WEB DESIGNING Assignment

MODULE: 2 (CSS)

1. What are the benefits of using CSS?

Ans. Following are the benefits of using CSS:

- External CSS provides a clean code due by separating HTML and CSS files
- Flexibility to modify design without changing HTML structure
- Uniform design through a single CSS file linked to multiple pages
- Easier design updates by changing one file
- Responsive design with media queries for different devices
- Advanced styling options
- Reusable classes and IDs across elements and pages
- Supports cross browsing compatibility

2. What are the disadvantages of CSS?

Ans. Following are the disadvantages of using CSS:

- Managing large CSS files can be cumbersome
- Large or poorly optimized CSS files can slow down page load times
- Overuse of styles and complex selectors can impact rendering performance
- Some advanced layout and design capabilities may require additional tools
- Reliance on external CSS files can lead to problems if the file fails to load or is unavailable
- CSS lacks the ability to implement logic, conditions or loops directly limiting its dynamism

3. What is the difference between CSS2 and CSS3?

Feature	CSS2	CSS3
Selectors	Basic selectors(class, ID, child)	Advanced selectors(attribute selectors, pseudo-classes, pseudo-elements)
Box model and layout	Basic box model, fixed and absolute positioning	Advanced layouts with flexbox and grid
Media queries	Limited support	Full support enabling responsive design
Transitions, Animations, Transformations, Shadow effects, Grid layout	Not supported	Supported
Backgrounds, Borders, Color and Opacity, Text effects and Fonts	Basic properties	Advanced properties like background-size, border-radius, background-image, RGBA, HSLA color models and opacity property, text-shadow, custom-fonts, text-overflow, word-wrap, text-stroke

4. Name a few CSS style components?

Ans. Below are a few CSS style components:

1. **Selectors:**
    - Class Selector (.classname)
    - ID Selector (#idname)
    - Element Selector (element)
    - Attribute Selector ([attribute=value])
2. **Box Model:**
    - Margins (margin)
    - Borders (border)
    - Padding (padding)
    - Content (width, height)
3. **Typography:**
    - Font Family (font-family)
    - Font Size (font-size)
    - Font Weight (font-weight)
    - Line Height (line-height)
    - Text Alignment (text-align)
    - Text Decoration (text-decoration)
    - Text Transform (text-transform)
4. **Color:**
    - Color (color)
    - Background Color (background-color)
    - Opacity (opacity)
5. **Positioning:**
    - Position (position: static, relative, absolute, fixed, sticky)
    - Top, Right, Bottom, Left (top, right, bottom, left)
    - Z-Index (z-index)
6. **Flexbox:**
    - Display (display: flex)
    - Flex Direction (flex-direction)
    - Justify Content (justify-content)
    - Align Items (align-items)
    - Flex Wrap (flex-wrap)
7. **Grid:**
    - Display (display: grid)
    - Grid Template Columns (grid-template-columns)
    - Grid Template Rows (grid-template-rows)
    - Grid Gap (grid-gap)
    - Grid Area (grid-area)

8. **Backgrounds:**
    - Background Image (background-image)
    - Background Position (background-position)
    - Background Size (background-size)
    - Background Repeat (background-repeat)
9. **Borders and Outlines:**
    - Border Style (border-style)
    - Border Width (border-width)
    - Border Color (border-color)
    - Border Radius (border-radius)
    - Outline (outline)
10. **Transitions and Animations:**
    - Transition (transition)
    - Animation (animation)
    - Keyframes (@keyframes)
11. **Transformations:**
    - Transform (transform: rotate, scale, translate, skew)
12. **Shadows:**
    - Box Shadow (box-shadow)
    - Text Shadow (text-shadow)

5. What do you understand by CSS opacity?

Ans. CSS opacity is a property which controls the transparency of an element. It defines how transparent or opaque an element is, with values ranging from 0 to 1, where 0 means completely transparent (invisible) and 1 means completely opaque (fully visible).

Syntax:

```
element{
        opacity:value;
 }
```

<u>For example:</u> (for 50% transparent)

```
.semi-transparent{
        opacity:0.5;
 }
```

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

Ans. The background color of an element can be changed using the `background-color` property in CSS, in several ways as follows:

```
      .named-color {
     background-color: blue;
   }
   .hex-color {
     background-color: #00ff00; /* Green color */
   }
```

```
.rgb-color {
  background-color: rgb(0, 0, 255); /* Blue color */
}
.rgba-color {
  background-color: rgba(255, 165, 0, 0.7); /* 70% transparent orange */
}
.hsl-color {
  background-color: hsl(120, 100%, 50%); /* Green color */
}
.hsla-color {
  background-color: hsla(240, 100%, 50%, 0.3); /* 30% transparent blue */
}
```

7. How can image repetition of the background be controlled?

Ans. Image repetition in the background can be controlled using the `background-repeat` property in CSS. This property allows to specify how the background image should be repeated (or not repeated) within an element. Below are the possible ways to use this property:

Syntax:

```
element {
      background-repeat: value;
}
```

1. repeat: the background image is repeated both horizontally and vertically.

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

2. repeat-x: the background image is repeated horizontally only.

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

3. repeat-y: the background image is repeated vertically only.

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

4. no-repeat: the background image is not repeated and will be displayed only once.

```
.no-repeat {
  background-image: url('image.jpg');
  background-repeat: no-repeat;
}
```

5. space: the background image is repeated as much as possible and with extra space distributed around the images.

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

6. round: the background image is repeated and scaled so that it fits the container exactly.

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

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

Ans. The background-position property in CSS is used to specify the initial position of a background image within an element. It allows controlling the placement of background image relative to the element's boundaries.

Syntax:

```
element {
        background-position: x y;
}
```

- x – specifies the horizontal position
- y – specifies the vertical position
- values for x and y can be specified in different ways as follows:

1. **keywords** like top, bottom, left, right, center is used to define the position relating to x and y

<u>Example:</u>

```
.example {
  background-image: url('image.jpg');
  background-position: top left; /* Aligns the image to the top left corner */
}
```

2. **percentage** values between 0% and 100% can be used for both horizontal and vertical position

<u>Example:</u>

```
.example {
  background-image: url('image.jpg');
  background-position: 50% 50%; /* Centers the image */
}
```

3. **length:** use of specific units like `px`, `em`, etc

```
.example {
  background-image: url('image.jpg');
  background-position: 10px 20px; /* Positions the image 10px from the left and 20px from the top */
}
```

4. **combination:** can combine different units and keywords

```
.example {
  background-image: url('image.jpg');
  background-position: 50px center; /* 50px from the left and centered vertically */
}
```

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

Ans. The `background-attachment` property controls the scrolling behavior of a background image. It specifies whether a background image scrolls with the rest of the page's content or remains fixed as the viewport scrolls. This is achieved using the following values:

1. **scroll:** this default behavior allows the image to scroll alongwith the content of the element

```
.example {
  background-image: url('image.jpg');
  background-attachment: scroll;
}
```

2. **fixed:** the background image remains fixed relative to the viewport and does not allow it to scroll with the content

```
.example {
  background-image: url('image.jpg');
  background-attachment: fixed;
}
```

3. **local:** the background image scrolls with the element's content but not the entire page

```
.example {
  background-image: url('image.jpg');
  background-attachment: local;
}
```

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

Ans. Background and color should be used as separate properties due to following reasons:

- Quick identification of which properties are affecting the background styling and which are affecting the text color, making the CSS code easier to read and understand.
- Allows independent control over background styles and text colors.
- Reduces redundancy and improves code efficiency.
- Enhances flexibility for conditional styling and design customization.
- Ensures broader browser compatibility.
- Promotes better user experience and customization options.

11. How to center block elements using CSS?

Ans. To center block elements using CSS, there are few approaches as follows, depending on layout needs:

a) **Using margin:**

```
.box {
    margin: 50px auto;}
```

b) **Using Flexbox:**

```
. box {
    display: flex;
justify-content: center;
    align-items: center;}
```

c) **Using grid:**

```
. box {
    display: flex;
    place-items: center;}
```

12. How to maintain CSS specifications?

Ans. Following are few ways to maintain CSS specifications:

- Structuring CSS files logically and grouping related styles together. Use of separate files for different components or sections of your website.
- Adopting a naming convention to ensure clarity and consistency in class names.
- Using selectors that are as specific as it helps to prevent unintended style overrides and makes the stylesheet easier to understand.
- Documenting complex or critical sections of CSS with comments to explain their purpose or usage. This aids in understanding and maintenance for future developers.
- Implementing responsive design principles from the start to handle various screen sizes and devices effectively.
- Ensuring stylesheets work correctly across different browsers and devices to maintain consistent appearance and behavior.

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

Ans. Integrating CSS as a web page can be done in following ways:

a) **Inline CSS:** applying CSS directly within the HTML element using the `style` attribute.
eg. <p style="color: blue; font-size: 20px;">This is an inline styled paragraph.</p>

b) **Internal CSS:** including CSS within the `<style>` tag inside the `<head>` section of HTML document.

```
eg. <html>
     <head>
         <style>
                 body {
                         font-family: Arial, sans-serif;}
                 .header {
                         color: blue;
                         font-size: 24px;}
         </style>
    </head>
    <body>
    <h1 class="header">This is a header</h1>
    </body>
    </html>
```

c) **External CSS:** linking to an external CSS file using the `<link>` tag inside the `<head>` section.

```
<html>
<head>
  <link rel="stylesheet" type="text/css" href="styles.css">
</head>
<body>
  <h1 class="header">This is a header</h1>
</body>
</html>
```

14. What is embedded style sheet?

Ans. An embedded style sheet is a way to include CSS directly within an HTML document. This is done by placing CSS rules inside a <style> tag within the <head> section of the HTML file. Embedded stylesheets are useful for applying styles to a specific page without the need for an external CSS file.

```
eg. <html>
     <head>
           <style>
                body {
                    font-family: Arial, sans-serif;}
                .header {
                     color: blue;
                     font-size: 24px;}
            </style>
       </head>
       <body>
       <h1 class="header">This is a header</h1>
       </body>
     </html>
```

15. What are the external style sheets?

Ans. To use an external style sheet, you include a <link> tag within the <head> section of your HTML document. The href attribute of the <link> tag specifies the path to the CSS file.

```
eg. <html>
     <head>
     <link rel="stylesheet" type="text/css" href="styles.css">
     </head>
     <body>
        <h1 class="header">This is a header</h1>
     </body>
     </html>
```

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

Advantages	Disadvantages
Centralized control: changes reflect across all linked pages	Additional HTTP request
Easy updates: update styles in one place	If external CSS is missing or incorrect, the page may not render correctly
Reusability, consistency, modularity	Managing multiple external CSS files can become complex in larger projects
Keeps HTML content separate from presentation	Debugging styles can be more challenging as you need to trace back to the external stylesheet
Keeps HTML size smaller and cleaner	

17. What is the meaning of CSS selector?

Ans. A CSS selector is a pattern used to select and style HTML elements within a web page. CSS selectors define which HTML elements the CSS rules will be applied to. There are various types of selectors, each serving a specific purpose and providing different levels of specificity for targeting elements.

1. **Universal selector (*):**
   - Selects all elements in the document

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

2. **Element selector**
   - Selects all elements of a given type

```
p {
   color: blue;
}
```

3. **Class selector**
   - Selects all elements with a specific class attribute

```
.highlight {
   background-color: yellow;
}
```

4. **Id selector**
   - Selects a single element with a specific ID attribute

```
#header {
   font-size: 24px;
}
```

5. **Attribute selector**
   - Selects elements based on the presence or value of a specific attribute.

```
[type="text"] {
   border: 1px solid #ccc;
}
```

6. **Pseudo-class selector**
    - Selects elements based on their state or position in the document.

```
a:hover {
   color: red;
}
```

7. **Pseudo-element selector**
    - Selects and styles specific parts of elements.

```
p::first-line {
   font-weight: bold;
}
```

8. **Descendant selector**
    - Selects elements that are descendants of a specified element.

```
div p {
   color: green;
}
```

9. **Child selector**
    - Selects elements that are direct children of a specified element.

```
ul > li {
   list-style-type: none;
}
```

10. **Adjacent sibling selector**
    - Selects an element that is immediately preceded by a specified element.

```
h1 + p {
   margin-top: 0;
}
```

11. **General sibling selector**
    - Selects all elements that are siblings of a specified element.

```
h1 ~ p {
   color: gray;
}
```

18. What are the media types allowed by CSS?

Ans. CSS allows several media types, each designed to tailor the presentation of content to different devices or environments which are as follows:

1. **all**: Suitable for all devices
   syntax: @media all { ... }
2. **print**: Intended for printed material and for documents viewed on a screen in print preview mode.
   syntax: @media print { ... }
3. **screen**: Used primarily for computer screens, tablets, smartphones, etc.
   syntax: @media screen { ... }
4. **speech**: Intended for speech synthesizers.
   syntax: @media speech { ... }

19. What is the rule set?

Ans. In CSS, a rule set (often simply called a rule) is a fundamental component that defines how elements on a web page should be styled. A rule set consists of a selector and a declaration block.

Components of a Rule Set

1. **Selector**: This specifies which HTML elements the rule applies to. It can be an element type, class, ID, or any other valid CSS selector.
2. **Declaration Block**: This contains one or more declarations, each consisting of a property and a value, enclosed in curly braces {}.

Syntax:

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

Example:

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

20. Create layouts…practical question.

Ans. Submitted separately in the folder…. The file name is assignment_card.html.