**HTML Study checklist**

### **✅ HTML Basics**

* Understand the structure of an HTML document (<!DOCTYPE html>, <html>, <head>, <body>)
* Learn HTML syntax and basic tags (<h1> to <h6>, <p>, <br>, <hr>, <strong>, <em>, <small>, <mark>)
* Use HTML comments (<!-- Comment -->)
* Understand the difference between block-level and inline elements

### **✅ HTML Elements & Attributes**

* Properly use common elements (<div>, <span>, <pre>, <code>, <blockquote>, <cite>)
* Learn about HTML attributes (global attributes like id, class, title, style, hidden, data-\*)
* Use **semantic elements** (<header>, <nav>, <section>, <article>, <aside>, <footer>)

### **✅ HTML Forms & Inputs**

* Create basic forms using <form>, <input>, <label>, <textarea>, <select>, <option>, <button>, <fieldset>, <legend>
* Use different input types (text, email, password, number, radio, checkbox, date, file, color, etc.)
* Implement **form validation** using attributes like required, pattern, maxlength, min/max, step
* Understand **form attributes** (action, method, enctype, target, novalidate, autocomplete)
* Use the <datalist> and <output> elements for enhanced user input

### **✅ HTML Tables**

* Create tables using <table>, <tr>, <td>, <th>, <caption>
* Apply attributes like colspan and rowspan
* Use <thead>, <tbody>, <tfoot> for better structure
* Style tables with CSS for better readability

### **✅ HTML Lists**

* Understand **unordered lists** (<ul> and <li>)
* Understand **ordered lists** (<ol> and <li>)
* Use **definition lists** (<dl>, <dt>, <dd>)

### **✅ HTML Links & Navigation**

* Use <a> for internal (href="#section") and external links (target="\_blank")
* Implement download attribute in <a> for file downloads
* Create navigation menus using <nav> and CSS

### **✅ HTML Images & Multimedia**

* Insert images using <img> with src, alt, width, height
* Use <figure> and <figcaption> for image descriptions
* Implement audio (<audio>) and video (<video>) with controls, autoplay, loop, muted
* Embed YouTube videos using <iframe>

### **✅ HTML Meta & SEO**

* Set page metadata using <meta> (charset, viewport, author, description, keywords)
* Use proper **title tags** (<title>)
* Implement **Open Graph (OG) meta tags** for social media sharing

### **✅ HTML Accessibility (A11Y)**

* Use **alt text** for images
* Ensure form labels (<label for="id">)
* Implement keyboard navigation support
* Add role attributes for ARIA landmarks (role="navigation", role="banner", etc.)

### **✅ HTML Graphics & Canvas**

* Use <canvas> for graphics rendering
* Create SVG graphics using <svg>, <circle>, <rect>, <line>, <text>
* Use the <map> and <area> elements for image maps

### **✅ HTML APIs & Advanced Topics**

* Understand **Geolocation API**
* Learn **Drag and Drop API**
* Implement **Web Storage API** (localStorage, sessionStorage)
* Understand **HTML Web Workers**
* Use <template> and <slot> for reusable HTML structures

### **✅ HTML Best Practices**

* Follow proper indentation and formatting
* Use meaningful element names (avoid <div> soup)
* Ensure proper nesting of elements
* Use **relative paths** for linking internal resources
* Validate HTML using **W3C Validator** (<https://validator.w3.org/>)

========================================================================

### **CSS Study Checklist**

This **CSS checklist** covers all important aspects from **basic to advanced topics**, ensuring a solid understanding of **Cascading Style Sheets (CSS)**.

## **✅ 1. CSS Basics**

* Understand **CSS Syntax** (selector { property: value; })
* Use **internal (<style>)**, **external (.css file)**, and **inline styles (style="color:red;")**
* Understand the **difference between HTML and CSS**

## **✅ 2. CSS Selectors**

* Universal Selector (\*)
* Element Selectors (h1, p, div)
* Class Selectors (.classname)
* ID Selector (#idname)
* Attribute Selectors ([type="text"], [href^="https"])
* Grouping Selectors (h1, h2, h3 { color: red; })
* Pseudo-classes (:hover, :focus, :nth-child(), :first-child, :last-child)
* Pseudo-elements (::before, ::after, ::first-letter, ::first-line)

## **✅ 3. CSS Box Model**

* Understand **Margin, Border, Padding, and Content**
* Use box-sizing: border-box; for consistent sizing
* Apply width, height, max-width, min-width

## **✅ 4. CSS Positioning & Layout**

* **Display Property** (block, inline, inline-block, none, flex, grid)
* **Position Property** (static, relative, absolute, fixed, sticky)
* **Float & Clear** (float: left/right;, clear: both;)
* **Z-index** for element stacking
* **Overflow Handling** (visible, hidden, scroll, auto)
* **Flexbox Layout** (display: flex;, justify-content, align-items, flex-wrap)
* **CSS Grid Layout** (display: grid;, grid-template-columns/rows, gap)

## **✅ 5. CSS Colors & Backgrounds**

* Use different color formats (HEX, RGB, HSL, named colors)
* Apply **gradients** (linear-gradient, radial-gradient)
* Set background properties (background-color, background-image, background-size, background-position, background-repeat)

## **✅ 6. CSS Typography**

* Use font families (font-family: Arial, sans-serif;)
* Font sizes (px, em, rem, %, vw/vh)
* Font weights (normal, bold, lighter, bolder)
* Apply **text properties** (text-align, text-transform, letter-spacing, word-spacing, line-height)
* Use **Google Fonts** and **@font-face**

## **✅ 7. CSS Borders & Shadows**

* Apply border properties (border, border-radius)
* Use box shadows (box-shadow: 5px 5px 10px rgba(0,0,0,0.5);)
* Text shadows (text-shadow: 2px 2px 5px gray;)

## **✅ 8. CSS Lists & Tables**

* Style **lists** (list-style-type, list-style-position, list-style-image)
* Style **tables** (border-collapse, border-spacing, table-layout, nth-child selectors)

## **✅ 9. CSS Animations & Transitions**

* Use transition (transition: all 0.5s ease-in-out;)
* Apply @keyframes for animations (animation-name, animation-duration, animation-iteration-count)
* Use transform (rotate(), scale(), translate(), skew())

## **✅ 10. CSS Media Queries & Responsive Design**

* Use @media for responsiveness (@media (max-width: 768px) {})
* Apply vw, vh, %, rem, em for flexible units
* Learn **Mobile-first vs Desktop-first Design**

## **✅ 11. CSS Variables & Custom Properties**

* Declare and use CSS variables (--primary-color: blue; color: var(--primary-color);)
* Use :root for global variables

## **✅ 12. CSS Best Practices**

* Use **minified CSS** for performance
* Organize CSS with comments (/\* Section: Typography \*/)
* Avoid !important unless necessary
* Use shorthand properties (margin: 10px 5px 15px;)