

scholareye.xyz



CHAPTER 08

Web Development

with HTML, CSS & JavaScript

Complete Exercise Solutions

MCQs • Short Questions • Long Questions

10

MCQs

10

Short Q

8

Long Q

Class 9 • Computer Science & Entrepreneurship

Study smarter at scholareye.xyz

1

Multiple Choice Questions

10 Questions

The correct option in every question is highlighted in green and marked with a tick, followed by a short explanation.

1

Which of the following tag is not a correct HTML tag?

- (a) `<div>`
- (b) `<span>`
- (c) `<head>`
- (d) `<footer>`

Check your textbook

All four tags listed here (`<div>`, `<span>`, `<head>`, `<footer>`) are valid HTML tags, so as printed none of the options is an incorrect tag. This looks like a misprint in the question or in one of the options. Please compare it with your book and confirm the answer with your teacher.

2

What does CSS stand for?

(a) Cascading Style Sheets ✓

(b) Computer Style Sheets

(c) Creative Style Sheets

(d) Colorful Style Sheets

Explanation

CSS stands for **Cascading Style Sheets**. It is the language used to describe the presentation and styling of a document written in HTML.

3

Which of the following tag is used to create a hyperlink in HTML?

(a) `<link>`(b) `<a>` ✓(c) `<href>`(d) `<nav>`**Explanation**

The `<a>` (anchor) tag is used to create hyperlinks. `<link>` is used to link external resources like CSS files. `<href>` is an attribute used inside the `<a>` tag, not a tag itself. `<nav>` is a semantic HTML5 tag for navigation sections.

4

Which property is used to change the background color in CSS?

(a) color

(b) background-color



(c) bgcolor

(d) background

Explanation

background-color is the correct CSS property. **color** changes the text color. **bgcolor** is an outdated HTML attribute, not a CSS property. **background** is a shorthand property that can set color, image, position, etc., but **background-color** is the specific property for the color.

5

Which HTML attribute is used to define inline styles?

(a) class

(b) style ✓

(c) font

(d) styles

Explanation

The **style** attribute is used to apply inline CSS directly to an HTML element (e.g., `<p style="color: red;">`). **class** is used to point to a class in a stylesheet. **font** is an outdated HTML tag. **styles** is incorrect.

6

Which of the following is the correct syntax for a CSS rule?

- (a) `selector {property: value;}` ✓
- (b) `selector: {property=value}`
- (c) `selector {property=value}`
- (d) `selector: {property: value;}`

Explanation

A standard CSS rule consists of a selector followed by a declaration block enclosed in curly braces `{}`. Inside the block, properties and values are separated by a colon `:` and end with a semicolon `;`. Therefore, `selector {property: value;}` is correct.

7

In JavaScript, which markup is used for comments?

(a) `/* */`

(b) `//`

(c) `<!-- -->`

(d) `/* */` and `//`



Explanation

JavaScript supports both single-line comments using `//` and multi-line comments using `/* */`. Option (c) `<!-- -->` is used for HTML comments. Therefore, option (d) is the most complete and correct answer.

8

How do you include JavaScript in an HTML document?

- (a) `<script src="script.js"></script>` ✓
- (b) `<java src="script.js"></java>`
- (c) `<js src="script.js"></js>`
- (d) `<code src="script.js"></code>`

Explanation

The `<script>` tag is the standard HTML tag used to embed or reference JavaScript. The `src` attribute is used to link to an external JavaScript file.

9

Which HTML tag is used to create an unordered list?

- (a) `<ol>`
- (b) `<ul>` ✓
- (c) `<li>`
- (d) `<list>`

Explanation

`<ul>` stands for Unordered List and creates a bulleted list. `<ol>` creates an Ordered (numbered) list. `<li>` defines individual list items. `<list>` is not a valid HTML tag.

10

Which tag is used to display a horizontal line in HTML?

(a) `<br>`

(b) `<hr>`



(c) `<line>`

(d) `<hline>`

Explanation

The `<hr>` tag stands for Horizontal Rule and is used to create a thematic break or horizontal line. `<br>` is used for a line break. `<line>` and `<hline>` are not valid HTML tags.

2

Short Questions

10 Questions

Q1

What is the purpose of the `<head>` tag in HTML?

The `<head>` tag is a container for metadata (data about data) and is placed between the `<html>` tag and the `<body>` tag. Its purpose is to hold information that is not displayed directly on the web page, such as the document title (`<title>`), character set declarations (`<meta charset="UTF-8">`), links to external CSS stylesheets (`<link>`), and JavaScript files (`<script>`). It essentially provides instructions to the browser about how to render the page.

Q2

Explain the difference between an ordered list and an unordered list in HTML.

An **ordered list** (`<ol>`) is used when the sequence of items matters. It automatically numbers each list item (e.g., 1, 2, 3 or A, B, C). An **unordered list** (`<ul>`) is used when the sequence does not matter, and it displays items with bullet points by default. Both use the `<li>` tag to define individual list items.

Q3**How do you add a comment in CSS?**

In CSS, comments are added using `/*` to start the comment and `*/` to end it. For example:

```
/* This is a CSS comment */  
p {  
    color: red; /* This is an inline comment */  
}
```

Q4**What are the different ways to apply CSS to an HTML document?**

There are three ways to apply CSS to HTML:

1. **Inline CSS:** Using the `style` attribute directly inside an HTML tag (e.g., `<p style="color: blue;">`).
2. **Internal CSS:** Placing CSS rules inside a `<style>` tag within the `<head>` section of the HTML document.
3. **External CSS:** Creating a separate `.css` file and linking it to the HTML document using the `<link>` tag in the `<head>` section. This is the most preferred method for large projects.

Q5

How can you include JavaScript in an HTML file?

JavaScript can be included in an HTML file in three ways:

1. **Inline:** Directly in an HTML event attribute (e.g., `<button onclick="alert('Hello')">`).
2. **Internal:** Placing JavaScript code inside a `<script>` tag within the `<head>` or `<body>` section.
3. **External:** Writing the JavaScript in a separate `.js` file and linking it using the `<script src="file.js"></script>` tag.

Q6

Describe the syntax for creating a hyperlink in HTML.

The syntax for creating a hyperlink uses the anchor tag `<a>`. The `href` attribute specifies the destination URL, and the text between the opening and closing tags is the clickable link text.

Syntax: `<a href="URL">Link Text</a>`

Example: `<a href="https://www.example.com">Visit Example</a>`

Q7**What is the function of the `<div>` tag in HTML?**

The `<div>` (division) tag is a generic block-level container used to group other HTML elements together. It does not have any inherent visual styling or semantic meaning. Its primary function is to serve as a container for applying CSS styles (using `class` or `id`) and manipulating elements with JavaScript, acting as a fundamental building block for web page layout.

Q8**How do you link an external CSS file to an HTML document?**

To link an external CSS file, you use the `<link>` tag inside the `<head>` section of the HTML document. The `rel` attribute specifies the relationship (stylesheet), and the `href` attribute specifies the path to the CSS file.

Example: `<link rel="stylesheet" href="styles.css">`

Q9**What is the use of the `<table>` tag in HTML?**

The `<table>` tag is used to create and display tabular data in a structured grid format consisting of rows and columns. Inside a `<table>`, the `<tr>` tag defines table rows, `<th>` defines header cells, and `<td>` defines standard data cells.

Q10**Explain the box model in CSS.**

The CSS Box Model is a fundamental concept that describes how every HTML element is rendered as a rectangular box. It consists of four layers, from inside to outside:

1. **Content:** The actual text, image, or media.
2. **Padding:** The transparent space between the content and the border.
3. **Border:** The visible line surrounding the padding and content.
4. **Margin:** The transparent space outside the border, separating the element from other elements.

3

Long Questions

8 Questions

Q1

Discuss the fundamental differences between HTML, CSS, and JavaScript in the context of web development.

In web development, these three technologies form the core foundation of the front-end, each serving a distinct purpose:

- **HTML (HyperText Markup Language):** It provides the **structure** and content of the web page. Think of it as the skeleton of a building. It defines elements like headings, paragraphs, images, and links.
- **CSS (Cascading Style Sheets):** It provides the **presentation** and styling. Think of it as the paint, wallpaper, and interior design of the building. It controls layout, colors, fonts, and responsiveness.
- **JavaScript:** It provides **behavior** and interactivity. Think of it as the electricity, plumbing, and elevators of the building. It allows the page to respond to user actions, manipulate the DOM, validate forms, and fetch data dynamically without reloading the page.

Together, HTML (structure), CSS (style), and JavaScript (behavior) create a complete, interactive, and visually appealing web experience.

Q2

Explain the process of setting up a development environment for web development. By discussing the necessary software and tools.

Setting up a web development environment requires a few essential tools:

- **Text Editor/IDE:** A code editor like Visual Studio Code (VS Code), Sublime Text, or Notepad++ is needed to write HTML, CSS, and JS code. VS Code is highly recommended due to its extensions (like Live Server) and integrated terminal.
- **Web Browser:** Modern browsers like Google Chrome, Mozilla Firefox, or Microsoft Edge are required to view and test the web pages. Chrome's Developer Tools (DevTools) are invaluable for debugging.
- **Version Control System (Git):** Git is used to track changes in code. A GitHub or GitLab account allows you to store code remotely and collaborate.
- **Node.js & npm (Optional):** For modern web development, Node.js is often needed to run build tools, package managers (npm), and frameworks like React or Vue.
- **Local Server (Optional):** Tools like XAMPP, WAMP, or the VS Code Live Server extension are used to simulate a web server environment locally.

Process:

1. Download and install a code editor (VS Code).
2. Install a modern web browser.
3. Create a project folder.
4. Open the folder in the editor.
5. Create `index.html`, `style.css`, and `script.js` files.
6. Start coding and preview in the browser.

Q3

Create a basic HTML page that includes a header, a paragraph, an image, and a hyperlink.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
    initial-scale=1.0">
  <title>My Basic Web Page</title>
</head>
<body>
  <!-- Header Section -->
  <header>
    <h1>Welcome to My Website</h1>
  </header>

  <!-- Paragraph Section -->
  <p>This is a simple paragraph demonstrating basic HTML
    structure. HTML is the backbone of the web.</p>

  <!-- Image Section -->
  

  <!-- Hyperlink Section -->
  <p>Visit <a href="https://www.example.com"
    target="_blank">Example.com</a> for more information.</p>
</body>
</html>
```

Q4

How do you style a table using CSS? Create a sample table and apply styles to it.

You can style a table using CSS by targeting the `<table>`, `<th>`, and `<td>` elements to control borders, padding, background colors, and text alignment. `border-collapse: collapse;` is commonly used to merge adjacent borders into a single border.

```
<!DOCTYPE html>
<html>
<head>
<style>
  table {
    width: 100%;
    border-collapse: collapse;
    font-family: Arial, sans-serif;
  }
  th, td {
    border: 1px solid #ddd;
    padding: 8px;
    text-align: left;
  }
  th {
    background-color: #4CAF50;
    color: white;
  }
  tr:nth-child(even) {
    background-color: #f2f2f2;
  }
  tr:hover {
    background-color: #ddd;
  }
</style>
</head>
<body>
  <table>
    <tr>
      <th>Name</th>
      <th>Age</th>
      <th>City</th>
    </tr>
```

```
<tr>
  <td>Alice</td>
  <td>24</td>
  <td>New York</td>
</tr>
<tr>
  <td>Bob</td>
  <td>30</td>
  <td>London</td>
</tr>
</table>
</body>
</html>
```

Q5

Describe the different CSS selectors and provide examples of each.

CSS Selectors are patterns used to select and style HTML elements. Common types include:

- **Element Selector:** Selects elements by their tag name.

Example: `p { color: blue; }` (Selects all `<p>` tags).

- **Class Selector:** Selects elements with a specific class attribute. Starts with a dot `.`

Example: `.my-class { font-size: 16px; }` (Selects all elements with `class="my-class"`).

- **ID Selector:** Selects a single element with a specific ID attribute. Starts with a hash `#`

Example: `#header { background: black; }` (Selects the element with `id="header"`).

- **Universal Selector:** Selects all elements on the page. `*`

Example: `* { margin: 0; padding: 0; }`

- **Descendant Selector:** Selects elements nested inside another element.

Example: `div p { color: red; }` (Selects all `<p>` inside a `<div>`).

- **Pseudo-class Selector:** Selects elements based on a specific state.

Example: `a:hover { color: orange; }` (Styles a link when the mouse hovers over it).

Q6

Explain the process of creating a responsive web page using CSS with the help of examples and explanations.

Creating a responsive web page means designing it to look good on all devices (desktops, tablets, and phones). This is primarily achieved using:

1. **Flexible Layouts:** Using percentages or relative units (`rem`, `em`, `vw`, `vh`) instead of fixed pixels for widths.
2. **Media Queries:** CSS rules that apply only when certain conditions are met (e.g., screen width).

Example:

```
/* Default styles for desktop */
.container {
  width: 80%;
  margin: auto;
}
/* Styles for screens smaller than 768px (tablets and phones) */
@media screen and (max-width: 768px) {
  .container {
    width: 100%;
    padding: 10px;
  }
}
```

```
}  
.sidebar {  
  display: none; /* Hide sidebar on mobile */  
}  
}
```

3. **Responsive Images:** Using `max-width: 100%;` and `height: auto;` on images so they scale down gracefully.
4. **Viewport Meta Tag:** Always include `<meta name="viewport" content="width=device-width, initial-scale=1.0">` in the HTML `<head>`.

Q7

Write a JavaScript function that changes the background color of a web page when a button is clicked. Provide the complete code and explain how it works.

```
<!DOCTYPE html>  
<html>  
<head>  
  <title>Background Color Changer</title>  
</head>  
<body>  
  <h1>Click the button to change the background color!</h1>  
  <button id="colorBtn">Change Color</button>  
  
  <script>  
    // Step 1: Select the button element  
    const button = document.getElementById('colorBtn');  
  
    // Step 2: Add an event listener to the button  
    button.addEventListener('click', function() {  
      // Step 3: Generate a random color or set a specific color  
      const colors = ['lightblue', 'lightgreen',  
        'lightcoral',  
        'lightyellow', 'lavender'];  
      const randomColor = colors[Math.floor(Math.random()  
        * colors.length)];
```

```
        // Step 4: Change the body's background color
        document.body.style.backgroundColor = randomColor;
    });
</script>
</body>
</html>
```

How it works

- We assign an `id` to the button to easily select it in JavaScript using `document.getElementById()`.
- We attach an event listener (`addEventListener`) to the button that listens for a `'click'`.
- When clicked, the function runs. It picks a random color from an array of predefined colors.
- Finally, it modifies the DOM by setting `document.body.style.backgroundColor` to the chosen color, instantly updating the page.

Q8

How do you add animations and transitions using CSS? Provide examples and explain the properties involved.

Transitions

Transitions allow property changes to occur smoothly over a specified duration. It requires a trigger (like `:hover`).

Properties: `transition-property` (what to animate),
`transition-duration` (how long),
`transition-timing-function` (speed curve),
`transition-delay`.

Example:

```
.box {  
  width: 100px;  
  height: 100px;  
  background: blue;  
  transition: background 0.5s ease, transform 0.5s;  
}  
.box:hover {  
  background: red;  
  transform: scale(1.2);  
}
```

Animations

Animations allow more complex, multi-step movements without needing a trigger. They use **@keyframes** to define the sequence of styles.

Properties: **animation-name**, **animation-duration**, **animation-iteration-count**, **animation-direction**.

Example:

```
@keyframes slideIn {  
  from { transform: translateX(-100%); opacity: 0; }  
  to { transform: translateX(0); opacity: 1; }  
}  
.animated-element {  
  animation: slideIn 1s ease-out forwards;  
}
```

The **@keyframes** rule defines the start (**from** or **0%**) and end (**to** or **100%**) states, and the **animation** property applies it to the element.



Thank You!

Keep learning, keep building, keep coding.

HTML Structure

The skeleton of every page

CSS Style

Colours, layout and design

JavaScript Behaviour

Makes pages come alive

scholareye.xyz

More chapter solutions, notes and study material