

Assignment No: 3

Unit = 3

1 Explain the structure of an HTML table provide an example of a basic table structure.

→ An HTML table is structure using several tags that define rows, columns, header's and table data. The basic components an HTML table are.

**<table>** :- The main container for the table.

**<tr>** :- Defines a row within the table.

**<th>** :- Defines a header cell within a row, usually bolded and centered by default.

**<td>** :- defines a cell that contains data within a row.

• Here's an example of A simple HTML table.

**<!DOCTYPE html>**

**<html lang="en">**

**<head>**

**<meta charset="UTF-8">**

**<meta name="viewport">**

**content="width=device width,**

**initial scale=1.0">**

**<title> Basic HTML table </title>**

**</head>**

**<body>**

```
<table border="1">
  <tr>
    <th> Name </th>
    <th> Age </th>
    <th> City </th>
  </tr>
  <tr>
    <td> Alice </td>
    <td> 24 </td>
    <td> New York </td>
  </tr>
</table>
</body>
</html>
```

Result :- This will produce a table with three columns (Name, age, city) and three rows of data. displaying each entry under its respective header.

2. How can you Align a table and its cell contents in HTML? Describe the CSS properties or HTML attributes used for Alignment and provide example demonstrating their Application.

→ Aligning tables and their contents in HTML can be done using CSS properties or in some cases older HTML attributes. Here's look at how you can align both the table itself and the cell contents.

browser compatibility ensures a seamless experience for users regardless of their chosen browser, preventing issues such as broken layout, missing functionality or styling inconsistencies that could affect the website's visibility and credibility.

- Here's why browser support is crucial in HTML development:

i) User Accessibility :- Some users rely on older browsers or browsers with specific accessibility features. Ensuring compatibility can make a site more inclusive.

ii) User Experience :- Users access website through various browsers and they expect the same experience regardless of which one they use.

iii) Brand Reputation :- Inconsistent functionality or performance across browsers can lead to perception of unreliability impacting a brand's credibility.

- Steps to ensure cross-browser compatibility:-

i) Use Standards :- Compliant HTML and CSS - stick to use standards. As these are generally supported by most modern browser's. Avoid browser-specific HTML, CSS or JavaScript that could lead to inconsistency.

ii) Polyfills and Transpiles :- polyfills and CSS prefixes help older browsers support newer web technologies. Allowing developers to write modern code while supporting legacy systems.

iii) Feature Detection :- Using tools like Modernizr allows developers to detect if a browser supports certain feature enabling.



Q.4

How can background support developer's can ensure that websites reach a broad audience enhance Accessibility deliver a better overall user's experience!

→ In HTML and CSS you can add background colors and images to a webpage using either inline styles or an external CSS file. Here are examples of both techniques.

i) Adding a background color:- You can add a background to the entire page or specific elements.

Code example:

```
<!DOCTYPE html>
<html lang = "en">
<head>
    <title> Background color </title>
    <style>
        body { background-color: lightblue; }
        content { 
            background-color: #f0f0f0;
            padding: 20px;
            border-radius: 8px;
        }
    </style>
</head>
<body>
    <div class = "content">
        <h1> Hello world </h1>
        <p> This is a section with a custom background color </p>
    </div>
</body>
</html>
```

```
</body>  
</html>
```

ii) Adding background image :- You can set a background image using CSS and control its position, size and respect behavior.

Code Example :-

```
<!DOCTYPE html>  
<html>
```

```
<head>
```

```
<title> Background Image's </title>
```

```
<style>
```

```
body { background-image: url('background.jpg'); }
```

```
</style>
```

```
</head>
```

```
<h1> welcome to My webpage </h1>
```

```
<p> This page have a background image </p>
```

```
</body>
```

```
</html>
```

Explain div and span tag in details

In HTML, `<div>` and `<span>` are two commonly used tags for grouping and organizing content. Both are generic container tags but they serve different purposes and are used in different contexts. Here's a detailed explanation of each:

i) `<div>` tag :-

Description:- The `<div>` tag is a block-level element used to group content.

Larger section of content together. It is often used to structure sections of a webpage such as header's, sidebars and main content areas.

**Usage :-** `<div>` is often used with `CSS` to apply styles, layout and organize content into logical sections.

**Attributes :-** you can use standard `HTML` attributes like `id`, `style` and `data`.

### EXAMPLE:

```

<!DOCTYPE HTML>
<html>
<head>
  <title> Div example </title>
  <style>   </style> </head>
<body>
  <div class = "container">
    <div class = "header">
      <div> website header </div>
    </div>
    <div class = "footer">
      <p> website footer </p>
    </div>
  </div>
</body>
</html>

```

- Description :-** The `<span>` tag is an inline element used to group smaller chunks of content within a block-level elements. It's primarily used to apply styling or behaviors to a part of text inline elements without affecting the layout.

Attribute's :- Similar to `<div>` it supports id, class, style and data attributes.

Example :-

```
<!DOCTYPE html>
<html>
<head>
  <title> Span Example </title>
</head>
<body>
  <p> welcome to our website we offer <span class="highlight">
    create deals </span> on a variety of products . Don't
    miss out: </p>
</body>
</html>
```

By understanding the purposes of `<div>` and `<span>` you can create a more structured organized and stylized HTML document.