MODULE 2 :- CSS

Que. 1) What are the benefits of using CSS?

Ans.:-

- **CSS saves time** − You can write CSS once and then reuse the same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.
- **Easy maintenance** − To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.
- **Global web standards** − Now HTML attributes are being deprecated and it is being recommended to use CSS. So it's a good idea to start using CSS in all the HTML pages to make them compatible with future browsers.
- **Platform Independence** − The Script offer consistent platform independence and can support latest browsers as well.

Que. 2) What are the disadvantages of CSS?

Ans.:-

- Cross-browser compatibility issues. What works with one browser might not always work with another, so web developers need to test for compatibility across multiple browsers.
- Confusion due to its many levels. CSS has many levels, which can be complicated for non-developers and beginners.
- Fixing not only one CSS file but also HTML tags associated with CSS selectors is often necessary.
- Different layout display.
- Vulnerability.

Que. 3) What is the difference between CSS2 and CSS3?

Ans.:-

Parameters	**CSS2**	**CSS3**
Released In	1998	1999
Styling Option	More than CSS	Large, along with the support for animations

Fonts	Web-safe fonts	Special fonts analogous to Google Fonts and Typecast
Selectors	Simple selectors	A sequence of simple selectors
Border box	Doesn't support border box property	Support border box property
Selector	Concept of simple selectors are present	Selectors were called as a sequence of simple selectors

Que. 4) Name a few CSS style components.

Ans.:-

- Color:- You can set the text color and background color of elements using properties like color and background-color.
- Typography:- CSS allows you to control the font family, size, weight, style, and spacing of text with properties like font-family, font-size, font-weight, font-style, line-height, and letter-spacing.
- Layout:- CSS provides properties for controlling the layout of elements, including width, height, margin, padding, and display.
- Positioning:- You can control the positioning of elements using properties like position, top, right, bottom, and left. These are often used for creating fixed headers or footers, sticky navigation bars, etc.
- Border and Box Shadow:- You can add borders to elements with properties like border, border-width, border-color, and border-radius. Additionally, you can apply box shadows using box-shadow.

Que. 5) What do you understand by CSS opacity?

Ans.:- The opacity property sets the opacity level for an element. The opacity-level describes the transparency-level, where 1 is not transparent at all, 0.5 is 50% see-through, and 0 is completely transparent. When using the opacity property to add transparency to the background of an element, all its child elements become transparent as well. This can make the text inside a fully transparent element hard to read. If you do not want to apply opacity to child elements, use RGBA color values instead.

EX.:-
```
<head>
<style>
 div {
  background-color: red;
opacity: 0.42;
```

```
  color: black;
}
</style>
</head>
<body>
<h1>The opacity Property</h1>
<div>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat pulvinar, at pulvinar felis blandit...</div>
</body>
```

Que. 6) How can the background color of an element be changed?

Ans.:- The background-color property of CSS is used to set the background of an element. We can set background color by selecting the element by its class name of id name and then apply the background-color property on it to set the background color.

EX:-
```
<head>
<style>
p {
  background-color: coral; }
h1{ background-color: red}
</style>
</head>
<body>
<h1>The background-color Property</h1>
<p>The background color can be specified with a color name.</p>
</body>
```

Que. 7) How can image repetition of the backup be controlled?

Ans.:-

- To control image repetition of the backup in CSS, we can use the background-repeat property.
- This property is used to repeat the background image both horizontally and vertically.
- We can use the “no-repeat” value for the background-repeat property if you do not want to repeat an image, in this case, the image will display once.
- We can use background-repeat property in value added and image repetition.

Ex.:-
```
<style>
body {
background-image: url("img.jpeg")
background-repeat: no-repeat;
}
</style>
```

Que. 8) What is the use of the background-position property?

Ans.:-

- The background-position property in CSS is used to set the starting position of a background image.
- By default, a background-image is placed at the top-left corner of an element, and repeated both vertically and horizontally.
- The position that is relative to the positioning layer can be set by using the background-origin property.
- For example, if we have an image that is 100px wide and 200px high and we want it to be positioned 10px from the left and 20px from the top of the element.

Ex.:-
```
<style>
body {
background-image: url("img.jpeg")
background-repeat: no-repeat;
background-position: 10px 20px;
}
</style>
```

Que. 9) Which property controls the image scroll in the background?

Ans.:-

- The background-attachment property in CSS is used to specify the kind of attachment of the background image with respect to its container.
- It can be set to scroll or make it remain fixed.
- Background-attachment: scroll, fixed, local, initial, inherit.

Ex.:-
```
body {
background-image: url('background-image.jpg');
background-repeat: no-repeat;
background-size: cover;
background-attachment: fixed;
}
```

Que. 10) Why should background and color be used as separate properties?

Ans.:-

- **Responsive Design:-** Separating background and color properties allows for more responsive design. You can change background images or colors independently based on screen size or other conditions.
- **CSS Customization:-** When working with frameworks or third-party stylesheets, separating background and color properties can make it easier to customize and override specific styles without affecting other aspects of the element's appearance.

Ex:-
```
div {
  background-image: url('background-image.jpg');
  background-color: lightblue;
  background-repeat: no-repeat;
  background-position: center;
  background-attachment: fixed;
}
```

```
.text-container {
  color: #333;
}
```

Que. 11) How to center block elements using CSS1?

Ans.:-

- **Block Elements Behavior:** Any element can be set to behave like a block by setting their display property to the block "display: block". There are some other elements like headers, div tags which are default blocks. So, they take the full line or full width of the web page or container holding it. Even if our content is 20% of the width of the webpage still the block property will reserve the full 100% width of the web page or container holding it.

- **How to center these block elements:** We have seen how this block element behaves, we observe that as they take full line width, to center them we only have the margin property that is controlling them horizontally. The margin can control the position of the block element both horizontally and vertically. To center them, we can adjust the margin property such that it is placed in the center.

Ex.:-

```
div {
width: 500px;
border: 2px solid black;
margin-left: auto;
margin-right: auto;
background-color: powderblue;
text-align: center;
}
```

Que. 12) How to maintain the CSS specifications?

Ans.:-

- **Documentation:-** Document your CSS code thoroughly. This includes comments within your stylesheet to explain the purpose of rules, classes, and IDs. Create documentation that outlines the structure of your CSS, naming conventions, and usage guidelines.
- **CSS Preprocessors:-** Consider using CSS preprocessors like Sass or Less. They offer features like variables, mixings, and nesting, which can help maintain styles more efficiently and make your codebase more organized.
- **Modularization:-** Divide your CSS into modular components. Use separate stylesheets or CSS files for different sections or components of your website. This makes it easier to find and update styles related to specific elements or features.
- **Code Reviews:-** Encourage code reviews within your development team. Peer reviews can help identify potential issues, ensure code quality, and promote knowledge sharing.

Que. 13) What are the ways to integrate CSS as a web page?

Ans.:-

- **Inline CSS:** Inline CSS is added directly to individual HTML elements using the style attribute. It's not typically recommended for large-scale styling due to its limited maintainability and reusability, but it can be useful for small, one-off styling adjustments.

Ex.:- Hello, Good Morning.

- **Internal CSS:** Internal CSS is placed within a <style> element in the <head> section of an HTML document. This method is suitable for small to medium-sized web pages or when you want to apply styles to a single page.

Ex.:-

```
<head>
   <style>
      p {
         color: blue;
         font-size: 16px;
      }
   </style>
</head>
<body>
   <p> Hello, Good Morning.</p>
</body>
```

- **External CSS:** External CSS is the most common and recommended method for styling web pages. You create a separate CSS file and link it to your HTML document using the <link> element in the <head> section. This approach promotes reusability and maintainability of styles across multiple pages.

♦ **Styles.css:-**

```
p {
    color: blue;
    font-size: 16px;
   }
```

♦ **Styles.html:-**

```
<html>
<head>
<link rel="stylesheet"  href="Styles.css">
</head>
<body>
   <p>Hello, Good Morning.</p>
</body>
</html>
```

Que. 14) What is embedded style sheets?

Ans.:- Embedded style sheets, also known as internal style sheets or inline style sheets, are a way to define CSS (Cascading Style Sheets) styles directly within an HTML document. CSS is a styling language used to control the visual presentation and layout of web pages. While external style sheets are defined in separate .css files and linked to HTML documents, embedded style sheets are written directly within the HTML document's <style> tags.

Ex.:-

```
<!DOCTYPE html>
<html>
<head>
   <title>Embedded Style Sheet Example</title>
   <style>
      body {
         font-family: Arial, sans-serif;
         background-color: red;
      }

      h1 {
         color: #333;
      }
      p {
         font-size: 16px;
      }
   </style>
</head>
<body>
   <h1> Heading</h1>
   <p>This is a paragraph with some text.</p>
</body>
</html>
```

Que. 15) What are the external style sheets?

Ans.:- External style sheets are separate CSS (Cascading Style Sheets) files that contain style rules and definitions for web page elements. These files are created independently of the HTML documents and are then linked to one or more HTML documents. External style sheets offer several advantages over embedded or inline styles, including Reusability, Modularity, Easier Maintenance.

Ex.:-

- **Styles.css:-**

```
p {
    color: blue;
    font-size: 16px;
   }
```

- **Styles.html:-**

```
<html>
<head>
<link rel="stylesheet"  href="Styles.css">
</head>
```

```
<body>
   <p>Hello, Good Morning.</p>
</body>
</html>
```

Que. 16) What are the advantages and disadvantages of using external style sheets?

- **Advantages of External Style Sheets:-**

– **Modularity:-** External style sheets allow you to create a single stylesheet that can be applied to multiple web pages. This promotes consistency in design across your entire website.

– **Separation of Concerns:-** External style sheets separate the content (HTML) from the presentation (CSS). This separation follows the principle of "separation of concerns," making it easier to manage and maintain both HTML and CSS code.

– **Efficiency:-** Browsers can cache external style sheets, which can speed up page loading times for returning visitors because the stylesheet only needs to be loaded once.

- **Disadvantages of External Style Sheets:-**

– **HTTP Request:-** Each external stylesheet requires an additional HTTP request, which can slightly increase page load times, particularly for the first-time visitor. However, this can be mitigated with techniques like minification and bundling.

– **Dependency:-** If the external style sheet fails to load for any reason (e.g., a broken link or network issues), the page's styling may be compromised, leading to a less-than-optimal user experience.

– **Cross-Origin Considerations:-** When linking to external stylesheets hosted on different domains, you may encounter cross-origin restrictions. This can be addressed with CORS (Cross-Origin Resource Sharing) headers on the server hosting the stylesheet.

Que. 17) What is the meaning of the CSS selector?

Ans.:- A selector is a pattern that is used to select and target HTML elements within an HTML document. CSS selectors are used to apply styles and formatting to specific elements or groups of elements on a web page. Selectors define which HTML elements should receive the styling rules defined in the CSS rules.

- **Type Selector:** This selector targets all instances of a specific HTML element type.
- **Class Selector:** This selector targets HTML elements with a specific class attribute. Class selectors are denoted by a dot (.) followed by the class name.
- **ID Selector:** This selector targets a single HTML element with a specific ID attribute. ID selectors are denoted by a hash (#) followed by the ID name.
- **Child Selector:** This selector targets an element that is a direct child of another element. It is denoted by a greater than (>).
- **Attribute Selector:** This selector targets elements with specific attributes and attribute values.

- **Pseudo-Class Selector:** This selector targets elements based on their state or position in the document.

Que. 18) What are the media types allowed by CSS?

Ans.:- Media queries can be used to check many things, such as:

- width and height of the viewport
- width and height of the device
- orientation (is the tablet/phone in landscape or portrait mode?)
- resolution

Screen: This media type is used for typical computer screens, including desktops, laptops, and some tablets. You would use this to define styles for the standard web browsing experience.
Print: This media type is used for print media, such as when a web page is printed. It allows you to define styles specifically for printed pages.
Speech: This media type is used for screen readers and other speech-based assistive technologies. It allows you to define styles that are appropriate for audio-based presentations.
Ex.:-

```
<html>
<head>
<style>
body {
  background-color: pink;
}
@media screen and (min-width: 480px) {
  body {
    background-color: lightgreen;
  }
}
</style>
</head>
<body>
<h1>Resize the browser window to see the effect! </h1>
<p>The media query will only apply if the media type is screen, and the viewport is 480px wide or wider. </p>
</body>
</html>
```

Que. 19) What is the rule set?

Ans.:- A rule set, often simply referred to as a "rule," is a fundamental structure that defines how specific HTML elements should be styled. A rule set consists of two main parts: a selector and a declaration block.

- Here's the basic structure of a CSS rule set:

```
selector {
property: value;
}
```

❖ **Selector:** The selector specifies which HTML element or elements the rule set should apply to. Selectors can target elements by their type, class, ID, attributes, or their position in the document hierarchy.

- **Type Selector:** p selects all <p> (paragraph) elements.
- **Class Selector:** .my-class selects all elements with the class "my-class."
- **ID Selector:** #my-element selects the element with the ID "my-element."
- **Descendant Selector:** .container p selects all <p> elements that are descendants of elements with the class "container."
- **Pseudo-Class Selector:** a:hover selects <a> elements when they are being hovered over.

- **Declaration Block:** The declaration block is enclosed within curly braces {} and contains one or more property-value pairs. Each property defines a specific aspect of the element's style, and the associated value specifies the desired styling for that property.

```
selector {
property1: value1;
property2: value2;
}
```

Ex.:-

```
p {
color: blue;
font-size: 16px;
}
```