HTML Documentation

HTML introduction

Introduction

HTML is a language made up of elements, which can be applied to pieces of text to give them different meaning in a document (Is it a paragraph? Is it a bulleted list? Is it part of a table?), structure a document into logical sections (Does it have a header? Three columns of content? A navigation menu?), and embed content such as images and videos into a page.

HTML (HyperText Markup Language) is a markup language that tells web browsers how to structure the web pages you visit. It can be as complicated or as simple as the web developer wants it to be. HTML consists of a series of elements, which you use to enclose, wrap, or mark up different parts of content to make it appear or act in a certain way.

Prerequisites

you should have at least basic familiarity with using computers and using the web passively.You should have a basic work environment set up (as detailed in Installing basic software), and understand how to create and manage files as detailed in Dealing with files.

Basics of HTML

In this article, we cover the absolute basics of HTML. To get you started, this article defines elements, attributes, and all the other important terms you may have heard. It also explains where these fit into HTML. You will learn how HTML elements are structured, how a typical HTML page is structured, and other important basic language features. Along the way, there will be an opportunity to play with HTML too!

HTML Documentation

Anatomy of an HTML element

The anatomy of our element is:

- # The opening tag: This consists of the name of the element (in this example, p for paragraph), wrapped in opening and closing angle brackets. This opening tag marks where the element begins or starts to take effect. In this example, it precedes the start of the paragraph text.

- # The content:This is the content of the element. In this example, it is the paragraph text.

- # The closing tag: This is the same as the opening tag, except that it includes a forward slash before the element name. This marks where the element ends. Failing to include a closing tag is a common beginner error that can produce peculiar results.

Example:

HTML

```
<p> My cat is very grumpy.</p>
```

Attributes

Attributes contain extra information about the element that won't appear in the content. In this example, the class attribute is an identifying name used to target the element with style information.

Example of class attribute:

```
<p class="editor-note">My cat is very grumpy
<p>
```

Attributes contain extra information about the element that won't appear in the content. In this example, the `class` attribute is an identifying name used to target the element with style information.

HTML Documentation

HTML Introduction

Basics of HTML

What is head in HTML

An attribute should have:

- A space between it and the element name. (For an element with more than one attribute, the attributes should be separated by spaces too.)
- The attribute name, followed by an equal sign.
- An attribute value, wrapped with opening and closing quote marks.

Anatomy of an HTML Document

Individual HTML elements aren't very useful on their own. Next, let's examine how individual elements combine to form an entire HTML page:

HTML
_____

Copy to Clipboard

```
<!doctype html>
<html lang="en-US">
<head>
<meta charset="utf-8" />
<title>My test page<title>
</head>
<body>
This is my page
<body>
</html>
```

- `!DOCTYPE html`: The doctype. When HTML was young (1991-1992), doctypes were meant to act as links to a set of rules that the HTML page had to follow to be considered good HTML. . `!DOCTYPE html` is the shortest string of characters that counts as a valid doctype. That is all you need to know!

- `<html></html>`: The `<html>` element. This element wraps all the content on the page. It is sometimes known as the root element.

HTML Documentation

- `<head></head>`: The `<head>` element. This element acts as a container for everything you want to include on the HTML page, that isn't the content the page will show to viewers. This includes keywords and a page description that would appear in search results, CSS to style content, character set declarations, and more. You will learn more about this in the next article of the series.

- `<title></title>`: The `<title>` element. This sets the title of the page, which is the title that appears in the browser tab the page is loaded in. The page title is also used to describe the page when it is.

- `<body></body>`: The `<body>` element. This contains all the content that displays on the page, including text, images, videos, games, playable audio tracks, or whatever else.

What is head in HTML

what is the html head

The head of an HTML document is the part that is not displayed in the web browser when the page is loaded. It contains information such as the page `title`, links to CSS (if you choose to style your HTML content with CSS), links to custom favicons, and other metadata (data about the HTML, such as the author, and important keywords that describe the document). Web browsers use information contained in the head to render the HTML document correctly.

Adding a title

this can be used to add a title to the document. This however can get confused with the `h1` element, which is used to add a top level heading to your body content — this

HTML Documentation

is also sometimes referred to as the page title. But they are different things!

- The `h1` element appears on the page when loaded in the browser — generally this should be used once per page, to mark up the title of your page content (the story title, or news headline, or whatever is appropriate to your usage.
- The `<title>` element is metadata that represents the title of the overall HTML document (not the document's content.

Meta element

Metadata is data that describes data, and HTML has an "official" way of adding metadata to a document — the `meta` element.

html

```
<meta charset="utf-8"/>
```

this is an just example.

Applying CSS and JavaScript to HTML

Just about all websites you use in the modern day will employ CSS to make them look cool, and JavaScript to power interactive functionality, such as video players, maps, games, and more. These are most commonly applied to a web page using the `link` element and the `script` element, respectively.

HTML text fundamentals

The basics:heading and paragraphs

One of HTML's main jobs is to give text structure so that a browser can display an

HTML Documentation

HTML Introduction

Basics of HTML

What is head in HTML

HTML document the way its developer intends. This article explains the way HTML can be used to structure a page of text by adding headings and paragraphs, emphasizing words, creating lists, and more.

In HTML, each paragraph has to be wrapped in a `<p>` element, like so:

HTML

```
<p>I am a paragraph, oh yes I am.</p>
```

Each heading has to be wrapped in a heading element:

HTML

```
<h1>I am a paragraph, oh yes I am.</h1>
```

Lists

Now let's turn our attention to lists. Lists are everywhere in life—from your shopping list to the list of directions you subconsciously follow to get to your house every day, to the lists of instructions you are following in these tutorials! On the web, we have three types of lists: unordered, ordered, and description.

Unordered and ordered lists are very common, and they're covered in this section. Description lists are less common, and we'll cover them in Advanced text formatting.

Lets talk about unordered list:

Unordered

Unordered lists are used to mark up lists of items for which the order of the items doesn't matter. Let's take a shopping list as an example:

```
<ul>
<li>milk<li>
<li>eggs<li>
<li>bread<li>
<li>hummus<li>
<ul>
```

HTML Documentation

Creating hyperlinks

what is a hyperlink

Hyperlinks are one of the most exciting innovations the Web has to offer. They've been a feature of the Web since the beginning, and are what makes the Web a web. Hyperlinks allow us to link documents to other documents or resources, link to specific parts of documents, or make apps available at a web address. Almost any web content can be converted to a link so that when clicked or otherwise activated the web browser goes to another web address (URL).

anatomy of a hyperlink

A basic link is created by wrapping the text or other content inside an `<a>` element and using the `href` attribute, also known as a Hypertext Reference, or target, that contains the web address.

```html
<p>

I'm creating a link to

<a href="https://www.mozilla.org/en-US/"> the Mozilla homepage</a>.

<p>
```

Reference

All the documentation in this page is taken from MDN.