navigation and header content -->
  <main id="main-content">
    <!-- main page content -->
  </main>
</body>
```

The `<aside>` HTML element represents a portion of a document whose content is only indirectly related to the document's main content. Asides are frequently presented as sidebars or call-out boxes

```html
<article>
  <p>
    The Disney movie <cite>The Little Mermaid</cite> was first released to
    theatres in 1989.
  </p>
  <aside>
    <p>The movie earned $87 million during its initial release.</p>
```

Figure element

The `<figure>` HTML element represents self-contained content, potentially with an optional caption, which is specified using the `<figcaption>` element. The figure, its caption, and its contents are referenced as a single unit

```html
<figure>
  <img src="https://developer.mozilla.org/media/cc0-images/elephant-660-480.jpg" alt="Elephant at sunset" />
  <figcaption>An elephant at sunset</figcaption>
</figure>
```