



## HTML Tutorial for Beginners

### Introduction

Overwhelmed by complicated frameworks or frustrated that your sites break across different devices? You are not alone! Every great website starts with HTML, or HyperText Markup Language: the structure, or “skeleton,” of the web. This beginner-friendly HTML tutorial cuts through the complexity to focus on building a solid semantic foundation for your content. Learn core concepts and gain the confidence to correctly structure any web page.

Ready to build? Download the full [HTML Course Syllabus](#) below to see how you'll go from zero to building your first structured web page!

### Why Students or Freshers Learn HTML?

Key Reasons to Learn HTML:

- **The Foundations of Web Development:** It is the entrance to front-end development, which works on the visual part of a website. A prerequisite for learning subsequent languages like CSS – for styling, and JavaScript – for interactivity.
- **Increasing Employability and Career Prospects:** Front-End Developer, Web Designer/UI/UX Designer, Content Editor/Manager, Digital Marketer. That's important, considering the modern frameworks of React and Angular rely on strong knowledge of HTML.
- **Quick and Easy for Beginners:** HTML is not a complex programming language per se, but a markup language, hence very easy to learn in a short time. This provides immediate visual feedback, which is a great motivator for the beginner and instills confidence.
- **Develops Core Technical Skills:** HTML helps in developing a more logical, or structured, way of thinking about the arrangement of information. It will increase your attention to detail since even a tiny misplaced tag can affect the layout.

Ready to convert your knowledge into a career? Here's a curated list of Top [HTML Interview Questions and Answers](#) that will help you crack your next job interview and get your first role in web development. Download Now.

Take your Knowledge  
Test Report

Check your Score



## Step-by-Step HTML Tutorial for Beginners

This is a step-by-step tutorial to learn the basics of HTML, which stands for HyperText Markup Language. By the end, you will have created your first simple web page. HTML

is the skeleton that provides structure to the content seen on the web and is, therefore, the most important language that should be learned first.

## Step 1. Installation and Setup: Your Essential Tools

You will need two things before you write your first line of code: a text editor and a web browser.

### 1.1 Choosing a Text Editor

A text editor is where you will be writing and saving your HTML files. You could use simple tools like Notepad on Windows or TextEdit on Mac, but professional editors offer features like syntax highlighting and auto-completion, which make coding a lot faster and easier.

- **Recommendation (Free and Popular): VS Code (Visual Studio Code).** It's lightweight, powerful, and the industry standard.
- **Action:** Download and install VS Code from the official website.

### 1.2 Using a Web Browser

Your web browser is the program that interprets your HTML code and displays a visual representation of that code as a web page. You probably already have one installed!

- **Action:** Make sure you have a current web browser.

### 1.3 Create Your Project Folder

It is a good practice to keep all the files belonging to a project in one place.

- **Action:** Create a new folder on your computer called MyFirstWebsite.

## Step 2. Writing Your First HTML Page

Now let's create and save your first file.

### 2.1 Create the HTML File

1. Open VS Code.
2. Go to File → New File.
3. Type the following code (don't worry about what it means yet, we'll cover that next).

```
<!DOCTYPE html>
```

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

```
<head>
```

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

```
<title>My First Web Page</title>
```

```
</head>
```

```
<body>
```

```
<h1>Hello World!</h1>
```

```
<p>This is my very first paragraph written in HTML.</p>
```

```
</body>
```

```
</html>
```

4. Go to File → Save As.
5. Go to your **MyFirstWebsite** directory.
6. Name the file *index.html* and make certain the "Save as type" is set to "HTML" or simply make sure the file name ends with the .html extension.

## 2.2 View Your Page in the Browser

1. Open your MyFirstWebsite folder.
2. Double-click the index.html file.
3. It will open a file in your default web browser with “Hello World!” as a large heading and a small paragraph underneath.

## Step 3. Understanding HTML structure and tags

Every HTML page has a required basic structure built from elements or tags.

### 3.1 Tags, Elements, and Attributes

- **Tag:** The set of instructions that defines an element. Tags are enclosed in angle brackets, such as `<p>`.
- **Element:** The opening tag followed by the content and a closing tag. Example: `<p>My content</p>`. Most tags do have a closing version, but some tags self-close like `<br>` for line breaks.
- **Attribute:** This specifies additional information about an element. These are written in the opening tag and include a name and a value (for example, ``).

### 3.2 The Boilerplate Code Explained

Let’s break down the code you just wrote:

Code	Explanation
<code>&lt;!DOCTYPE html&gt;</code>	Declaration. Tells the browser which version of HTML it’s reading (in this case, HTML5).
<code>&lt;html&gt;...&lt;/html&gt;</code>	The root element that wraps all other content on the page.

<code>&lt;head&gt;...&lt;/head&gt;</code>	Contains metadata—information about the page that isn't displayed on the page itself (like the title, character set, etc.).
<code>&lt;meta charset="UTF-8"&gt;</code>	Specifies the character encoding (standard for most languages).
<code>&lt;title&gt;...&lt;/title&gt;</code>	The text that appears in the browser tab or window title.
<code>&lt;body&gt;...&lt;/body&gt;</code>	Contains visible content of the web page (headings, paragraphs, images, links).

## Step 4. Core HTML Elements (The Content Tags)

The `<body>` section is where you place all the content the user sees. Here are the most essential content tags:

### 4.1 Headings: `<h1>` to `<h6>`

Headings are a way of structuring and outlining your content. `<h1>` is the most important `<h6>`; it's the main title, and `h6` is the least important. You should only have one `<h1>` on a page.

```
<h1>Main Title of the Page</h1>
```

```
<h2>Section Header</h2>
```

```
<h3>Sub-Section Topic</h3>
```

### 4.2 Paragraphs: `<p>`

Used for standard blocks of text.

*<p>This is a paragraph of text. It will automatically wrap based on the size of the browser window.</p>*

### **4.3 Lists: <ul>, <ol>, and <li>**

Used to format items in a reader-friendly style.

- **Unordered List (<ul>):** Items are listed with bullets.
- **Ordered List (<ol>):** Items are numbered.
- **List Item (<li>):** Defines each item in the list.

*<h3>My Favorite Fruits (Unordered)</h3>*

*<ul>*

*<li>Apples</li>*

*<li>Bananas</li>*

*<li>Oranges</li>*

*</ul>*

*<h3>Steps for Making Tea (Ordered)</h3>*

*<ol>*

*<li>Boil Water</li>*

*<li>Steep Tea Bag</li>*

*<li>Add Milk and Sugar (Optional)</li>*

*</ol>*

## Step 5. Links and Images: Making Your Site Interactive

HTML really shines when you connect content together.

### 5.1 Hyperlinks: <a>

The anchor element, <a>, is used to create a link to another page or resource. It requires the href attribute, which means Hypertext Reference and is used to specify the destination.

```
<p>Visit the official <a href="https://www.google.com">Google</a> website.</p>
```

```
<p><a href="about.html">Learn more about us</a></p>
```

- **Target Attribute:** Target attribute to force a link to open in a new tab:  
target="\_blank"

```
<a href="https://www.google.com" target="_blank">Google (Opens in new tab)</a>
```

### 5.2 Images: <img>

The image element (img) is used to display pictures. It is a self-closing tag and requires two key attributes:

- **src (Source):** The path or URL of the image file.
- **alt text:** This provides a description of the image which screen readers and search engines then read (important for accessibility and SEO).

```

```

**Note:** The flower.jpg image file has to be in the same folder where your index.html file lives, for this to work.

## Step 6. Semantic and Structural Elements

HTML5 introduced semantic tags, which give meaning to the content they contain, helping search engines and developers understand the page's structure better.

Tag	Purpose
<header>	The introductory content for a page or a section (often contains navigation and the site logo).
<nav>	A section containing navigation links (like the main menu).
<main>	The primary, unique content of the document (there should only be one per page).
<article>	An independent, self-contained piece of content (like a blog post, news story, or forum comment).
<section>	A generic thematic grouping of content, often with a heading.
<footer>	The content at the bottom of the page or a section (often contains copyright info, contact details, or related links).

## 6.1 A More Structured Example

Let's structure the content of Step 2 with semantic tags:

```
<!DOCTYPE html>
```

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

```
<head>
```

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

```
  <title>My Structured Page</title>
```

```
</head>
```

```
<body>
```

```
  <header>
```

```
    <h1>Welcome to My Blog</h1>
```

```
    <nav>
```

```
      <ul>
```

```
        <li><a href="index.html">Home</a></li>
```

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

```
        <li><a href="contact.html">Contact</a></li>
```

```
      </ul>
```

```
    </nav>
```

```
  </header>
```

```
  <main>
```

```
<article>
```

```
  <h2>First Post Title</h2>
```

```
  <p>This is the content of my first blog post. It sits within an article tag.</p>
```

```
</article>
```

```
<section>
```

```
  <h2>Related Topics</h2>
```

```
  <p>Content related to the main article goes here.</p>
```

```
</section>
```

```
</main>
```

```
<footer>
```

```
  <p>&copy; 2024 My Website. All rights reserved.</p>
```

```
</footer>
```

```
</body>
```

```
</html>
```

## Step 7. Tables and Forms (Advanced Basics)

### 7.1 Creating a Table: `<table>`

Tables are used to display data in rows and columns.

Tag	Purpose
<table>	The container for the entire table.
<thead>	The container for the row(s) that define the column headers.
<tbody>	The container for the main data rows.
<tr>	Table Row. Defines a row in the table.
<th>	Table Header. Used for column headings (text is bold and centered by default).
<td>	Table Data. Defines a standard data cell.

<table>

<thead>

<tr>

<th>Product</th>

<th>Price</th>

</tr>

</thead>

<tbody>

<tr>

```
<td>Laptop</td>

<td>$1200</td>

</tr>

<tr>

<td>Monitor</td>

<td>$300</td>

</tr>

</tbody>

</table>
```

## 7.2 Form creation: `<form>` and `<input>`

Forms are an important means whereby user data is solicited or retrieved, such as logins, contact forms, and questionnaires.

- **`<form>`**: This is the container for all form elements. The `action` attribute indicates where the data should be sent.
- **`<label>`**: Gives a user-friendly description to input field.
- **`<input>`**: The most usual form element. The `type` attribute changes its behavior (e.g., text, email, password, submit).

```
<form action="/submit-data-url" method="post">
```

```
<div>
```

```
<label for="name">Your Name:</label>
```

```
<input type="text" id="name" name="user_name">
```

```
</div>
```

```
<div>
```

```
<label for="email">Your Email:</label>
```

```
<input type="email" id="email" name="user_email">
```

```
</div>
```

```
<input type="submit" value="Send Message">
```

```
</form>
```

By now, you have learned how to structure your content with HTML. However, your page is still unstyled, it looks plain.

Here's where CSS, or Cascading Style Sheets, comes in.

- HTML is for structure and content.
- Presentation and style in CSS refers to colors, fonts, and layout.

You are now ready to embark on the journey of styling your content, which is the next natural step after mastering HTML. Way to build your first website structure!

That's all well and good in theory, but practice makes perfect. The best way to cement your HTML knowledge is by completing practical tasks. Download our Complete [HTML Challenges and Solutions](#) Guide!

## Real Time Examples for HTML Tutorial for Learners

HTML is not theoretical; it's the living structure behind everything you see and interact with online. Here are some practical, real-world examples that demonstrate the importance of pure HTML:

## The Blog Post Structure

- When you read an article on a website or blog, its structure is defined through HTML.
- Notice above that the main title is wrapped in an `<h1>` tag, subheadings use `<h2>` or `<h3>`, the body text is in `<p>` tags, and images are embedded using the `<img>` tag, with *alt* text describing the image.
- **Goal:** Ensures that the content is clear, readable, and properly indexed by the search engines.

## Navigation Menus (`<nav>`)

- Navigation is used on every website to help users get around: Home, About Us, Contact, etc.
- HTML's `<nav>` element provides an unordered list of links (`<ul>` of links `<a>`), semantically grouping them.
- **Goal:** Offers an accessible map of the website, to made visually appealing with CSS.

## Online Registration or Login Forms

- Any time you sign up for a service you will use an HTML form.
- The `<form>` tag will hold all the input elements; for example, `<input type="text">`, `<input type="email">`, `<input type="password">`, among others, and `<label>`.
- **Goal:** Defines precisely what kind of fields are needed to collect user data, and which action is to be taken when the user clicks the last `<input type="submit">` button.

Ready to build your first real project? Download our list of [HTML Project Ideas](#) tailored for beginners to apply these concepts and build a portfolio!

## FAQs About HTML Tutorial for Beginners

### 1. How to learn HTML for beginners?

Begin with the basic structure (`<!DOCTYPE html>`, `<html>`, `<head>`, `<body>`). Focus on common tags like headings (`<h1>`), paragraphs (`<p>`), links (`<a>`), and images (`<img>`). Use HTML tutorials and a code editor like VS Code. Practice by building simple web pages.

### 2. How to do HTML step by step?

1. Setup Code Editor (VS Code). 2. Create an index.html file. 3. Write the basic HTML boilerplate. 4. Add content using semantic tags: `<header>`, `<main>`, `<footer>`. 5. Open the file in your browser and see your result. Practice consistently.

### 3. What are the 20 tags in HTML?

The most essential 20 tags in HTML include structural (`<html>`, `<head>`, `<body>`, `<div>`, `<span>`), content (`<h1>`-`<h6>`, `<p>`, `<ul>`, `<li>`), media/links (`<a>`, `<img>`), forms (`<form>`, `<input>`, `<button>`, `<label>`), and semantic (`<header>`, `<nav>`, `<main>`, `<footer>`).

### 4. Can I learn HTML in 7 days?

Yes, you can learn basic syntax and most-used core tags in 7 days, putting in a few hours per day. HTML is a markup language, not a complex programming language in which true proficiency and semantic best practices will take longer than a week to learn.

### 5. Who is the king of CSS?

There isn't a single known "King of CSS," because it's a pretty subjective title. Still, Eric Meyer and Chris Coyier are very frequently mentioned among the most influential people in the CSS community due to their foundational work, educational resources, and contributions to web standards.

### 6. What is HTML basic code?

The basic or boilerplate code is the minimal, required structure for any HTML5 document:

```
<!DOCTYPE html>
<html>
<head>
  <title>Page Title</title>
</head>
<body>
  </body>
</html>
```

### **7. Is Python or HTML harder?**

Generally, Python is considered more challenging than HTML. HTML is a markup language and is used for the structure of content; it's pretty straightforward. Python is a full programming language that involves logic, variables, loops, and data structures.

### **8. Is 2 hours a day enough to learn coding?**

Yes, 2 hours a day is a great, sustainable amount for learning to code. Consistency is more important than duration. Focused practice, building small projects, and daily review of concepts will definitely be highly effective in building muscle memory and retaining information within 2 hours. Explore [HTML salary for freshers](#).

### **9. What is the difference between HTML and CSS?**

HTML gives the skeleton of the web page, providing the structure and content. CSS, Cascading Style Sheets, does the presentation and style: the appearance, colors, fonts, and layout. They come together to make a complete web page.

### **10. What is the C symbol in HTML?**

The C symbol in HTML is generally represented by © in the source code; it's the HTML entity for the copyright symbol (`&#169;`). It is used within footers to denote ownership rights and shows as © in the browser.

## **Conclusion**

You've conquered the skeleton of the web! You have grasped the overall structure, the basic syntax, and the must-know tags for every single website. Remember, HTML is the

indispensable foundation; every front-end skill you will learn next will be based on this knowledge. Do not stop here; the next logical step is learning CSS to bring color and life into your structures.

Ready to start designing and styling your pages? Want to go from structure to professional design and layout? Enroll now in our full CSS and [HTML course in Chennai!](#)