Day 1 - HTML Foundations

Complete Study Guide

Master the Building Blocks of the Web

Table of Contents

1. Introduction to HTML

What is HTML?

HTML (HyperText Markup Language) is the standard language for creating web pages. It provides the structure and content of websites, acting as the skeleton upon which all web content is built.

Why Learn HTML?

- **Foundation of the Web:** Every website uses HTML as its core structure

- **Easy to Learn:** Simple syntax and logical structure make it beginner-friendly

- **Universal:** Works on all browsers and devices

- **Gateway Skill:** Opens doors to CSS, JavaScript, and web development

How HTML Works

HTML uses tags (enclosed in angle brackets) to define elements. Most tags come in pairs with an opening tag and a closing tag. The browser reads these tags and displays the content accordingly.

> **Key Concept: HTML is markup, not programming. You're marking up text to give it meaning and structure, not writing logic or calculations.**

2. HTML Page Structure

The Anatomy of an HTML Document

Every HTML document follows a specific structure. Think of it like building a house - you need a foundation, walls, and a roof in the right order.

2.1 DOCTYPE Declaration

The very first line of any HTML document. It tells the browser which version of HTML you're using.

```
<!DOCTYPE html>
```

Why it matters: Without this, browsers might display your page incorrectly by using 'quirks mode' instead of standards mode.

2.2 The HTML Element

The root element that wraps all content on the page. It should include the lang attribute to specify the language.

```
<html lang="en"> ... </html>
```

The 'lang' attribute helps screen readers pronounce content correctly and assists search engines in understanding your content's language.

2.3 The HEAD Section

Contains metadata and information ABOUT the page, not displayed directly to users.

Essential elements inside HEAD:

- **meta charset:** Defines character encoding (UTF-8 supports all languages)

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

- **meta viewport:** Makes your page responsive on mobile devices

```
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
```

- **title:** The text shown in browser tabs and search results

```
<title>Day 1 - Profile</title>
```

Think of HEAD as the backstage area - users don't see it, but it's essential for the show to run smoothly.

2.4 The BODY Section

Contains all visible content: text, images, links, videos - everything users see and interact with.

```
<body> ... all your visible content ... </body>
```

3. Text Content Elements

3.1 Headings (h1 - h6)

Headings create hierarchy and structure in your content. There are six levels, from h1 (largest/most important) to h6 (smallest/least important).

Best Practices:

• Use only ONE h1 per page (typically your main title)

• Don't skip levels (don't jump from h1 to h3)

• Use headings for structure, not just to make text bigger

• Think of headings like a book outline: h1 is the book title, h2 are chapters, h3 are sections within chapters

Example hierarchy:

```
<h1>My Personal Website</h1>

<h2>About Me</h2>

<h3>My Education</h3>

<h3>My Experience</h3>

<h2>My Projects</h2>
```

3.2 Paragraphs

Paragraphs are blocks of text, automatically adding space before and after.

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

Important: Multiple spaces or line breaks in your code are displayed as a single space in the browser. To create actual line breaks, use the br tag.

3.3 Line Breaks

The br tag creates a line break without starting a new paragraph.

```
<br /> or <br>
```

When to use:

• In addresses

• In poems where line structure matters

• Rarely in regular paragraphs (usually separate paragraphs are better)

Note: br is a 'self-closing' tag - it doesn't wrap content, so it doesn't need a closing tag. The slash at the end is optional in HTML5.

Understanding Text Semantics

In HTML, we use tags to give meaning to content, not just appearance. This helps screen readers, search engines, and browsers understand your content better.

4. Links and Images

4.1 Creating Links (Anchor Tags)

Links connect web pages together. The 'a' tag (anchor) creates clickable links.

Basic syntax:

```
<a href="https://example.com">Click here</a>
```

Key attributes:

- **href:** The destination URL (where the link goes)

- **target="_blank":** Opens link in a new tab/window

- **rel="noreferrer":** Privacy protection when opening external links in new tabs

Types of links:

1. **External links:** Link to other websites

```
<a href="https://www.google.com" target="_blank" rel="noreferrer">Visit
Google</a>
```

2. **Internal links:** Link to other pages in your site

```
<a href="about.html">About Us</a>
```

3. **Email links:** Open user's email client

```
<a href="mailto:contact@example.com">Email Me</a>
```

Best practices for links:

- Use descriptive text - avoid 'click here'

- Always include rel="noreferrer" with target="_blank" for security

- Make sure links clearly indicate where they go

4.2 Adding Images

Images make web pages visually engaging. The 'img' tag embeds images into your page.

Basic syntax:

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

Essential attributes:

- **src:** The path/URL to the image file (REQUIRED)

- **alt:** Alternative text description (REQUIRED for accessibility)

- **width/height:** Size in pixels (optional, but recommended)

Why alt text matters:

- Screen readers read it aloud to visually impaired users

- Displays if image fails to load

- Helps search engines understand image content

- Legal requirement for accessibility compliance

Writing good alt text:

- Describe what's IN the image, not just what it is

- Keep it concise but informative

- Skip phrases like 'image of' or 'picture of'

Examples:

```
Good: <img src="sunset.jpg" alt="Orange sunset over calm ocean waters" />
Poor: <img src="sunset.jpg" alt="sunset" />
```

Image file paths:

- Same folder: src="photo.jpg"

- Subfolder: src="images/photo.jpg"

- External URL: src="https://example.com/photo.jpg"

5. Lists and Organization

5.1 Unordered Lists (Bullet Points)

Use unordered lists when the order doesn't matter - like a shopping list or features list.

Structure:

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

Key points:

- 'ul' stands for unordered list
- Each item goes inside an 'li' (list item) tag
- Browsers automatically add bullet points

5.2 Ordered Lists (Numbered)

Use ordered lists when sequence matters - like recipe steps or rankings.

Structure:

```
<ol>
<li>First step</li>
<li>Second step</li>
<li>Third step</li>
</ol>
```

Key points:

- 'ol' stands for ordered list
- Browsers automatically number the items
- Numbers update automatically if you add/remove items

5.3 When to Use Each Type

Choose **unordered lists** for:

- Features or benefits
- Navigation menus
- Sets of related items

Choose **ordered lists** for:

- Step-by-step instructions
- Rankings or top-ten lists
- Any sequence where order matters

Common Mistakes to Avoid

- Don't put text directly in ul or ol - always use li tags
- Don't use lists just for indentation - that's a CSS job
- Lists can be nested, but make sure to close all tags properly

6. File Structure Best Practices

6.1 Why File Organization Matters

Good file organization makes your project easier to maintain, share, and scale. It's like keeping a tidy workspace - you can find things faster and work more efficiently.

6.2 Recommended Folder Structure

For your Day 1 project, create this structure:

```
day-1/
■■■ index.html
■■■ images/
■■■ profile.jpg
```

6.3 Naming Conventions

Follow these rules when naming files and folders:

• Use lowercase letters only

• Use hyphens instead of spaces (profile-photo.jpg not profile photo.jpg)

• Keep names descriptive but concise

• Avoid special characters (except hyphens and underscores)

• The main page should always be named 'index.html'

> Why 'index.html'? When you visit a website folder, servers automatically look for index.html as the default page to display.

6.4 Image Management

• Store all images in an 'images' folder

• Use appropriate formats: JPG for photos, PNG for graphics with transparency

• Keep file sizes reasonable (compress large images)

• Use descriptive names: team-photo.jpg instead of img001.jpg

6.5 Creating Separate Project Folders

Create a new folder for each day's work to keep projects organized:

• day-1/ - HTML foundations project

• day-2/ - Next lesson's project

• day-3/ - And so on...

This prevents files from mixing and makes it easy to revisit previous work.

7. Assessment Questions Explained

Question 1: HTML stands for what?

Understanding: HTML stands for HyperText Markup Language. Break it down:

- **HyperText:** Text with links that connect documents

- **Markup:** Tags that give structure and meaning to content

- **Language:** A standardized way to communicate with browsers

Question 2: Which tag contains metadata and links to CSS?

Understanding: The HEAD tag contains metadata (data about data). This includes:

- Character encoding settings

- Page title shown in browser tabs

- Links to stylesheets (CSS)

- Meta descriptions for search engines

> Remember: HEAD is for the browser and search engines, BODY is for users.

Question 3: Largest heading tag?

Understanding: h1 is the largest and most important heading. Think of it like:

- h1 = Book title (only one per page)

- h2 = Chapter titles

- h3 = Section titles

- h4-h6 = Subsections (used less frequently)

Question 4: Tag used to create a link?

Understanding: The 'a' tag (anchor) creates links. It needs the 'href' attribute to specify where the link goes.

> The 'a' tag literally 'anchors' a link to text or images.

Question 5: Tag used to display an image?

Understanding: The 'img' tag embeds images. It's a self-closing tag, meaning it doesn't wrap content - it IS the content.

Required parts: src (source path) and alt (alternative text description).

Question 6: Attribute to open link in a new tab?

Understanding: target="_blank" opens links in a new tab. Always pair it with rel="noreferrer" for security.

> Why noreferrer? It prevents the new page from accessing your original page's window object - a security best practice.

Question 7: Ordered list tag?

Understanding: 'ol' creates an ordered (numbered) list. Use it when the sequence of items matters, like steps in a recipe or ranked items.

Question 8: Attribute used for image description?

Understanding: The 'alt' attribute provides alternative text for images. It's crucial for:

- Accessibility (screen readers read it aloud)

- SEO (helps search engines understand images)

- Fallback (displays if image doesn't load)

- Legal compliance (required by accessibility laws)

8. Practical Task Guidelines

Building Your Profile Page

The practical task asks you to create a one-page profile. Here's how to approach it:

Step 1: Plan Your Content

Before writing code, sketch out what you want to include:

- Your name (use h1)
- A profile photo (use img with good alt text)
- A short bio (2-3 paragraphs)
- Skills list (unordered or ordered list)
- Contact link (email or social media)

Step 2: Set Up Your File Structure

Create these folders and files:

- Main folder: day-1/
- HTML file: index.html
- Images folder: images/
- Place your profile photo in the images folder

Step 3: Start with Structure

Begin with the basic HTML structure:

- DOCTYPE declaration
- HTML tag with lang attribute
- Complete HEAD section (charset, viewport, title)
- Empty BODY section (you'll fill this next)

Step 4: Add Content Systematically

Add elements one at a time, testing as you go:

1. Add your name as an h1 heading
2. Insert your profile image with descriptive alt text
3. Write your bio in paragraph tags
4. Create a skills list (ul or ol)

5. Add a contact link

Submission Checklist - Understanding Each Requirement

1. **Valid HTML structure (doctype, head, body):**

Your page must have all three basic parts in the correct order. Open your page in a browser and check that it displays properly.

2. **At least one image with alt text:**

Include your profile photo with meaningful alt text that describes you or the image content.

3. **At least one external link that works:**

Add a link to your email, LinkedIn, GitHub, or any professional profile. Click it to verify it works!

Testing Your Work

Before submitting, verify:

- Open your HTML file in a web browser - does it display correctly?

- Click all links - do they work?

- Check your image - does it load? Does it have alt text?

- Read through your content - any typos or formatting issues?

9. Quick Reference Guide

Essential Tags Cheat Sheet

Tag	Purpose	Example
`html`	Root element	&lt;html lang="en"&gt;...&lt;/html&gt;
`head`	Contains metadata	&lt;head&gt;...&lt;/head&gt;
`body`	Contains visible content	&lt;body&gt;...&lt;/body&gt;
`h1-h6`	Headings	&lt;h1&gt;Main Title&lt;/h1&gt;
`p`	Paragraph	&lt;p&gt;Text here&lt;/p&gt;
`br`	Line break	&lt;br /&gt; or &lt;br&gt;
`a`	Link	&lt;a href="url"&gt;Text&lt;/a&gt;
`img`	Image	&lt;img src="path" alt="description" /&gt;
`ul`	Unordered list	&lt;ul&gt;&lt;li&gt;Item&lt;/li&gt;&lt;/ul&gt;
`ol`	Ordered list	&lt;ol&gt;&lt;li&gt;Item&lt;/li&gt;&lt;/ol&gt;
`li`	List item	&lt;li&gt;Item text&lt;/li&gt;

Essential Attributes Cheat Sheet

Attribute	Used With	Purpose
`href`	a (link)	Specifies the URL destination
`target`	a (link)	"_blank" opens in new tab
`rel`	a (link)	"noreferrer" for security with _blank
`src`	img	Path to the image file
`alt`	img	Description for accessibility
`lang`	html	Language of the page content
`charset`	meta	Character encoding (UTF-8)

Common Mistakes and Solutions

Mistake	Solution
Forgetting closing tags	Every opening tag needs a closing tag (except self-closing ones like img, br)
Missing alt text on images	Always include alt attribute, even if empty for decorative images
Using incorrect DOCTYPE	Always use &lt;!DOCTYPE html&gt; as the first line
Spaces in file names	Use hyphens: profile-photo.jpg not profile photo.jpg
Opening links without rel	Always add rel="noreferrer" when using target="_blank"

Multiple h1 tags	Use only ONE h1 per page for the main title

> **Remember: HTML is about structure and meaning, not appearance. Focus on using the right tags for the right content, and you'll build a solid foundation for web development!**

Good luck with your Day 1 project! Take your time, test frequently, and don't hesitate to refer back to this guide.