

Hema Coding School

YouTube Link: <https://www.youtube.com/@HemaCodingSchool>

1. <iframe> :

- Represents a nested browsing context.
- Primarily used to include resources from other domains or subdomains, but can also include content from the same domain.
- Strength: The embedded code is 'live' and can communicate with the parent document.
- Commonly used for embedding external web pages or third-party content.

[<iframe></iframe>](#)

2. <embed>:

- Standardized in HTML5 (previously non-standard).
- Provides an integration point for external (typically non-HTML) applications or interactive content.
- Used for embedding content for browser plugins (except SVG and HTML, which are handled differently).
- Details of what can be done with the embedded content depend on the specific browser plugin.
- `<embed src="myvideo.mp4" type="video/mp4">1`.

3. <object>:

- Represents an external resource that can be treated as an image, a nested browsing context, or processed by a plugin.
- Versatile but less commonly used than <iframe> and <embed>.
- `<object data="myflash.swf" type="application/x-shockwave-flash"></object>`

```
<iframe src="https://www.w3schools.com/html/default.asp" width="600"
height="600" frameborder="0"></iframe>
<object data="../Day 04/Day 04 Form and Media.pdf" width="600"
height="600"></object>
<embed src="../Day 04/Bank.mp4" width="600" height="600">
```

1. Anchor Tag (<a>):

- The <a> HTML element (or anchor element) creates hyperlinks to web pages, files, email addresses, or locations within the same page.
- Key attributes:
 - href: Specifies the URL that the hyperlink points to. It can be any valid URL scheme (e.g., HTTP, mailto, tel, data).
 - Example: `Visit Example`
- Content within each <a> should indicate the link's destination.

2. Link Tag (<link>):

Hema Coding School

YouTube Link: <https://www.youtube.com/@HemaCodingSchool>

- The <link> tag defines the relationship between the current document and an external resource.
- Common uses:
 - Linking to external style sheets (CSS) for styling web pages.
 - Adding a favicon (small icon) to your website.
- The <link> element is an empty tag, containing only attributes.
- `<link rel="stylesheet" href="styles.css">`

```
<a href="https://www.w3schools.com/html/default.asp" target="_blank"
title="this is w3">w3schools</a>
  <a href="../Day 03/text-format-and-structure.html">TextFormat</a>
  <ul>
<li><a href="../Day 04/form-image-audio-video.html">Form and
Image</a></li>
<li><a href="../Day 03/Day 03 Text Formate Structure.pdf">Day 03
Notes</a></li>
<li><a href="../Day 04/Nature 2.jpg" title="Nature
image">Image</a></li>
  </ul>
```

Selectors:

CSS selectors are patterns used to select elements on a web page so that styles can be applied to them. Understanding and mastering CSS selectors is crucial for effective web design and development.

Types of CSS Selectors

1. Basic Selectors

- Universal Selector (*):

- Selects all elements.
- Example: `*{ color: red; }`

- Type Selector (Element Selector):

- Selects all elements of a given type.

Hema Coding School

YouTube Link: <https://www.youtube.com/@HemaCodingSchool>

- Example: `p { font-size: 16px; }`

- Class Selector:

- Selects all elements with a specific class attribute.
- Example: `.classname { color: blue; }`

- ID Selector:

- Selects a single element with a specific ID attribute.
- Example: `#idname { margin: 10px; }`

- Attribute Selector:

- Selects elements based on the presence or value of an attribute.
- Example: `[type="text"] { border: 1px solid black; }`

2. Combinator Selectors

- Descendant Selector (` `):

- Selects elements that are descendants of a specified element.
- Example: `div p { color: green; }`

- Child Selector (>):

- Selects direct children of a specified element.
- Example: `div > p { color: orange; }`

- Adjacent Sibling Selector (+):

- Selects an element that is the next sibling of a specified element.
- Example: `h1 + p { margin-top: 0; }`

- General Sibling Selector (~):

- Selects all siblings of a specified element.
- Example: `h1 ~ p { color: gray; }`

3. Pseudo-class Selectors

- :hover:

- Selects an element when it is being hovered over.
- Example: a:hover { color: red; }

- :first-child:

- Selects the first child of a parent element.
- Example: p:first-child { font-weight: bold; }

- :nth-child(n):

- Selects the nth child of a parent element.
- Example: li:nth-child(2) { background-color: yellow; }

4. Pseudo-element Selectors

- ::before:

- Inserts content before an element's content.
- Example: p::before { content: "Note: "; }

- ::after:

- Inserts content after an element's content.
- Example: p::after { content: " End."; }

Interview Questions and Answers

1. Question: What is the difference between a class selector and an ID selector in CSS?

- Answer: A class selector (e.g., .classname) is used to select elements with a specific class attribute and can be applied to multiple elements. An ID selector (e.g., #idname) is used to select a single element with a specific ID attribute and should be unique within the document.

2. Question: How do you apply styles to all elements in a document?

Hema Coding School

YouTube Link: <https://www.youtube.com/@HemaCodingSchool>

- Answer: Use the universal selector (*). For example, `*{ margin: 0; padding: 0; }` applies zero margin and padding to all elements.

3. Question: How would you select all `<p>` elements that are direct children of a `<div>` element?

- Answer: Use the child combinator selector. For example, `div > p { color: blue; }`.

4. Question: Explain the use of the `:hover` pseudo-class.

- Answer: The `:hover` pseudo-class applies styles to an element when the user hovers over it with a pointing device (e.g., a mouse). For example, `a:hover { color: red; }` changes the text color of links to red when hovered.

5. Question: What does the `::before` pseudo-element do?

- Answer: The `::before` pseudo-element inserts content before an element's actual content. It is often used for adding decorative content or icons. For example, `p::before { content: "Note: "; }` adds the text "Note: " before each `<p>` element.

6. Question: How can you select an element based on an attribute value?

- Answer: Use the attribute selector. For example, `[type="text"] { border: 1px solid black; }` selects all elements with a type attribute equal to "text" and applies a border.

7. Question: Describe the `:nth-child(n)` pseudo-class and provide an example.

- Answer: The `:nth-child(n)` pseudo-class selects elements based on their position within a parent element. For example, `li:nth-child(2) { background-color: yellow; }` selects the second `` element within its parent and changes its background color to yellow.

8. Question: How do you select an element that is immediately followed by another specific element?

- Answer: Use the adjacent sibling selector (+). For example, `h1 + p { margin-top: 0; }` selects the `<p>` element that immediately follows an `<h1>` element and sets its top margin to zero.

CSS Style Types:

Hema Coding School

YouTube Link: <https://www.youtube.com/@HemaCodingSchool>

1. Inline Styles

Definition: Inline styles are applied directly to an HTML element using the style attribute.

Usage:

- Suitable for quick, one-time changes.
- Not recommended for large projects due to maintenance challenges and lack of reusability.

Example:

html

```
<p style="color: blue; font-size: 16px;">This is an inline styled paragraph.</p>
```

2. Internal (Embedded) Styles

Definition: Internal styles are defined within the <style> element inside the <head> section of an HTML document.

Usage:

- Useful for single-page websites or specific page styling.
- Helps keep styles in one place but still within the HTML document.

Example:

html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
  p {
```

```
    color: green;
```

```
    font-size: 18px;
```

```
}  
</style>  
</head>  
<body>  
  <p>This paragraph is styled using internal styles.</p>  
</body>  
</html>
```

3. External Stylesheets

Definition: External styles are defined in separate CSS files linked to the HTML document using the <link> element.

Usage:

- Ideal for large, multi-page websites.
- Promotes reusability and easier maintenance of styles.

Example:

HTML File:

```
html  
<!DOCTYPE html>  
<html>  
<head>  
  <link rel="stylesheet" type="text/css" href="styles.css">  
</head>  
<body>  
  <p>This paragraph is styled using an external stylesheet.</p>  
</body>  
</html>
```

CSS File (styles.css):

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

Interview Questions and Answers

1. Question: What is the difference between inline, internal, and external CSS?

- Answer: Inline CSS is applied directly to an element using the style attribute. Internal CSS is defined within a <style> element in the HTML document's <head>. External CSS is written in a separate CSS file and linked to the HTML document using the <link> element. Inline CSS is suitable for quick fixes, internal CSS for single-page styling, and external CSS for maintaining styles across multiple pages.

2. Question: When should you use inline styles in CSS?

- Answer: Inline styles should be used for quick, one-off style changes, such as testing or applying unique styles that are unlikely to be reused. They are not recommended for large projects due to poor maintainability and reusability.

3. Question: How do you link an external CSS file to an HTML document?

- Answer: You link an external CSS file to an HTML document using the <link> element within the <head> section of the HTML document. The href attribute specifies the path to the CSS file.

- Example:

html

```
<link rel="stylesheet" type="text/css" href="styles.css">
```

4. Question: What are the benefits of using external stylesheets?

Hema Coding School

YouTube Link: <https://www.youtube.com/@HemaCodingSchool>

- Answer: External stylesheets promote reusability and easier maintenance by keeping styles separate from HTML content. They allow for consistent styling across multiple pages and reduce redundancy, making it easier to manage large projects.

5. Question: How can you dynamically change CSS styles using JavaScript?

- Answer: You can dynamically change CSS styles using JavaScript by manipulating the DOM. You can directly set the style property of an element, use class names to apply predefined styles, or modify the CSS rules in a stylesheet.

- Example:

html

```
document.getElementById("myElement").style.color = "blue";
```

6. Question: What is the purpose of the <style> element in HTML?

- Answer: The <style> element is used to embed internal CSS within an HTML document. It allows you to define CSS rules that apply to the current HTML document and is typically placed within the <head> section.

- Example:

html

```
<style>
```

```
  body {
```

```
    background-color: lightblue;
```

```
  }
```

```
</style>
```