**HTML**

**DEVELOPER**

**NOTES**

**👨🏻‍💻 Name: Mr. Rupesh Lal**

**🌐 Website: www.itmodem.com**

**📞 WhatsApp: +91 7870886355**

**💼 Profession: Software Engineer**

**🧱 HTML (HyperText Markup Language)**

**🔰 Section 1: Introduction to HTML**

* What is HTML?
* HTML History & Versions
* Structure of an HTML Document
* Basic HTML Tags (<html>, <head>, <body>, etc.)
* File Extension & How to Run HTML

**🔡 Section 2: Text & Formatting Tags**

* Headings (<h1> to <h6>)
* Paragraph (<p>)
* Line Breaks and Horizontal Rule (<br>, <hr>)
* Bold, Italic, Underline (<b>, <i>, <u>)
* Superscript & Subscript (<sup>, <sub>)

**🔗 Section 3: Links and Anchors**

* Creating Hyperlinks (<a>)
* Target Attribute
* Internal vs External Linking
* Bookmark Links

**🖼️ Section 4: Images**

* Inserting Images (<img>)
* src, alt, width, height attributes
* Image Formats (JPG, PNG, SVG, etc.)

**📝 Section 5: Lists**

* Ordered Lists (<ol>)
* Unordered Lists (<ul>)
* Description Lists (<dl>, <dt>, <dd>)

**🧩 Section 6: Tables**

* Creating Tables (<table>, <tr>, <td>)
* Table Head and Body
* rowspan and colspan
* Table Styling (with CSS)

**📦 Section 7: Forms**

* <form> Basics
* Input Types (text, radio, checkbox, password, etc.)
* <textarea>, <select>, <option>, <label>
* Form Attributes (action, method, etc.)
* Form Validation Basics (HTML5)

**🏷️ Section 8: Semantic HTML**

* Importance of Semantic Tags
* Common Semantic Elements: <header>, <footer>, <nav>, <main>, <section>, <article>, <aside>

**🧠 Section 9: Advanced HTML Topics**

* iframe
* Audio & Video Embedding
* Meta Tags (<meta>, viewport)
* SEO Basics in HTML
* Accessibility (alt text, aria-\* attributes)
* HTML Entities

**🧱 HTML (HyperText Markup Language)**

**🔰 Chapter 1: Introduction to HTML**

**📌 What is HTML?**

* **HTML** stands for **HyperText Markup Language**.
* It is the **standard markup language** used to create and design **web pages**.
* It **structures the content** on the web (headings, paragraphs, links, images, etc.).
* HTML uses **tags** enclosed in **angle brackets** (e.g., <p>, <h1>, etc.).
* It is **not a programming language**, but a **markup language**.

**🕰️ HTML History & Versions**

| **Version** | **Description** |
| --- | --- |
| HTML 1.0 | Initial version (1993), very limited features |
| HTML 2.0 | Introduced basic features like forms (1995) |
| HTML 3.2 | Added support for scripting (1997) |
| HTML 4.01 | Improved CSS support and structure (1999) |
| XHTML | Stricter XML-based version of HTML |
| **HTML5** | Current major version (2014+), supports audio, video, canvas, and semantic tags |

**🧱 Structure of an HTML Document**

<!DOCTYPE html>

<html>

<head>

<title>Page Title</title>

</head>

<body>

<h1>Main Heading</h1>

<p>This is a paragraph.</p>

</body>

</html>

* <!DOCTYPE html> – Declares the document as HTML5.
* <html> – Root element of the HTML page.
* <head> – Contains metadata (title, styles, scripts).
* <body> – Contains visible content shown to users.

**🏷️ Basic HTML Tags**

| **Tag** | **Purpose** |
| --- | --- |
| <html> | Root of the HTML document |
| <head> | Metadata, title, and links |
| <title> | Title of the page (seen in browser tab) |
| <body> | Main content of the webpage |
| <h1>–<h6> | Headings, <h1> is largest |
| <p> | Paragraph |
| <a> | Hyperlink |
| <img> | Image insertion |
| <br> | Line break |
| <hr> | Horizontal rule |

**💡 File Extension & How to Run HTML**

* HTML files are saved with the extension **.html** or **.htm**.
* You can **run** an HTML file by:
  1. Creating the file in a text editor (VS Code, Notepad, etc.).
  2. Saving it as index.html.
  3. Double-clicking the file or opening it in a browser like Chrome or Firefox.

🔡 **Chapter 2: Text & Formatting Tags**

**🔡 Section 2: Text & Formatting Tags**

HTML provides various tags to structure and style **text content**. These tags define how the text appears on the web page.

**🏷️ Headings (<h1> to <h6>)**

* HTML provides six levels of headings:
  + <h1>: Largest and most important heading
  + <h6>: Smallest heading

<h1>This is H1</h1>

<h2>This is H2</h2>

<h3>This is H3</h3>

<h4>This is H4</h4>

<h5>This is H5</h5>

<h6>This is H6</h6>

✅ Use only **one <h1>** per page for SEO and accessibility.

**📑 Paragraph (<p>)**

* Used to define **a block of text** (a paragraph).
* Automatically adds space above and below the text.

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

**↩️ Line Breaks (<br>) and Horizontal Rule (<hr>)**

* <br>: Adds a **line break** (moves content to the next line).
  + It is a **self-closing tag**.

<p>Line one<br>Line two</p>

* <hr>: Creates a **horizontal line** to separate content.

<p>Above the line</p>

<hr>

<p>Below the line</p>

**✍️ Text Formatting Tags**

| **Tag** | **Description** | **Example Code** | **Output Example** |
| --- | --- | --- | --- |
| <b> | Bold text | <b>Bold Text</b> | **Bold Text** |
| <i> | Italic text | <i>Italic Text</i> | *Italic Text* |
| <u> | Underlined text | <u>Underlined Text</u> | Underlined Text |

**📐 Superscript (<sup>) & Subscript (<sub>)**

* <sup>: Renders text **above the baseline** (e.g., exponents)
* <sub>: Renders text **below the baseline** (e.g., chemical formulas)

<p>Water formula: H<sub>2</sub>O</p>

<p>Area of square: a<sup>2</sup></p>

Output:

* Water formula: H₂O
* Area of square: a²

🔗 **Chapter 3: Links and Anchors**

**🔗 Section 3: Links and Anchors**

HTML uses the <a> tag (anchor) to create **hyperlinks** that connect web pages, files, emails, or specific sections of the same page.

**🔗 Creating Hyperlinks (<a>)**

* The <a> tag is used to create a **clickable link**.
* Uses the href attribute to define the destination URL.

<a href="https://www.google.com">Visit Google</a>

Output: [Visit Google](https://www.google.com/)

**🎯 Target Attribute**

* The target attribute defines **how the link opens**.

| **Value** | **Description** |
| --- | --- |
| \_self | Opens in the **same tab** (default) |
| \_blank | Opens in a **new tab/window** |
| \_parent | Opens in the parent frame |
| \_top | Opens in the full body of the window |

<a href="https://www.google.com" target="\_blank">Open in new tab</a>

Output: [Open in new tab](https://www.google.com/)

**🧭 Internal vs External Linking**

| **Type** | **Description** | **Example** |
| --- | --- | --- |
| **Internal** | Links to another page **within** the website | <a href="about.html">About Us</a> |
| **External** | Links to a page **outside** the website | <a href="https://github.com">GitHub</a> |

**🔖 Bookmark Links (Page Anchors)**

* You can link to a **specific part** of the same page using id.

**1. Add an id to the target element:**

<h2 id="contact">Contact Section</h2>

**2. Create a link to that section:**

<a href="#contact">Go to Contact Section</a>

When clicked, the page scrolls to the element with id="contact".

🖼️ **Chapter 4: Images**

**🖼️ Section 4: Images**

HTML uses the <img> tag to **display images** on a webpage. Images make content **visually appealing** and help in storytelling or branding.

**🖼️ Inserting Images (<img>)**

* The <img> tag is a **self-closing tag**.
* It requires at least the src and alt attributes.

<img src="images/photo.jpg" alt="Beautiful scenery">

✅ This will show an image named photo.jpg from the images folder.

**🛠️ Common Attributes of <img>**

| **3Attribute** | **Description** |
| --- | --- |
| src | Specifies the **image file path** or URL |
| alt | Alternative text for accessibility or when image fails to load |
| width | Width of the image (in px or %) |
| height | Height of the image (in px or %) |

<img src="dog.png" alt="Cute dog" width="300" height="200">

ℹ️ You can control dimensions using either HTML attributes or **CSS**.

**🖼️ Image Formats**

| **Format** | **Full Form** | **Usage/Description** |
| --- | --- | --- |
| .jpg | JPEG | Good for **photographs**, compressed, slightly lossy |
| .png | Portable Network Graphics | Supports **transparency**, ideal for **logos and icons** |
| .gif | Graphics Interchange Format | Supports **animation**, limited to 256 colors |
| .svg | Scalable Vector Graphics | Resolution-independent, best for **icons and illustrations** |
| .webp | Web Picture Format | Modern format offering **high quality + compression**, supported in major browsers |

**✅ Best Practices for Using Images**

* Always include **alt** text for accessibility and SEO.
* Use optimized images for faster loading.
* Use **relative paths** (images/pic.jpg) for local files and **absolute URLs** (https://example.com/pic.jpg) for web-hosted images.
* Prefer **SVG** for icons and **WebP** for modern web optimization.

📝 **Chapter 5: Lists**

**📝 Section 5: Lists**

HTML provides three types of lists to **organize content** in a structured, readable way: ordered, unordered, and description lists.

**🔢 Ordered Lists (<ol>)**

* Used when the **order or sequence matters** (e.g., steps, rankings).
* Each list item is wrapped inside an <li> (list item) tag.

<ol>

<li>Wake up</li>

<li>Brush your teeth</li>

<li>Have breakfast</li>

</ol>

Output:

1. Wake up
2. Brush your teeth
3. Have breakfast

🛠️ **Customizing Ordered Lists:**

<ol type="A" start="3">

<li>Item One</li>

<li>Item Two</li>

</ol>

* type="1", "A", "a", "I", "i" changes numbering style
* start="3" sets the starting number/letter

**🔘 Unordered Lists (<ul>)**

* Used when the **order doesn’t matter** (e.g., features, groceries).
* Also uses <li> tags.

<ul>

<li>HTML</li>

<li>CSS</li>

<li>JavaScript</li>

</ul>

Output:

* HTML
* CSS
* JavaScript

🛠️ **Customizing Bullets with CSS:**

ul {

list-style-type: square;

}

Values: disc (default), circle, square, none

**📘 Description Lists (<dl>, <dt>, <dd>)**

* Used to define **terms and their descriptions** (e.g., glossaries, FAQs).

<dl>

<dt>HTML</dt>

<dd>A markup language for creating web pages.</dd>

<dt>CSS</dt>

<dd>A stylesheet language for designing web pages.</dd>

</dl>

Output:  
**HTML**  
A markup language for creating web pages.  
**CSS**  
A stylesheet language for designing web pages.

**✅ Summary Table**

| **List Type** | **Tags Used** | **When to Use** |
| --- | --- | --- |
| Ordered List | <ol>, <li> | When order/sequence matters |
| Unordered List | <ul>, <li> | For unordered items/bullet lists |
| Description List | <dl>, <dt>, <dd> | For definitions, FAQs, glossaries |

🧩 **Chapter 6: Tables**

**🧩 Section 6: Tables**

Tables in HTML are used to **display structured data** in rows and columns. HTML provides a set of tags to build and customize tables.

**📐 Creating Tables (<table>, <tr>, <td>)**

| **Tag** | **Description** |
| --- | --- |
| <table> | Defines the **entire table** |
| <tr> | Defines a **table row** |
| <td> | Defines a **table data cell** |

<table>

<tr>

<td>Row 1, Cell 1</td>

<td>Row 1, Cell 2</td>

</tr>

<tr>

<td>Row 2, Cell 1</td>

<td>Row 2, Cell 2</td>

</tr>

</table>

**🧠 Table Head and Body**

* Use <thead> for the header section
* Use <tbody> for the body section
* Use <th> for header cells (bold & centered by default)

<table border="1">

<thead>

<tr>

<th>Name</th>

<th>Age</th>

</tr>

</thead>

<tbody>

<tr>

<td>Alice</td>

<td>22</td>

</tr>

<tr>

<td>Bob</td>

<td>25</td>

</tr>

</tbody>

</table>

Output: A table with headers “Name” and “Age”.

**↔️ rowspan and colspan**

* rowspan merges **cells vertically**
* colspan merges **cells horizontally**

<table border="1">

<tr>

<th rowspan="2">Name</th>

<th colspan="2">Marks</th>

</tr>

<tr>

<td>Math</td>

<td>Science</td>

</tr>

<tr>

<td>Alice</td>

<td>85</td>

<td>90</td>

</tr>

</table>

**🎨 Table Styling (with CSS)**

Use CSS to improve the appearance of tables:

<style>

table {

border-collapse: collapse;

width: 100%;

}

th, td {

padding: 10px;

border: 1px solid #ccc;

text-align: left;

}

th {

background-color: #f2f2f2;

}

tr:hover {

background-color: #f9f9f9;

}

</style>

**✅ Best Practices**

* Always use <th> for headers.
* Use scope="col" or scope="row" for accessibility.
* Use border-collapse: collapse; for cleaner borders.

📦 **Chapter 7: Forms**

**📦 Section 7: Forms**

HTML forms are used to **collect user input** and send it to a server for processing.

**🧱 <form> Basics**

* A <form> contains form controls like text fields, checkboxes, buttons, etc.
* It has two important attributes:
  + action – URL where form data is sent
  + method – HTTP method (GET or POST)

<form action="/submit" method="post">

<!-- form elements go here -->

</form>

**🧾 Input Types**

| **Input Type** | **Description** | **Example** |
| --- | --- | --- |
| text | Single-line text input | <input type="text"> |
| password | Hides characters | <input type="password"> |
| email | Validates email input | <input type="email"> |
| number | Numeric input only | <input type="number"> |
| checkbox | Multiple selection option | <input type="checkbox"> |
| radio | Single selection from a group | <input type="radio"> |
| submit | Submits the form | <input type="submit"> |
| reset | Resets form fields | <input type="reset"> |
| date | Date picker | <input type="date"> |
| file | Upload file | <input type="file"> |

<input type="text" placeholder="Enter your name">

**✍️ <textarea>, <select>, <option>, <label>**

* **<textarea>**: Multi-line text input

<textarea rows="4" cols="30">Enter your message...</textarea>

* **<select> and <option>**: Drop-down menu

<select name="city">

<option value="delhi">Delhi</option>

<option value="mumbai">Mumbai</option>

</select>

* **<label>**: Connects a label to a form input using for and id

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

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

**⚙️ Form Attributes**

| **Attribute** | **Description** |
| --- | --- |
| action | URL to send form data to |
| method | HTTP method (get or post) |
| autocomplete | Enables/disables autofill (on/off) |
| target | Where to open response (\_blank, \_self, etc.) |

<form action="/submit" method="post" autocomplete="off" target="\_blank">

**✅ Form Validation Basics (HTML5)**

HTML5 provides **built-in validation** using attributes:

| **Attribute** | **Purpose** |
| --- | --- |
| required | Field must be filled |
| minlength | Minimum number of characters |
| maxlength | Maximum number of characters |
| pattern | Regular expression match |
| type | Ensures valid format (email, url) |

<input type="email" required>

<input type="text" minlength="3" maxlength="10">

<input type="text" pattern="[A-Za-z]+" title="Only letters allowed">

**🧪 Example Form**

<form action="/submit" method="post">

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

<input type="text" id="name" name="name" required>

<br><br>

<label for="message">Message:</label><br>

<textarea id="message" name="message" rows="4" cols="30"></textarea>

<br><br>

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

</form>

🏷️ **Chapter 8: Semantic HTML**

**🏷️ Section 8: Semantic HTML**

**Semantic HTML** refers to HTML elements that clearly describe their **meaning and purpose** in a web page — both to browsers and developers.

**🌟 Importance of Semantic Tags**

✅ Improves **readability** and **code structure**  
✅ Enhances **SEO** (Search Engine Optimization)  
✅ Boosts **accessibility** for screen readers  
✅ Easier **maintenance and collaboration**

🧠 Instead of using generic <div> tags everywhere, semantic tags provide **contextual meaning**.

**📦 Common Semantic Elements**

| **Tag** | **Description** |
| --- | --- |
| <header> | Represents the top section of a page or section (e.g., logo, nav, title) |
| <footer> | Defines the bottom section (e.g., contact, copyright) |
| <nav> | Holds the **navigation links** for the website |
| <main> | Represents the **primary content** of the document |
| <section> | A **thematic grouping** of content (e.g., services, about us) |
| <article> | Independent, self-contained content (e.g., blog post, news) |
| <aside> | Side content like ads, tips, or sidebars |

**🧩 Example: Semantic HTML Layout**

<!DOCTYPE html>

<html>

<head>

<title>Semantic HTML Example</title>

</head>

<body>

<header>

<h1>My Portfolio</h1>

<nav>

<a href="#home">Home</a>

<a href="#projects">Projects</a>

</nav>

</header>

<main>

<section id="about">

<h2>About Me</h2>

<p>I am a web developer...</p>

</section>

<article>

<h2>Latest Blog Post</h2>

<p>This is a sample article...</p>

</article>

<aside>

<h3>Quick Tips</h3>

<p>Use semantic tags for better SEO!</p>

</aside>

</main>

<footer>

<p>© 2025 Mr. Rupesh Lal | All rights reserved.</p>

</footer>

</body>

</html>

**✅ Best Practices**

* Don’t overuse <div> and <span> when semantic tags are available.
* Use <section> for grouping related content.
* Use <article> for reusable, standalone pieces.
* Always include at least one <main> tag for accessibility.

🧠 **Chapter 9: Advanced HTML Topics**

**🧠 Section 9: Advanced HTML Topics**

This section introduces powerful and essential features of HTML used in **media embedding**, **metadata**, **SEO**, and **accessibility**.

**🖼️ <iframe> – Embedding External Content**

* Embeds another HTML page inside the current one.
* Commonly used to embed YouTube videos, maps, or other websites.

<iframe src="https://www.example.com" width="600" height="400"></iframe>

✅ Attributes:

* src: URL of the embedded page
* width & height: dimensions
* frameborder, allowfullscreen, loading

⚠️ Tip: Not all websites allow embedding due to **CORS** or **security policies**.

**🎵 Audio & 🎬 Video Embedding**

**🔊 <audio> Tag**

<audio controls>

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

Your browser does not support the audio element.

</audio>

**🎥 <video> Tag**

<video width="600" controls>

<source src="video.mp4" type="video/mp4">

Your browser does not support the video tag.

</video>

✅ Attributes:

* controls: shows play/pause buttons
* autoplay, loop, muted, poster

**🧠 Meta Tags (<meta>)**

Meta tags provide metadata about the HTML document (in the <head> section).

<meta charset="UTF-8">

<meta name="description" content="Personal Portfolio of Rupesh Lal">

<meta name="keywords" content="HTML, CSS, MERN, Web Development">

<meta name="author" content="Rupesh Lal">

✅ Important for:

* SEO
* Browser rendering
* Character encoding

**📱 Viewport Meta Tag**

Ensures proper scaling and responsiveness on mobile devices.

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

Must-have for **responsive design**.

**🔍 SEO Basics in HTML**

* Use **semantic tags** (<header>, <main>, <section>, <article>)
* Add **alt text** for all images
* Write descriptive **<title>** and <meta> descriptions
* Use **heading hierarchy** properly (<h1> to <h6>)

<title>Best Frontend Developer in India | ItModem</title>

**♿ Accessibility (A11y)**

Accessibility ensures your website is usable by people with disabilities.

✅ Key Features:

* alt attribute for images
* label elements for inputs
* tabindex for keyboard navigation
* aria-\* attributes for screen readers

<img src="user.jpg" alt="Profile photo of Rupesh Lal">

<input aria-label="Search field" type="text">

**🔠 HTML Entities**

HTML entities are used to display special characters in HTML.

| **Character** | **Entity Code** | **Output** |
| --- | --- | --- |
| < | &lt; | < |
| > | &gt; | > |
| & | &amp; | & |
| " | &quot; | " |
| ' | &apos; | ' |
| © | &copy; | © |
| ® | &reg; | ® |
| ₹ | &#8377; | ₹ |

**✅ Summary**

This section enables you to:

* Embed **multimedia** & third-party content
* Write **SEO-friendly** and **accessible** HTML
* Handle **special characters** with ease

**🧪 HTML Practice Exercises**

**🌱 Beginner Level**

1. **HTML Skeleton Page**
   * Create a basic HTML page using:
   * <!DOCTYPE html>
   * <html>
   * <head>
   * <title>Your Name</title>
   * </head>
   * <body>
   * <h1>Welcome to My Page</h1>
   * <p>This is my first HTML document.</p>
   * </body>
   * </html>
2. **Text Formatting Page**
   * Use these tags: <h1> to <h6>, <p>, <b>, <i>, <u>, <sup>, <sub>, <br>, <hr>.
3. **Create a Hyperlink Page**
   * Add:
     + A link to Google (external)
     + A link to about.html (internal)
     + A bookmark link that scrolls to a “Contact” section
4. **Image Showcase**
   * Add three images using <img> with src, alt, width, and height.
5. **List Practice**
   * Create:
     + One ordered list
     + One unordered list
     + One description list (with <dl>, <dt>, <dd>)

**🧰 Intermediate Level**

1. **Table Creation**
   * Create a student marks table with rowspan, colspan, <thead>, <tbody>, <th>, and <td>.
2. **Form Design**
   * Create a form with:
     + Name, Email, Password fields
     + Radio buttons for gender
     + Checkboxes for skills
     + Dropdown to choose a city
     + A submit button
3. **Semantic Layout**
   * Build a layout using only semantic tags:
   * <header></header>
   * <nav></nav>
   * <main>
   * <section></section>
   * <article></article>
   * <aside></aside>
   * </main>
   * <footer></footer>
4. **Embed Media**
   * Use:
     + <audio> with controls and a sample MP3
     + <video> with controls and sample MP4
     + <iframe> for embedding YouTube
5. **Meta & SEO Tags**

* Create a basic HTML page that includes:
  + <meta charset="UTF-8">
  + <meta name="description">
  + <meta name="keywords">
  + <meta name="author">
  + <title> tag with relevant SEO-friendly text