



HTML Interview Questions and Answers

Introduction

Due to globalization, almost every country in the world is adapting to digital technologies. With this adaptation, most companies are starting to have a digital presence through web pages and websites. So, the demand for HTML is now more than ever. On the face of it, HTML might seem like a basic language to learn, but learning it and making a career out of it will surely be a fulfilling and long-lasting profession for you. We hope that these **HTML interview questions and answers** will help you in your journey to land a career in HTML. Begin your web development career with our [HTML training in Chennai](#).

List of HTML Interview Questions for Freshers

1. What is HTML?
2. Define Void elements in HTML.
3. Explain tags and attributes in HTML.
4. Explain the distinction between GET and POST methods in HTML forms.

5. What is semantic HTML?

Find what we cover in our web development course with our [HTML course syllabus](#).

HTML Interview Questions and Answers for Freshers

1. What is HTML?

In 1991, Tim Berners-Lee, a British computer scientist, along with his team at CERN (European Organization for Nuclear Research), introduced HTML. HTML, an abbreviation for HyperText Markup Language, serves as the primary language for crafting and formatting documents across the internet. Through the use of markup tags, it organizes various elements like text, images, links, and multimedia within web pages. By defining structures such as headings, paragraphs, and lists, HTML lays the foundation for creating cohesive web content.

2. Define Void elements in HTML.

Void elements in HTML are those that lack any content or text and thus don't necessitate a closing tag. Instead, they're represented by self-closing tags. These elements serve to insert items such as images, line breaks, or horizontal rules onto a webpage without additional content. Common void elements encompass `<img>`, `<br>`, `<hr>`, `<input>`, `<meta>`, `<link>`, and `<area>`.

**Take your Knowledge
Test Report**

Check your Score



3. Explain tags and attributes in HTML.

- In HTML, tags are elements enclosed in angle brackets that define parts of a webpage's structure or content. They usually come in pairs: an opening tag indicates the beginning of an element, while a closing tag marks its end.
- Attributes, on the other hand, provide additional information about an element within its opening tag. They consist of a name-value pair and modify the

element's behavior or appearance. For example, the src attribute in the tag specifies the image's source URL. In HTML, tags and attributes work together to create structured and interactive web content.

4. Explain the distinction between GET and POST methods in HTML forms.

The GET method transmits form data as part of the URL query string, making it visible in the browser's address bar. It's typically used for transferring small amounts of data. Conversely, the POST method sends form data within the request body, keeping it hidden from the URL. This method is preferred for transferring large amounts of data securely.

5. What is semantic HTML?

Semantic HTML entails employing HTML elements that convey meaning beyond just presentation. This approach aids search engines and assistive technologies in comprehending the structure and intent of the content. Examples of semantic elements include <header>, <footer>, <nav>, <article>, <section>, among others.

Learn the basics of web development with our [HTML tutorial for beginners](#).

List of HTML Interview Questions for Experienced

1. Differentiate HTML and XHTML.
2. Discuss the advantages of collapsing white space in HTML.
3. Differentiate HTML and HTML5.
4. What are the ways to redirect to a particular section of a page using HTML?
5. Describe the layout structure of HTML.

Read here to know the best [HTML salary in India for freshers and experienced](#), and boost your earning potential.

HTML Technical Interview Questions and Answers for Experienced

1. Differentiate HTML and XHTML.

Aspect	HTML	XHTML
Syntax	Lenient, allows for some errors	Stricter, follows rules inherited from XML
Parsing	More forgiving, tolerates minor errors	Stringent, flags errors for XML deviations
Document Structure	Optional tags can be omitted	Mandates well-defined document structure
Quoting Attributes	Quotes optional unless necessary	Attribute values must be enclosed in quotes
Case Sensitivity	Case-insensitive for tags and attributes	Case-sensitive for both tags and attributes
MIME Type	Uses text/html	Requires application/xhtml+xml or application/xml
Purpose	Widely used, standard for web development	Enhanced interoperability and compatibility with XML-based technologies

2. Discuss the advantages of collapsing white space in HTML.

Collapsing white space in HTML is when consecutive white space characters like spaces, tabs, and newlines are displayed as a single space character. This feature offers several benefits:

- **Enhanced Readability:** It maintains a neater and more legible HTML code, allowing developers to structure their code clearly without impacting the page's appearance.

- **Decreased File Size:** By collapsing white space, redundant characters in the HTML document are minimized, resulting in smaller file sizes. This can enhance page loading times, particularly for users with slower internet connections.
- **Uniform Layout:** Collapsing white space ensures that the webpage's layout remains consistent across various browsers and devices, regardless of the white space quantity in the HTML code.
- **Prevention of Unintended Layout Alterations:** Without collapsing white space, surplus white space characters may inadvertently influence element spacing on the webpage, leading to unintended layout discrepancies.

3. Differentiate HTML and HTML5.

Aspect	HTML	HTML5
Semantic Elements	Limited semantic elements	Introduces new semantic elements (<header>, <footer>, <nav>, <article>, <section>, <aside>) for clearer structure and clarity of web content
Multimedia Integration	Requires third-party plugins like Flash	Built-in support for audio and video playback through <audio> and <video> elements, eliminating the need for third-party plugins
Graphics Rendering	Limited capabilities	Introduces <canvas> element for dynamic rendering of graphics, charts, animations, and games directly within the browser without plugins
Form Advancements	Limited input types and attributes	Introduces new input types (e.g., email, url, date) and attributes (e.g., required, placeholder) for form elements, simplifying form creation

		and validation
Offline Capabilities	Lacks built-in offline support	Incorporates features like Application Cache and Web Storage (localStorage and sessionStorage) for offline operation and local data storage
Enhanced Accessibility	Basic accessibility features	Introduces various features and attributes (<figure>, <figcaption>, alt, aria-*) to enhance accessibility for users with disabilities
Mobile Adaptability	Limited support for mobile devices	Tailored for mobile-friendliness with responsive design, touch events, and device orientation detection, optimizing web applications for smartphones and tablets
Backward Compatibility	Compatibility issues with older browsers	Ensures backward compatibility with prior HTML versions, allowing gradual updating of existing web pages without disrupting compatibility

4. What are the ways to redirect to a particular section of a page using HTML?

To direct users to a specific section of a webpage using HTML, utilize the anchor (<a>) element with the href attribute. Here's a concise guide:

First, designate the section you wish to direct to by assigning an id attribute to its HTML element.

```
<div id="section1">
```

```
<!-- Section 1 Content -->
```

```
</div>
```

```
<div id="section2">
```

```
<!-- Section 2 Content -->
```

```
</div>
```

Then, create a link to the desired section by specifying the URL of the page followed by # and the id of the section.

```
<a href="#section1">Go to Section 1</a>
```

```
<a href="#section2">Go to Section 2</a>
```

Clicking these links redirects users to the corresponding sections within the same page.

To direct to a section on a different page, include the page's filename followed by # and the section's id.

```
<a href="otherpage.html#section1">Go to Section 1 on Other Page</a>
```

This redirects users to the specified section of the other page upon clicking the link.

**Take your Knowledge
Test Report**

Check your Score



5. Describe the layout structure of HTML.

The layout structure of an HTML document typically adheres to a hierarchical arrangement, encompassing diverse elements that delineate the content, organization, and styling of a webpage. Here's a breakdown of its core constituents:

- **Document Type Declaration (<!DOCTYPE>):** Positioned at the outset of the document, this declaration denotes the HTML version in use, ensuring accurate rendering by web browsers.
- **HTML Element (<html>):** Serving as the root element, <html> encapsulates all other elements within the HTML document.
- **Head Section (<head>):** Within the <head> section reside meta-information like the document title, character encoding, external stylesheets, scripts, and meta tags.
- **Document Metadata (Meta Tags):** These tags, housed within the <head> section, furnish supplementary information such as keywords, descriptions, and viewport settings.
- **Document Title (<title>):** The <title> element specifies the document's title, visible in the browser's title bar or tab.
- **Document Stylesheets (<link> or <style>):** Stylesheets, linked externally via <link> or embedded within <style>, dictate the presentation and layout of the document's content.
- **Document Scripts (<script>):** Scripts, including JavaScript, are incorporated in the <head> section or towards the end of the <body> section to enrich interactivity and functionality.
- **Body Section (<body>):** Central to the layout, the <body> section encompasses the primary content of the webpage, comprising text, images, links, forms, and other elements.
- **Structural Elements:** Elements such as <header>, <footer>, <nav>, <main>, <article>, <section>, and <aside> lend semantic meaning and structure to the content within the <body> section.
- **Text Elements:** Various elements like headings (<h1>, <h2>, <h3>, etc.), paragraphs (<p>), lists (, ,), and inline elements (, , , etc.) organize and format textual content.
- **Media Elements:** Elements such as , <audio>, <video>, and <iframe> enable the embedding of images, audio, video, and external content within the document.
- **Hyperlinks (<a>):** Employing the <a> element, hyperlinks facilitate navigation within the document or to external resources.

- **Forms and Input Elements (<form>, <input>, <textarea>, etc.):** Forms, along with various input elements, gather user input, supporting diverse interactions like text input, checkboxes, radio buttons, and dropdown menus.
- **Comments (<!-- -->):** Enclosed within <!-- and --> tags, comments furnish explanatory notes within the HTML code, unrendered by web browsers.

Conclusion

Success in an HTML interview hinges on demonstrating mastery of Semantic tags, Accessibility (ARIA), and modern HTML5 APIs. Showing you can build structured, SEO-friendly layouts is the first step toward becoming a professional web developer. These HTML Interview Questions and Answers are curated specifically to help you in grasping HTML just enough to ace your interview easily.

Ready to build the web? Master modern frontend development with our [software training institute in Chennai](#).