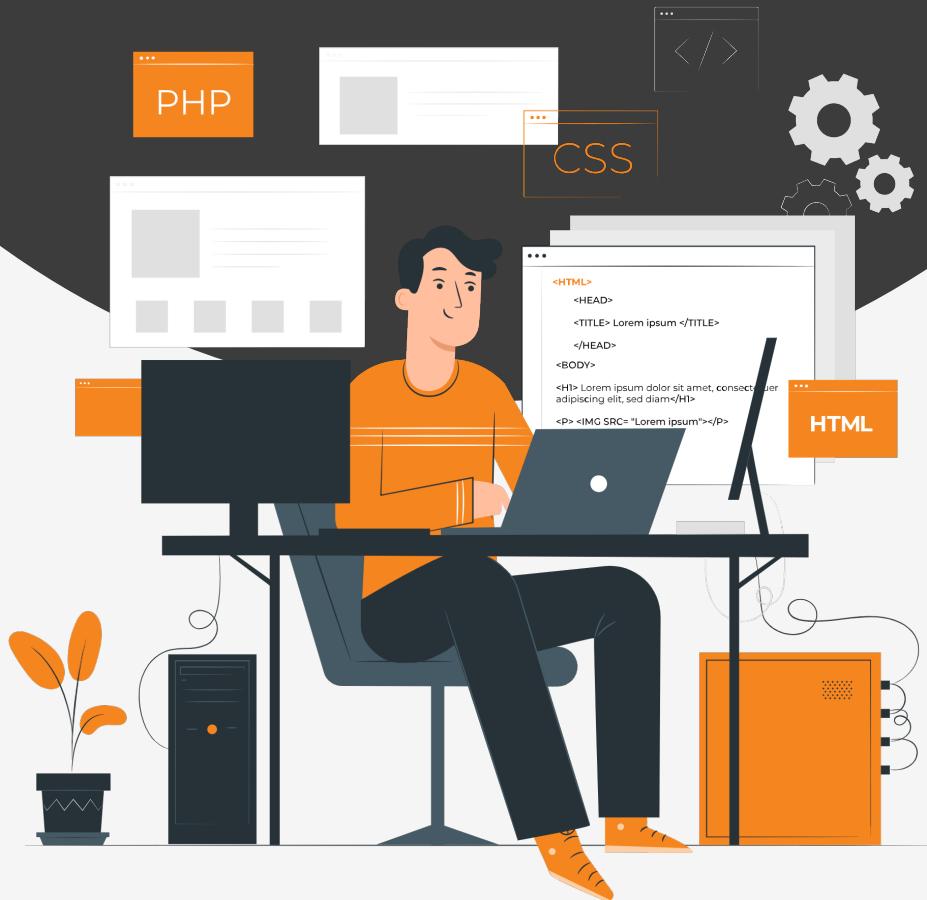


Lesson:

HTML Entities



Topics Covered

1. Introduction to HTML Entities
2. Purpose and usage of HTML entities
3. ASCII character set and Unicode
4. Commonly used HTML entities
5. Numeric Character References vs. Named Character Entities
6. Special text tags (<kbd>, <code>, etc...)

Introduction to HTML Entities

The reserved characters used in HTML documents are known as HTML entities. They are not seen on the typical keyboard. They provide a wide range of characters, allowing you to add icons, geometric shapes, mathematical operators, and so on.

HTML entities are useful when you want to display special characters that may conflict with HTML syntax or when you want to ensure proper rendering across different platforms and browsers that may have different character encoding settings.

It's important to note that with the advancement of web technologies, the need for using HTML entities has diminished to some extent.

Let us discuss more about the HTML entities

Purpose and usage of HTML entities

The main two purposes are:

1. Displaying Special Characters
2. Handling Reserved Characters

Displaying Special Characters

HTML entities are used to display characters that have special meanings in HTML or characters that cannot be easily typed or displayed directly.

For example, if you want to display the copyright symbol (©) or the trademark symbol (™) on a webpage, you would use the corresponding HTML entities "©" and "™", respectively.

Handling Reserved Characters

HTML uses certain characters for its syntax and structure, such as the less-than symbol (<), the greater-than symbol (>), and the ampersand symbol (&). These reserved characters can cause issues if used directly in HTML code, as they may be interpreted as part of the markup. By using HTML entities like "<", ">", and "&", you can display these characters as text on a webpage without affecting the HTML structure.

The entity code is enclosed in ampersands (&) and semicolons (;) to differentiate it from regular text.

Unset

```
&entity_name;
```

// OR

```
&#entity_number;
```

For example, to display the less-than symbol (<), you would use "<". The browser then interprets the HTML entity and renders the appropriate character on the webpage.

HTML entities are especially useful when working with characters that are not available on the keyboard or when you want to ensure consistent rendering of characters across different platforms and browsers. They help in maintaining the integrity of HTML code and displaying special characters accurately in HTML documents.

Link to character reference chart: [Link](#)

Commonly used HTML entities

The reserved characters used in HTML documents are known as HTML entities. They are not seen on the typical keyboard. They provide a wide range of characters, allowing you to add icons, geometric shapes, mathematical operators, and so on.

HTML entities are useful when you want to display special characters that may conflict with HTML syntax or when you want to ensure proper rendering across different platforms and browsers that may have different character encoding settings.

It's important to note that with the advancement of web technologies, the need for using HTML entities has diminished to some extent.

Let us discuss more about the HTML entities

Result	Description	Entity Name	Entity Number
<	Less than	<	<
>	greater than	>	>
&	ampersand	&	&
”	double quotation mark	"	"
’	single quotation mark (apostrophe)	'	'
©	copyright	©	©
®	registered trademark	®	®

Numeric Character References vs. Named Character Entities

Numeric Character References represent characters using their Unicode (code point) in decimal or hexadecimal format.

Example

```
Unset
&#60; // for less than (<)
&#62; // for greater than (>)
&#38; // for ampersand (&)
```

Named Character Entities represent characters using predefined names that correspond to specific characters.

Example

```
Unset
&lt; // for less than (<)
&gt; // for greater than {>}
&amp; // for ampersand (&)
```

ASCII character set and Unicode

ASCII

The ASCII (American Standard Code for Information Interchange) was one of the first encoding standards to offer a truly universal character set for basic electronic communications. It is a widely used character encoding standard that represents characters using 7 bits and 8 bits, allowing for a total of 128 and 256 characters respectively. The ASCII character set indeed uses 8 bits to represent a total of 256 characters. Originally, ASCII was defined using 7 bits, allowing for 128 characters. However, an extended version of ASCII known as "extended ASCII" or "ASCII-8" uses the 8th bit to represent an additional 128 characters, bringing the total to 256.

ASCII includes basic Latin letters (A-Z, a-z), digits (0-9), punctuation marks, control characters, and some special characters.

Link to ASCII characters reference table: [Link](#)

Unicode

Unicode is a character encoding standard that aims to represent almost every character from every writing system in the world including over 149,000 characters. Unicode provides a unique numerical value (code point) for every character including punctuation

marks, mathematical symbols, technical symbols, arrows, emojis, and various scripts, including Latin, Greek, Cyrillic, Chinese, Japanese, Korean, Arabic, Hebrew, and many more, allowing for a much larger range of characters to be represented compared to ASCII. Since its inception, Unicode has been adopted by all modern software providers, allowing the transportation of data through devices, applications, and platforms without corruption.

It is now used in all major operating systems, browsers, search engines, laptops, smartphones, and across the internet.

The Unicode standard is maintained by the Unicode Consortium, a non-profit organization that exists to develop and promote the Unicode Standard. The Unicode Standard supports three encoding forms: UTF-8, UTF-16, and UTF-32.

What is UTF?

UTF stands for "Unicode Transformation Format". Unicode can be implemented by different character sets. The most commonly used encodings are UTF-8 and UTF-16. The default character encoding in HTML-5 is UTF-8, HTML 4 supports UTF-8. HTML 5 supports both UTF-8 and UTF-16

What is the difference between Unicode and UTF-8?

Unicode is a character set while UTF-8 IS encoding.

Unicode is basically a list of characters with unique decimal numbers (code point)

Let's look at how the code point of Alphabets

A = 65

B = 66

C = 67

D = 68 ...etc

Encoding is how these numbers are translated into binary numbers which the computer can understand and store.

Example

Let's convert "hey" to Unicode

hey = 104 101 121

Now let's convert the "hey" to UTF-8 encoding

hey = 01101000 01100101 01111001

So we can say that Encoding converts numbers into binary. Character sets convert characters to numbers.

Special text tags (`<kbd>`, `<code>`)

Let's discuss about some special text tags in HTML.

- `<kbd>`

The `<kbd>` tag is used to indicate user input, typically representing keyboard input or commands.

By default, the content wrapped in the `<kbd>` tags is displayed in a monospace font

Example

```
Unset
<p>
  press
  <kbd>Ctrl</kbd> + <kbd>S</kbd>
  to save a file or document
</p>
```

OUTPUT

press Ctrl + S to save a file or document

In the above example output, you can notice that the Ctrl and S is in different font

- **<code>**

The `<code>` tag is used to represent code snippets of code within an HTML document. By default, the content wrapped in the `<code>` tags is displayed in a monospace font.

```
Unset
<code>

  let name = nasikh; <br />
  let age = 21;
</code>
```

OUTPUT

```
let name = nasikh;
let age = 21;
```

You can see that the change in the font and the readability of the code also increases

- **<pre>**

The `<pre>` tag is used to preserve whitespace and line breaks in the content within it. It is typically used to display preformatted text, such as ASCII art or text that needs to maintain its original spacing.

Example

```
Unset
<pre>
Welcome to PW skills.

    -----
   |   |   \     /
  |---|   \ / \ /
   |           \|  \/
</pre>
```

OUTPUT

Welcome to PW skills.



From the output, it's clearly understandable that the texts inside `<pre>` tag will be maintaining the original spacing as we written inside the code

- **<samp>**

The `<samp>` tag is used to represent sample output or results from a computer program. It is often used in combination with the `<code>` tag to display both code snippets and its output.

Example

```
Unset
<pre>
<p>code:</p>
<code>print("Welcome to PW Skills!");</code>
<p>output:</p>
<samp>Welcome to PW Skills!</samp>
```

OUTPUT

code:

```
print("Welcome to PW Skills!");
```

output:

```
Welcome to PW Skills!
```

- **<blockquote>**

The `<blockquote>` tag is used to indicate a block of quoted text or a longer excerpt from another source. It is typically used to distinguish quoted content from the rest of the text.

Example

```
Unset
<blockquote>
  <p>
    "Pure Hardwork, No Shortcuts!" -Inside blockquote
  </p>
</blockquote>
<p>
  "Pure Hardwork, No Shortcuts!" -without blockquote
</p>
```

OUTPUT

"Pure Hardwork, No Shortcuts!" -Inside blockquote

"Pure Hardwork, No Shortcuts!" -without blockquote

- **<abbr>**

The `<abbr>` tag is used to define an abbreviation or acronym. It allows developers to specify the full meaning of the abbreviation, which can be displayed as a tooltip when the user hovers over the abbreviation.

Example

```
Unset
<p>The <abbr title="Physics wallah">PW</abbr> was founded in 2016 as a YouTube
Channel by Alakh Pandey</p>
```

OUTPUT

The PW was founded in 2016 as a YouTube Channel by Alakh Pandey

Physics wallah

When we move our cursor on the text PW , we can see that the tooltip has appeared.

- **<mark>**

The `<mark>` tag is used to highlight or mark a specific portion of the text. It is typically used to draw attention to a particular section or emphasize important content.

Example

```
Unset
<p>Please remember to <mark>submit your assignment by Friday</mark>.</p>
```

OUTPUT

Please remember to [submit your assignment by Friday](#).

The text is highlighted as you can see in the above text

Using these tags helps improve the readability and understanding of the content, particularly when presenting code examples, keyboard shortcuts, or user input within HTML documents.