Introduction to HTML

Alex Manochio

Introduction to HTML HTML (HyperText Markup Language) is the foundation of web development. It defines the structure of web pages and is interpreted by browsers to display text, images, and other resources.

Table of Contents

1. What is HTML?
2. History of HTML
3. HTML Syntax and Structure
4. Basic HTML Elements
5. Text Formatting Tags
6. Hyperlinks and Navigation
7. Images and Media Embedding
8. Lists: Ordered, Unordered, Definition
9. Tables: Structuring Data
10. Forms: Collecting User Data
11. Semantic HTML: Accessible, Meaningful Web Pages
12. Meta Tags and SEO Optimization
13. Multimedia Elements
14. Scripting with HTML
15. Linking Stylesheets and Scripts
16. Comments and Notes in HTML
17. Entities and Special Characters
18. Canvas and SVG Graphics
19. Accessibility Features
20. Responsive Web Design Principles
21. HTML APIs Overview
22. Drag and Drop API
23. Geolocation API
24. Web Storage API
25. Web Workers
26. HTML Templates
27. Iframes and Embedded Content
28. Best Practices in HTML
29. Advanced Meta Information
30. Future of HTML: HTML6 and Beyond

What is HTML? HTML is a markup language used for structuring content on the internet. It allows developers to define elements like headings, paragraphs, images, and links.

History of HTML

- 1991: HTML 1.0 by Tim Berners-Lee
- 1995: HTML 2.0
- 1997: HTML 3.2 and 4.0
- 2014: HTML5 introduced rich media features without plugins

HTML Syntax and Structure Every HTML document starts with a document type declaration:

```html
<!DOCTYPE html>
```

Followed by:

```html
<html>
<head>
<title>Page Title</title>
</head>
<body>
<p>Hello, world!</p>
</body>
</html>
```

Text Formatting Tags

- `<h1>`-`<h6>`: Headings
- `<p>`: Paragraph
- `<br>`: Line break
- `<hr>`: Horizontal rule
- `<strong>` / `<b>`: Bold text
- `<em>` / `<i>`: Italic text
- `<mark>`: Highlighted text
- `<del>`: Deleted text
- `<ins>`: Inserted text
- `<sub>` / `<sup>`: Subscript and superscript

Hyperlinks and Navigation

```
<a href="https://example.com" target="_blank">Example Page</a>
```

- `target="_blank"` opens the link in a new tab.
- `title="description"` adds hover text.

Images and Media

- `<img src="logo.png" alt="Company Logo" width="200">`
- Audio:

```
<audio controls>
<source src="sound.mp3" type="audio/mpeg">
</audio>
```

- Video:

```
<video controls width="320">
<source src="clip.mp4" type="video/mp4">
</video>
```

Lists

- Ordered List (`<ol>`)
- Unordered List (`<ul>`)
- Definition List (`<dl>`)

Tables Use `<table>` for structured data.

```
<table>
<tr><th>Title</th><th>Year</th></tr>
<tr><td>HTML5 Guide</td><td>2024</td></tr>
</table>
```

Forms and Inputs Collect user data via forms.

```
<form action="submit.php" method="post">
<input type="text" name="username">
<input type="submit">
</form>
```

Supports inputs like `email`, `password`, `file`, `checkbox`, `radio`, `color`, `date`, and `range`.

Semantic HTML

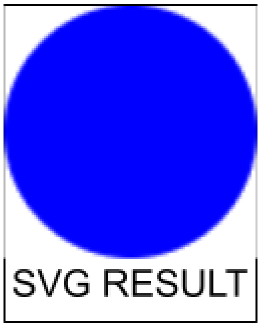
- `<header>`, `<footer>`, `<main>`: Layout sections
- `<section>`, `<article>`: Group related content
- `<aside>`: Sidebar content
- `<nav>`: Navigation menus

Meta Tags and SEO Enhance page visibility and behavior.

```
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta name="description" content="A full HTML tutorial">
```

HTML Multimedia Canvas and SVG elements enable dynamic graphics.

```
<canvas id="myCanvas"></canvas>
<svg height="100" width="100">
<circle cx="50" cy="50" r="40" fill="blue" />
</svg>
```



HTML APIs Modern browsers support APIs accessible through HTML.

Drag and Drop

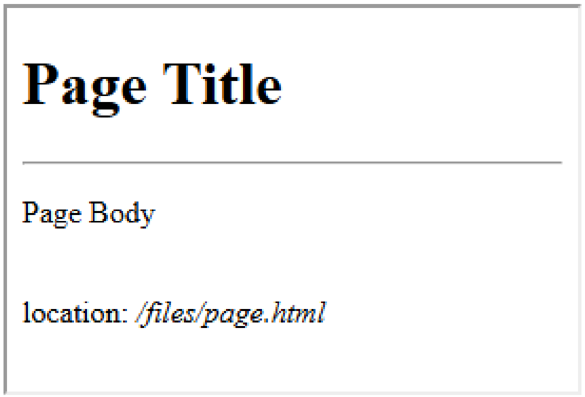
- Enable items to be dragged and dropped between areas.

Geolocation

- Get device location using `navigator.geolocation`.

Web Storage

- `localStorage` and `sessionStorage` for data persistence.



Page Title

Page Body

location: */files/page.html*

Web Workers

- Run scripts in the background without affecting UI.

HTML Templates Reusable fragments via `<template>`.

Iframes and Embedded Content

```
<iframe src="page.html" width="300" height="200"></iframe>
```

Comments

```
<!-- This is a comment -->
```

Entities

- `&lt;`, `&gt;`, `&amp;`, `&copy;`, `&euro;`

Best Practices

- Use semantic tags
- Avoid deprecated tags (`<font>`, `<center>`)
- Validate HTML
- Write accessible and mobile-friendly code

Fancy Styling with Fonts and CSS Use Google Fonts or @font-face.

```
<link href="https://fonts.googleapis.com/css?family=Roboto&display=swap" rel="styleshe
<style>
body { font-family: 'Roboto', sans-serif; }
</style>
```

Responsive Design Use viewport meta tag and CSS media queries.

Future of HTML Exploring WebAssembly, HTML6, and tighter JavaScript integrations.