

Explore More

Subscription : Premium CDAC NOTES & MATERIAL @99



Contact to Join
Premium Group



Click to Join
Telegram Group

<CODEWITHARRAY'S/>

For More E-Notes

Join Our Community to stay Updated

TAP ON THE ICONS TO JOIN!

	codewitharrays.in freelance project available to buy contact on 8007592194	
SR.NO	Project NAME	Technology
1	Online E-Learning Platform Hub	React+Springboot+MySql
2	PG Mates / RoomSharing / Flat Mates	React+Springboot+MySql
3	Tour and Travel management System	React+Springboot+MySql
4	Election commition of India (online Voting System)	React+Springboot+MySql
5	HomeRental Booking System	React+Springboot+MySql
6	Event Management System	React+Springboot+MySql
7	Hotel Management System	React+Springboot+MySql
8	Agriculture web Project	React+Springboot+MySql
9	AirLine Reservation System / Flight booking System	React+Springboot+MySql
10	E-commerce web Project	React+Springboot+MySql
11	Hospital Management System	React+Springboot+MySql
12	E-RTO Driving licence portal	React+Springboot+MySql
13	Transpotation Services portal	React+Springboot+MySql
14	Courier Services Portal / Courier Management System	React+Springboot+MySql
15	Online Food Delivery Portal	React+Springboot+MySql
16	Muncipal Corporation Management	React+Springboot+MySql
17	Gym Management System	React+Springboot+MySql
18	Bike/Car ental System Portal	React+Springboot+MySql
19	CharityDonation web project	React+Springboot+MySql
20	Movie Booking System	React+Springboot+MySql

freelance_Project available to buy contact on 8007592194		
21	Job Portal web project	React+Springboot+MySql
22	LIC Insurance Portal	React+Springboot+MySql
23	Employee Management System	React+Springboot+MySql
24	Payroll Management System	React+Springboot+MySql
25	RealEstate Property Project	React+Springboot+MySql
26	Marriage Hall Booking Project	React+Springboot+MySql
27	Online Student Management portal	React+Springboot+MySql
28	Resturant management System	React+Springboot+MySql
29	Solar Management Project	React+Springboot+MySql
30	OneStepService LinkLabourContractor	React+Springboot+MySql
31	Vehical Service Center Portal	React+Springboot+MySql
32	E-wallet Banking Project	React+Springboot+MySql
33	Blogg Application Project	React+Springboot+MySql
34	Car Parking booking Project	React+Springboot+MySql
35	OLA Cab Booking Portal	React+NextJs+Springboot+MySql
36	Society management Portal	React+Springboot+MySql
37	E-College Portal	React+Springboot+MySql
38	FoodWaste Management Donate System	React+Springboot+MySql
39	Sports Ground Booking	React+Springboot+MySql
40	BloodBank mangement System	React+Springboot+MySql

41	Bus Tickit Booking Project	React+Springboot+MySql
42	Fruite Delivery Project	React+Springboot+MySql
43	Woodworks Bed Shop	React+Springboot+MySql
44	Online Dairy Product sell Project	React+Springboot+MySql
45	Online E-Pharma medicine sell Project	React+Springboot+MySql
46	FarmerMarketplace Web Project	React+Springboot+MySql
47	Online Cloth Store Project	React+Springboot+MySql
48	Train Ticket Booking Project	React+Springboot+MySql
49	Quizz Application Project	JSP+Springboot+MySql
50	Hotel Room Booking Project	React+Springboot+MySql
51	Online Crime Reporting Portal Project	React+Springboot+MySql
52	Online Child Adoption Portal Project	React+Springboot+MySql
53	online Pizza Delivery System Project	React+Springboot+MySql
54	Online Social Complaint Portal Project	React+Springboot+MySql
55	Electric Vehical management system Project	React+Springboot+MySql
56	Online mess / Tiffin management System Project	React+Springboot+MySql
57		React+Springboot+MySql
58		React+Springboot+MySql
59		React+Springboot+MySql
60		React+Springboot+MySql

Spring Boot + React JS + MySQL Project List

Sr.No	Project Name	YouTube Link
1	Online E-Learning Hub Platform Project	https://youtu.be/KMjyBaWmgzg?si=YckHuNzs7eC84-IW
2	PG Mate / Room sharing/Flat sharing	https://youtu.be/4P9clHg3wvk?si=4uEsi0962CG6Xodp
3	Tour and Travel System Project Version 1.0	https://youtu.be/-UHOBywHaP8?si=KHHfE_A0uv725f12
4	Marriage Hall Booking	https://youtu.be/VXz0kZQi5to?si=ILOS-QG3TpAFP5k7
5	Ecommerce Shopping project	https://youtu.be/vJ_C6LkhrZ0?si=YhcBylSErvdn7paq
6	Bike Rental System Project	https://youtu.be/FlzsAmIBCbk?si=7ujQTJqEgkQ8ju2H
7	Multi-Restaurant management system	https://youtu.be/pvV-pM2Jf3s?si=PgvnT-yFc8ktrDxB
8	Hospital management system Project	https://youtu.be/lynlouBZvY4?si=CXzQs3BsRkjKhZCw
9	Municipal Corporation system Project	https://youtu.be/cVMx9NVyl4I?si=qX0oQt-GT-LR_5jF
10	Tour and Travel System Project version 2.0	https://youtu.be/_4u0mB9mHXE?si=gDiAhKBowi2gNUKZ

Sr.No	Project Name	YouTube Link
11	Tour and Travel System Project version 3.0	https://youtu.be/Dm7nOdpasWg?si=P_Lh2gcOFhlyudug
12	Gym Management system Project	https://youtu.be/J8_7Zrkg7ag?si=LcxV51ynfUB7OptX
13	Online Driving License system Project	https://youtu.be/3yRzsMs8TLE?si=JRI_z4FDx4Gmt7fn
14	Online Flight Booking system Project	https://youtu.be/m755rOwdk8U?si=HURvAY2VnizlyJlh
15	Employee management system project	https://youtu.be/ID1iE3W_GRw?si=Y_jv1xV_BljhrD0H
16	Online student school or college portal	https://youtu.be/4A25aEKfei0?si=RoVgZtxMk9TPdQvD
17	Online movie booking system project	https://youtu.be/Lfjv_U74SC4?si=fiDvrhhrjb4KSIsm
18	Online Pizza Delivery system project	https://youtu.be/Tp3izreZ458?si=8eWAOzA8SVdNwlyM
19	Online Crime Reporting system Project	https://youtu.be/0UlzReSk9tQ?si=6vN0e70TVY1GOwPO
20	Online Children Adoption Project	https://youtu.be/3T5HC2HKyT4?si=bntP78niYH802I7N

Q - 1) What is HTML?

HTML stands for HyperText Markup Language. It's the standard language used to create and design web pages and web applications. HTML structures the content on a web page by using elements, which are enclosed in tags. These elements can define headings, paragraphs, links, images, lists, and other types of content.

Example:

```
<!DOCTYPE html>
```

My First Web Page

Hello, World!

This is a paragraph.

This is a link

In this snippet:

`<!DOCTYPE html>` declares the document type.

`<html>` is the root element.

`<head>` contains metadata about the document.

`<title>` sets the title of the web page.

`<body>` contains the content of the web page.

`<h1>` defines a heading.

`<p>` defines a paragraph. `<a>` creates a hyperlink. HTML works alongside CSS (Cascading Style Sheets) for styling and JavaScript for functionality to create fully interactive web experiences.

Q - 2) What are the different versions of HTML?

HTML (Hypertext Markup Language) has undergone several versions since its inception. Here's a brief overview of the major versions:

HTML 1.0 (1991) - The original version, introduced by Tim Berners-Lee. It was very basic, supporting simple text formatting and linking. HTML 2.0 (1995) - Standardized by the Internet Engineering Task Force (IETF), it added features like tables, forms, and text alignment. HTML 3.2 (1997) - Introduced by the World Wide Web Consortium (W3C), it included support for style sheets, scripting, and more advanced layout options. HTML 4.0 (1997) - Added support for scripting, style sheets, and improved support for multimedia. It had three versions: HTML 4.0 Strict, HTML 4.0 Transitional, and HTML 4.0 Frameset, each with different rules for backward compatibility. HTML 4.01 (1999) - A minor revision of HTML 4.0, which included bug fixes and minor improvements. XHTML 1.0 (2000) - A

reformulation of HTML 4.01 in XML, making it more rigorous and consistent. It aimed for greater compatibility with XML tools. HTML5 (2014) - A major update to HTML, it introduced new elements and APIs for modern web applications, improved support for multimedia, and aimed to be backward compatible with older versions. HTML5 is now the standard for modern web development. HTML5.1 (2016) - An update to HTML5 that included improvements and clarifications based on user feedback and evolving web practices. HTML5.2 (2017) - Further refined HTML5 with new features and enhancements, including updates to the specifications for web applications. HTML Living Standard - The latest version of HTML, maintained by the Web Hypertext Application Technology Working Group (WHATWG). It's a continuously evolving specification that builds on HTML5 and incorporates ongoing updates and improvements. Each version has brought new features and improvements to better support the evolving needs of web development.

Q - 3) What is the purpose of the `<!DOCTYPE html>` declaration?

The `<!DOCTYPE html>` declaration is used to specify the HTML version and document type to the web browser. It ensures that the browser renders the page in standards mode rather than quirks mode, which helps in maintaining consistent and predictable rendering across different browsers.

For HTML5, which is the current standard, `<!DOCTYPE html>` is a simplified and universal way to ensure that the document adheres to HTML5 specifications. It's essential for modern web development to ensure proper behavior and compatibility of web pages.

Q - 4) What is the difference between HTML and XHTML?

HTML (HyperText Markup Language) and XHTML (Extensible Hyper Text Markup Language) are both markup languages used for creating web pages, but they have some key differences:

Syntax Rules: HTML: More lenient with syntax. For instance, you can omit end tags for certain elements, and it's forgiving of unclosed tags or attributes without quotes. XHTML: Stricter and follows XML syntax rules. All tags must be properly closed, attributes must be quoted, and the document must be well-formed XML. **Document Structure:** HTML: Can be somewhat forgiving with errors and still render content in a browser. XHTML: Requires a well-formed document; any syntax errors may prevent the document from being displayed. **Compatibility:** HTML: Browsers are very accommodating with HTML and will often render it even if there are errors. XHTML: Since it's based on XML, it requires more strict adherence to syntax rules. Browsers need to handle XHTML as XML documents, and they may not always handle errors gracefully. **Standards:** HTML: Older versions, like HTML 4.01, are less strict compared to XHTML. XHTML: Aimed to combine the flexibility of HTML with the strictness of XML. XHTML 1.0 is similar to HTML 4.01, but XHTML 1.1 and XHTML 2.0 introduce more stringent rules and newer features. **Transition:** XHTML was developed to be a bridge between HTML and XML, but HTML5 has since become the dominant standard, incorporating many features and improvements that were originally part of XHTML.

Q - 5) What are semantic elements in HTML? Give examples.

Semantic elements in HTML are elements that clearly describe their meaning in a human- and machine-readable way. They provide context to the content within them, making it easier for search engines and other user agents to understand the structure and importance of the content. Here are some examples:

: Represents the introductory content or a set of navigational links. Typically used at the top of a page or section.

Example:

Website Title

Home

About

: Defines a section of navigation links.

Example:

Home

Services

: Specifies the main content of a document. It should be unique to the document and exclude headers, footers, and navigation links.

Example:

Article Title

This is the main content of the article.

: Represents a self-contained piece of content that could be distributed independently, such as a blog post or news article

Example:

Blog Post Title

This is the content of the blog post.

: Defines a section of content, usually with its own heading. It is used to group related content.

Example:

Section Title

Content related to this section.

: Represents the footer for a document or section, typically containing information like the author, copyright, or related links.

Example:

© 2024 Your Website

: Represents content that is tangentially related to the content around it, such as sidebars or pull quotes.

Example:

Related Information

Some additional information related to the main content.

Using these semantic elements helps improve the accessibility and SEO of your web pages by clearly defining the roles of different parts of your content.

Q - 6) What is the purpose of the

tag? The

tag in HTML is used to contain metadata and other information about the document that isn't directly visible on the page itself. It includes elements such as:

: Specifies the title of the document, which appears in the browser's title bar or tab.

: Provides metadata like the character set, viewport settings, and description. : Links to external resources like stylesheets.

: Contains internal CSS styles.

tags.

Example:

<!DOCTYPE html>

Inline JavaScript Example

Hello, World!

External JavaScript File:

You can also keep your JavaScript code in a separate file and link it to your HTML document using the

Placement of

At the End of

: Placing the

Q - 21) What is the tag used for?

The `<iframe>` tag in HTML is used to embed another HTML document within the current document. It creates an inline frame, which allows you to display content from another source or webpage inside your own page. Here's a basic example:

In this example, the `<iframe>` embeds the content from "https://www.example.com" and sets its width and height to 600px and 400px, respectively.

The `src` attribute specifies the URL of the page to be displayed, while `width` and `height` define the size of the `iframe`. Other attributes you can use include `title` (for accessibility), `frameborder` (to specify whether to display a border), and `sandbox` (to restrict the `iframe`'s abilities).

Q - 22) How do you define a section of a web page in HTML5?

In HTML5, you can define a section of a web page using the `<section>` element. The

`<section>` tag represents a standalone section of content that is thematically related and could be independently distributable. It's useful for grouping content into logical sections and can be accompanied by a heading to describe the content of that section.

```
<!DOCTYPE html>
```

Example Page

My Web Page

Introduction

This is the introduction section of the page.

Main Content

This section contains the main content of the page.

Footer information

In this example, the page is divided into sections for the introduction and main content. Each

`<section>` element can be styled individually with CSS or manipulated with JavaScript as needed.

Q - 23) What are HTML entities? Give an example.

HTML entities are special codes used in HTML to represent characters that have a special meaning in HTML or are not easily typable on a keyboard. They start with an ampersand (&), followed by the entity name or code, and end with a semicolon (;). They ensure that these characters are displayed correctly in web browsers.

For example:

& represents the ampersand (&) character. < represents the less-than sign (<). > represents the greater-than sign (>). So if you want to display the text 5 < 10 in HTML, you would write it as 5 < 10.

Q - 24) How do you make a webpage responsive using HTML?

To make a webpage responsive using HTML, you need to use a combination of HTML and CSS. Here's a basic overview:

Viewport Meta Tag: Add the viewport meta tag in the

section of your HTML. This helps control the layout on mobile browsers.

Flexible Grid Layouts: Use CSS to create flexible grid layouts. For example, use percentage-based widths or CSS Grid/Flexbox to make elements adapt to different screen sizes.

```
.container { display: flex; flex-wrap: wrap; }
```

```
.item { flex: 1 1 100%; /* Adjusts the item size based on available space */ box-sizing: border-box; }
```

Media Queries: Use CSS media queries to apply different styles based on the viewport size.

```
/* Default styles for larger screens */ .box { width: 100%; }
```

```
/* Styles for devices with a max width of 600px */ @media (max-width: 600px) { .box { width: 100%; } }
```

```
/* Styles for devices with a min width of 601px */ @media (min-width: 601px) { .box { width: 50%; } }
```

Fluid Images: Make images responsive by setting their max-width to 100%.

```
img { max-width: 100%; height: auto; }
```

Responsive Typography: Use relative units (like em, rem, or percentages) for font sizes to ensure text scales appropriately on different devices.

```
body { font-size: 1rem; /* 16px if the root font size is 16px */ }
```

```
h1 { font-size: 2rem; /* 32px if the root font size is 16px */ }
```

By combining these techniques, you can create a responsive webpage that looks good on a variety of devices.

Q - 25) What is the element used for?

The `` element in HTML is used to define multiple versions of an image for different conditions, like varying screen sizes or device resolutions. It allows you to provide different

image sources based on specific criteria, which helps improve performance and responsiveness.

Here's a basic example of how it works:

In this example:

The elements specify different image files to use based on the viewport width. The element serves as a fallback if none of the conditions in the elements are met. The browser will choose the most appropriate image based on the conditions specified in the media attributes of the elements.

Q - 26) What are ARIA attributes in HTML?

ARIA (Accessible Rich Internet Applications) attributes are used in HTML to improve accessibility for users with disabilities. They help to make web content and applications more understandable and navigable by screen readers and other assistive technologies.

Here are some key ARIA attributes:

aria-label: Provides an accessible name for an element. For example, X. aria-labelledby: Refers to the ID of another element that provides a label for the current element. This is useful for elements that need to be described by a separate piece of text. For example, . aria-describedby: Points to an element that provides a description of the current element. For instance, . aria-live: Indicates that an element will be updated dynamically and how those updates should be announced by assistive technologies. For example, aria-live="polite". aria-checked: Specifies the state of a checkbox or radio button (e.g., true, false, or mixed). aria-expanded: Indicates whether a collapsible element is expanded or collapsed. For example, aria-expanded="true". aria-hidden: Hides an element from screen readers. This can be useful for hiding decorative elements that are not useful to assistive technologies. For example, aria-hidden="true". role: Defines the role of an element, such as a button, navigation landmark, or dialog. For example,

. ARIA attributes enhance the semantic meaning of HTML elements and ensure that web applications are more accessible to everyone.

Q - 27) How can you include an external JavaScript file in HTML?

To include an external JavaScript file in your HTML, you use the

Hello, World!

src Attribute: Specifies the path to the external JavaScript file. defer Attribute: Ensures that the script is executed after the HTML is fully parsed. This is optional but recommended for scripts that don't need to run until the DOM is fully loaded. Placement: While the script tag is often placed in the

, it can also be placed just before the closing

tag to ensure that it executes after the HTML content has loaded. If the file is in the same directory as your HTML file, you can just use the file name:

Q - 28) What are data attributes in HTML?

Data attributes in HTML allow you to store extra information on HTML elements without affecting their presentation or functionality. They are particularly useful for adding custom data to elements that can be accessed later through JavaScript.

A data attribute is defined by adding an attribute to an HTML element that starts with data- followed by a name of your choosing. For example:

User Information

In this example, data-user-id and data-role are custom data attributes. You can then access these attributes in JavaScript using the dataset property:

```
const div = document.querySelector('div'); console.log(div.dataset.userId); // Outputs: 12345 console.log(div.dataset.role); // Outputs: admin
```

Data attributes are a versatile way to embed additional information in HTML elements, and they can be particularly handy for various tasks like configuration, state management, or passing metadata.

Q - 29) How do you use the data-* attribute in HTML?

The data-* attribute in HTML is a way to store custom data on HTML elements. This attribute allows you to embed custom information directly in your HTML elements that can be accessed and manipulated using JavaScript. Here's a quick guide on how to use it:

Basic Usage Add a data-* Attribute to an Element:

User Info

Access data-* Attributes Using JavaScript:

```
// Get the element const div = document.querySelector('div');  
  
// Access data attributes const userId = div.getAttribute('data-user-id'); const role = div.getAttribute('data-role');  
  
console.log(userId); // Output: 12345 console.log(role); // Output: admin
```

Modify data-* Attributes Using JavaScript:

```
// Set new data attribute div.setAttribute('data-user-id', '67890');  
  
// Get updated data attribute console.log(div.getAttribute('data-user-id')); // Output: 67890
```

Access data-* Attributes Using Dataset Property:

```
// Access using dataset property const userId = div.dataset.userId; const role = div.dataset.role;
```

```
console.log(userId); // Output: 67890 console.log(role); // Output: admin
```

Attribute Naming: The data-* attribute name must start with data-, followed by a name of your choice. The name must only contain lowercase letters, digits, hyphens, and underscores. **JavaScript Access:** JavaScript automatically converts hyphens to camelCase. For instance, data-user-id will be accessed as dataset.userId. This method is useful for storing extra information that doesn't belong in the HTML semantics but needs to be accessed by JavaScript.

Q - 30) What is the purpose of the viewport meta tag?

The viewport meta tag is used in HTML to control the layout and scaling of a webpage on mobile devices. It helps ensure that web pages are displayed correctly on different screen sizes and resolutions by setting parameters for the viewport's width, initial scale, and more.

Here's a basic example of how it's used:

width=device-width sets the viewport width to match the device's screen width. initial-scale=1.0 sets the initial zoom level to 100%, meaning no zoom is applied when the page is first loaded. Using this tag helps create responsive web designs that work well on both desktops and mobile devices.

Q - 31) How do you define an HTML document's character encoding?

To define a character encoding in an HTML document, you use the

tag within the

section of your HTML. The charset attribute specifies the character encoding for the document. Here's how you typically do it:

```
<!DOCTYPE html>
```

Document Title

In this example, UTF-8 is specified as the character encoding. UTF-8 is a popular encoding that supports a wide range of characters from different languages and is generally recommended for most web content.

Q - 32) What is the difference between

and

tags? The

and

tags in HTML are both used for structuring content, but they have different purposes:

: This tag is used to define a section of a document. It represents a thematic grouping of content, typically with a heading. Sections can be used to group related content together, and they help in organizing the structure of the page. For example, you might use a for different parts of a webpage like “Introduction,” “Main Content,” and “Conclusion.”

Introduction

This is the introduction section.

: This tag is used for self-contained content that could be distributed or reused independently, such as a blog post, news article, or user comment. An article should make sense on its own and be independently distributable. The

tag is often used for content that could be a stand-alone piece.

Blog Post Title

This is the content of the blog post.

Q - 33) How do you include a font in an HTML document?

To include a font in an HTML document, you can use the @font-face rule in your CSS or link to a web font service like Google Fonts. Here's how you can do both:

Using @font-face Download the Font Files: Make sure you have the font files (e.g., .ttf, .woff, .woff2). Add the Font to Your CSS: @font-face { font-family: 'MyCustomFont'; src: url('path/to/font.woff2') format('woff2'), url('path/to/font.woff') format('woff'); font-weight: normal; font-style: normal; }

body { font-family: 'MyCustomFont', sans-serif; }

font-family: The name you want to use for the font. src: The path to the font files and their format. font-weight and font-style: Optional properties to define the font's weight and style. Using Google Fonts Go to Google Fonts: Visit Google Fonts. Select a Font: Choose a font and customize it if needed. Embed the Font: Google Fonts provides a link tag to include in the of your HTML document.

Use the Font in Your CSS:

body { font-family: 'Roboto', sans-serif; }

Both methods allow you to customize your website's typography with different fonts!

Q - 34) What is the use of the tag in forms?

The tag in HTML is used to define labels for form elements. It helps improve the accessibility and usability of forms. Here's how it works:

Associating Labels with Form Elements: The tag can be associated with a form control (like

,

, or) either by using the for attribute, which matches the id of the form control, or by placing the form control inside the tag. This association allows users to click on the label to focus or activate the corresponding form element, making the form easier to use.

Improving Accessibility: Screen readers and other assistive technologies use the label to describe the purpose of the form control to users with visual impairments. This makes forms more accessible to everyone. Providing Descriptive Text: Labels provide a clear and descriptive text for each form control, helping users understand what information they need to enter. Here's an example of using the tag with the for attribute:

Name:

Email:

In this example, clicking on the "Name" or "Email" label will focus on the corresponding input field.

Q - 35) How do you handle form validation in HTML?

Form validation in HTML can be handled in several ways, from basic HTML attributes to more advanced JavaScript techniques. Here's a breakdown:

1. HTML5 Built-in Validation: HTML5 provides several attributes that can be used directly in your form elements for validation:

required: Makes the field mandatory. type: Defines the type of data expected (e.g., email, number, url), and browsers will automatically validate the input based on the type. min and max: Specify the range of acceptable values for fields like number or date. pattern: Allows you to specify a regular expression that the input must match. maxlength and minlength: Define the length of the input.

Submit

JavaScript Validation: For more complex validation logic or custom messages, you can use JavaScript. You can validate form data when the form is submitted or when the user inputs data.

Submit

CSS for Validation Styles: CSS can be used to visually indicate valid or invalid input.

Custom Validation Messages: You can customize the validation messages using JavaScript or HTML attributes like title.

When the input doesn't match the pattern, the browser will display the message defined in the title attribute.

Using a combination of these methods can help you create a robust form validation system that improves user experience and data integrity.

Q - 36) What is the `<button>` tag used for?

The `<button>` tag in HTML is used to create clickable buttons. These buttons can be used to submit forms, trigger JavaScript functions, or perform other actions. Here's a basic example:

Click Me

In this example:

`type="button"` specifies that the button is a generic button that doesn't submit a form. `onclick="alert('Button clicked!')"` is an event handler that executes JavaScript when the button is clicked. The text between the tags ("`Click Me`") is the label displayed on the button. You can also use the `type` attribute to define different button behaviors:

`type="submit"`: Submits the form it is associated with. `type="reset"`: Resets the form fields to their default values.

Q - 37) How do you embed a video in HTML?

To embed a video in HTML, you can use the `<video>` tag. Here's a basic example:

Your browser does not support the video tag.

In this example:

`width` and `height` set the dimensions of the video player. `controls` adds playback controls (play, pause, volume) to the video. `src` specifies the video file and its format. You can include multiple `src` elements with different formats for compatibility with various browsers. The text "Your browser does not support the video tag." provides a fallback message for browsers that do not support the `<video>` element.

Q - 38) What is the difference between `` and `` tags?

The `` and `` tags both make text bold, but they serve different purposes:

(**Bold**): The `` tag is used to apply a bold style to text. It's a presentational element, meaning it's primarily about styling without conveying any special importance or meaning. For example: This text is bold.

(**Strong Importance**): The `` tag is used to indicate that the enclosed text is of strong importance or emphasis. It also renders text in bold, but it carries semantic meaning, suggesting that the text is significant in context. This can be useful for accessibility tools, like screen readers, to understand the emphasis of the text. For example:

This text is important.

Q - 39) What is the difference between and tags?

Both and tags are used in HTML to apply emphasis to text, but they serve slightly different purposes:

(Italic): This tag is used for text that should be displayed in italics. It typically denotes text that is set off from the normal text for stylistic reasons, such as technical terms, foreign words, or phrases with a different tone. It's more about presentation rather than meaning.

This text is in italics.

(Emphasis): This tag is used to emphasize text, indicating that it should be stressed in the spoken form or interpreted with more importance. It usually renders text in italics by default, but its primary function is semantic, meaning it conveys a meaning or emphasis that can be important for screen readers and search engines.

This text is emphasized.

Q - 40) How do you create a responsive image in HTML?

To create a responsive image in HTML, you can use the CSS max-width property to ensure the image scales proportionally with its container. Here's a simple way to do it:

HTML Code:

CSS Code:

```
.responsive-img { max-width: 100%; height: auto; }
```

In this example:

max-width: 100%; makes sure the image does not exceed the width of its container. height: auto; maintains the image's aspect ratio, so it scales proportionally. You can also use the srcset attribute in the tag to serve different image sizes based on the viewport size or resolution:

This approach helps optimize image loading for various devices and screen sizes.

Q - 41) What are HTML5 input types?

HTML5 introduced a variety of new input types that enhance the functionality and user experience of forms. Here's a rundown of the new input types you can use:

: Validates that the input is a properly formatted email address. : Validates that the input is a properly formatted URL. : Allows users to enter a telephone number; some browsers may show a number keypad. : Provides a number input field with up and down arrows to increase or decrease the value. : Creates a slider for selecting a value from a range. : Allows

users to pick a date from a calendar. : Allows users to enter a time. : Allows users to enter a date and time. : Allows users to select a month and year. : Allows users to select a week and year. : Provides a color picker for selecting a color. These input types help ensure that the data entered is in the correct format and can provide users with a more intuitive and efficient way to fill out forms.

Q - 42) How do you set a default value for a form field in HTML?

To set a default value for a form field in HTML, you can use the value attribute within your form field element. Here's how you can do it for different types of form fields:

Text Input:

Text Area:

Select Dropdown:

Option 1 Option 2 Option 3

Radio Buttons:

Male Female

Checkbox:

Subscribe to newsletter

In each case, the default value is pre-filled or pre-selected when the form is initially loaded.

Q - 43) What is the `<noscript>` tag used for?

The `<noscript>` tag is used in HTML to provide alternative content for users who have JavaScript disabled in their browsers. It's useful for ensuring that users who can't or choose not to run JavaScript still get some meaningful content or a message indicating that JavaScript is required for certain features on the webpage. For example:

JavaScript is required to view this site properly. Please enable JavaScript in your browser settings.

This tag can be placed anywhere in the HTML document, and the content inside it will only be displayed if JavaScript is not enabled.

Q - 44) How do you create a hyperlink that opens in a new tab?

To create a hyperlink that opens in a new tab, you need to use the target attribute in your HTML. Here's the basic syntax:

Click here to visit Example

In this code:

href specifies the URL of the page you want to link to. target="_blank" tells the browser to open the link in a new tab. Just replace "https://www.example.com" with your desired URL and "Click here to visit Example" with your link text.

Q - 45) What are the different ways to include CSS in an HTML document?

There are three main ways to include CSS in an HTML document:

Inline CSS: This involves adding CSS styles directly to an HTML element using the style attribute. It's useful for quick, one-off styles but can be cumbersome for larger projects.

This is a blue text.

Internal CSS: You place CSS rules within a tag inside the

section of your HTML document. This method is good for styling a single document.

This is a red text.

External CSS: You link to an external CSS file using a tag in the section. This method is preferred for large projects because it allows you to separate HTML and CSS, making maintenance easier.

This text is styled by an external CSS file.

And the styles.css file might look like this:

```
p { color: green; font-size: 20px; }
```

Each method has its use cases, and often, a combination of them is used depending on the project requirements.

Q - 46) What is the role of the

tag in an HTML document? The

tag in an HTML document is used to define the title of the web page. This title is displayed on the browser's title bar or tab and is also used by search engines to index the page and by social media platforms when the page is shared. It is an important element for both user experience and SEO.

Here's a quick example of how it's used:

```
<!DOCTYPE html>
```

My Awesome Website

Welcome to My Awesome Website!

In this example, “My Awesome Website” is what will appear on the browser tab when the page is loaded.

Q - 47) How do you create a checkbox in HTML?

To create a checkbox in HTML, you use the `input` element with the type attribute set to “checkbox”. Here’s a basic example:

Example Checkbox

In this example:

The `label` element is associated with the checkbox using the `for` attribute, which should match the id of the checkbox. This makes the label clickable, which toggles the checkbox. The `id` attribute of the `input` element uniquely identifies the checkbox. The `name` attribute is used when submitting form data to group related checkboxes. The `value` attribute specifies the value that will be sent to the server if the checkbox is checked. You can also use the `checked` attribute to make the checkbox selected by default:

Q - 48) What is the use of the rel attribute in the `link` tag?

The `rel` attribute in the `link` tag specifies the relationship between the current document and the linked resource. It’s often used in HTML to define various types of links that can provide additional information or functionality. Here are some common uses:

`stylesheet`: Links to a CSS stylesheet that applies styles to the document. `icon`: Specifies a favicon or an icon that represents the website. `canonical`: Indicates the preferred URL for the content, helping with SEO. `alternate`: Provides alternative versions of the document, such as translated versions or different formats. `preload`: Instructs the browser to preload resources to improve performance. Example:

Each value for `rel` helps the browser understand how to handle the linked resource in relation to the current page.

Q - 49) How do you add an HTML comment?

To add an HTML comment, you use the following syntax:

Comments in HTML are not displayed in the browser but can be seen in the page’s source code. They’re useful for leaving notes or explanations within the code.

Q - 50) What is the significance of the

`footer` tag in HTML5? The

`footer` tag in HTML5 is used to define a footer section for a webpage or a specific section of a webpage. This tag is significant for several reasons:

Semantic HTML: It provides a clear, semantic way to structure the content of a webpage. By using

, you indicate that the content within it is related to the footer of the document or section.

Content Placement: The

tag typically contains information such as copyright notices, contact details, or links to related documents. It often appears at the bottom of a webpage or section. Accessibility and SEO: Using semantic tags like

helps screen readers and search engines understand the structure and importance of different parts of your content. It improves accessibility and can contribute to better SEO practices. Styling and Layout: While the

tag itself doesn't provide any styling, it can be targeted with CSS for layout and design purposes, making it easier to apply consistent styling to footer sections across your site. Here's a basic example of how you might use the

tag:

© 2024 My Website. All rights reserved.

Contact us: contact@example.com

In this example, the

tag contains a copyright notice and a contact email link, clearly defining the footer section of the page.



<https://www.youtube.com/@codewitharrays>



<https://www.instagram.com/codewitharrays/>



<https://t.me/codewitharrays> Group Link: <https://t.me/ccee2025notes>



[+91 8007592194](tel:+918007592194) [+91 9284926333](tel:+919284926333)



codewitharrays@gmail.com



<https://codewitharrays.in/project>