**Assignment- 2**

1.What are the benefits of using CSS?

There are a number of benefits of CSS, including:

**1) Faster Page Speed**

More code means slower page speed. And CSS enables you to use less code. CSS allows you to use one CSS rule and apply it to all occurrences of a certain tag within an HTML document.

**2) Better User Experience**

CSS not only makes web pages easy on the eye, it also allows for user-friendly formatting. When buttons and text are in logical places and well organized, user experience improves.

**3) Quicker Development Time**

With CSS, you can apply specific formatting rules and styles to multiple pages with one string of code. One cascading style sheet can be replicated across several website pages. If, for instance, you have product pages that should all have the same formatting, look, and feel, writing CSS rules for one page will suffice for all pages of that same type.

**4) Easy Formatting Changes**

If you need to change the format of a specific set of pages, it’s easy to do so with CSS. There’s no need to fix every individual page. Just edit the corresponding CSS stylesheet and you’ll see changes applied to all the pages that are using that style sheet.

**5) Compatibility Across Devices**

Responsive web design matters. In today’s day and age, web pages must be fully visible and easily navigable on all devices. Whether mobile or tablet, desktop, or even smart TV, CSS combines with HTML to make responsive design possible.

2. What are the disadvantages of CSS?

There are a few downsides while using CSS. One must know these disadvantages so that he or she is aware and takes care of them while designing a website.

### **1. Confusion due to many CSS levels**

Beginners are more vulnerable to this issue. They might get confused while opting to learn CSS as there are many levels of CSS such as CSS2, CSS3, etc.

### **2. Cross-Browser Issues**

Different browsers work differently. So, you have to check that changes implemented in the website via CSS codes are reflected properly among all browsers.

### **3. Security Issues**

Security is important in today’s world driven by technology and data. One of the major disadvantages of CSS is that it has limited security.

### **4. Extra Work for Developers**

Design services are required to consider and test all CSS codes across different browsers for compatibility. Due to developers testing compatibility for different browsers, their workload increases.

3.What is the difference between CSS2 and CSS3?

1. CSS was originally released in 1996 and consists of properties for adding font properties such as typeface and emphasis color of text, backgrounds, and other elements. CSS2 was released in 1998 with added styles for other media types so that it can be used for page layout designing. CSS3 was released in 1999 and presentation-style properties were added in it that allows you to build a presentation from documents.
2. Unlike CSS2, which was comprised of a single document, CSS3 has its specifications divided into many individual modules, which makes CSS3 a whole lot easier to handle.
3. With CSS3, the designers can now use special fonts, like those available in Google Fonts and Typecast. Earlier, with CSS and CSS2, designers could only use “web-safe fonts” for being 100% sure to use fonts that would always display the same on every machine.
4. While CSS and CSS2 had ‘simple selectors’, CSS3 calls the components as ‘a sequence of simple selectors’.
5. CSS3 came up with some key web design considerations like rounded borders that help in rounding up the borders without any hassle. This turned out to be a huge plus point for developers who were struggling with initial versions of CSS borders.
6. CSS3 has the capability to split text sections into multiple columns so that it can be read like a newspaper. In CSS2, the developers had difficulty because the standard was not equipped with automatically breaking the text so that it fits within a box.

4.Name a few CSS style components?

**At its most basic level, CSS consists of two components:**

* Properties: These are human-readable identifiers that indicate which stylistic features you want to modify. For example, font-size , width , background-color .
* Values: Each property is assigned a value. This value indicates how to style the property.

5.What do you understand by CSS opacity?

The **opacity** CSS property sets the opacity of an element. Opacity is the degree to which content behind an element is hidden, and is the opposite of transparency.

C1=[R1,G1,B1] is the foreground pixel color. C2=[R2,G2,B2] is the background pixel color. p1 is the opacity percentage of the foreground pixel. p2 is the opacity percentage of the background pixel.

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

To add background color in HTML, **use the CSS background-color property**. Set it to the color name or code you want and place it inside a style attribute. Then add this style attribute to an HTML element, like a table, heading, div, or span tag.

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

In this article, we will see how an image repetition of the backup is controlled in CSS. This task can be achieved by using the *background-repeat property* that will help us to control the repetition of the image.

The **background-repeat property** in CSS is used to repeat the background image both horizontally and vertically. It also decides whether the background image will be repeated or not.

**Syntax:**

background-repeat: repeat|repeat-x|repeat-y|no-repeat|initial|inherit;

**Example 1:** In the example, we will make use of the repeat-x to repeat the image in the horizontal direction.

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

The **background-position** property in CSS is mainly used to sets the initial position for the background image ie., it is used to set an image at a certain position. The position that is relative to the positioning layer, can be set by using the [background-origin](https://www.geeksforgeeks.org/css-background-origin-property/) property.

**Syntax:**

background-position: value;

**Note:**The [background-image](https://www.geeksforgeeks.org/css-background-image-property/) is placed default to the top-left corner of an element with a repetition on both horizontally & vertically.

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

To set the scrolling of an image in the background, use the background-attachment property.

## Example

You can try to run the following code to learn how to work with the background-attachment property:

<!DOCTYPE html>

<html>

   <head>

      <style>

         body {

            background-image: url('/css/images/css.jpg');

            background-repeat: no-repeat;

            background-attachment: fixed;

            background-attachment:scroll;

         }.

      </style>

   </head>

   <body>

      <p>The background-image is fixed. Try to scroll down the page.</p>

      <p>The background-image is fixed. Try to scroll down the page.</p>

      <p>The background-image is fixed. Try to scroll down the page.</p>

      <p>The background-image is fixed. Try to scroll down the page.</p>

      <p>The background-image is fixed. Try to scroll down the page.</p>

      <p>The background-image is fixed. Try to scroll down the page.</p>

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

here are two reasons behind this:

* It enhances the legibility of style sheets. The background property is a complex property in CSS, and if it is combined with color, the complexity will further increase.
* Color is an inherited property while the background is not. So this can make confusion further.

11.How to center block elements using CSS1?

here are two ways of centering block level elements:  
  
1. By setting the properties margin-left and margin-right to auto and width to some explicit value:  
  
BODY {width: 30em; background: cyan;}  
P {width: 22em; margin-left: auto; margin-right: auto}  
  
In this case, the left and right margins will each be four ems wide, since they equally split up the eight ems left over from (30em - 22em). Note that it was not necessary to set an explicit width for the BODY element; it was done here to keep the math clean.  
  
Another example:  
  
TABLE {margin-left: auto; margin-right: auto; width: 400px;}  
In most legacy browsers, a table's width is by default determined by its content. In CSS-conformant browsers, the complete width of any element (including tables) defaults to the full width of its parent element's content area. As browser becaome more conformant, authors will need to be aware of the potential impact on their designs.

12. How to maintain the CSS specifications?

The Specification defines how CSS properties should be implemented by browser vendors along with detailed algorithms, code samples and tabular information.

The Specification also include:

* The syntax and data types of the language
* Detailed explanation on CSS Selectors
* How you can assign values to properties
* The Cascade (the "C" in CSS)
* How inheritance works
* The Box Model e.t.c

Explanation on some of these topic are short and easy to understand while others are explained in great detail.

The Specification also specify how stylesheets can be included in your web document and how to target specific media e.g print or screen.

The CSS Specification **prior to CSS3 was a single Specification**, CSS3 on the other hand **is divided into Modules** which are **Independent Specifications** that can be worked on by different author(s) at different paces, that's why we have Selector Level 3 Specification, CSS Color 4, CSS Backgrounds and so on. Some of these modules are revisions of CSS2.1, and some are newly created, but all fall under the banner of CSS3.

The Specification should be your guide if you need to understand how a specific property or feature works behind the scene and how it works with other CSS properties. And if you are comfortable reading algorithms you won't get bored reading the CSS Specification.

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

**CSS can be added to HTML documents in 3 ways:**

1. Inline - by using the style attribute inside HTML elements.
2. Internal - by using a <style> element in the <head> section.
3. External - by using a <link> element to link to an external CSS file

13. What is embedded style sheets?

**Embedded Stylesheet:**It allows you to define styles for a particular HTML document as a whole in one place. This is done by embedding the **<style></style>** tags containing the CSS properties in the head of your document. Embedded style sheets are particularly useful for HTML documents that have unique style requirements from the rest of the documents in your project. However, if the styles need to be applied across multiple documents, you should link to an external style sheet instead of using individual embedded style sheets. Using embedded stylesheets holds a distinct advantage over inline styles which only allow you to address one HTML element at a time.

**Syntax:**The CSS syntax for embedded style sheets is exactly the same as other CSS code, apart from the fact that it is now wrapped within the <style></style> tags. The <style> tag takes the ‘type’ attribute that defines the type of style sheet being used (ie. text/CSS).

**Example 1:** Below is an HTML document with the CSS styling for the entire web page enclosed within the <style></style> tags. These properties would be applied to all corresponding elements in the HTML document.

14.What are the external style sheets?

# External style sheets

To apply a rule to multiple pages, an external style sheet is used. An external style sheet is a separate CSS file that can be accessed by creating a link within the head section of the webpage. Multiple webpages can use the same link to access the stylesheet.

The link to an external style sheet is placed within the head section of the page.

<head><link rel="stylesheet" type="text/css" href="mystyle.css"></head>

The actual style sheet file will contain CSS rules that are then applied across the entire page. For example:

body { background-color: ghostwhite;}h1 { color: blue; font-size: 20px; font-family: verdana; font-style:italic;}

In this case, the background color of the webpage will be ghostwhite and any h1 headings will appear in verdana font, as size twenty blue text in italic style.

External style sheets have the following advantages over internal and inline styles:

* one change to the style sheet will change all linked pages
* you can create classes of styles that can then be used on many different HTML elements
* consistent look and feel across multiple web pages
* improved load times because the css file is downloaded once and applied to each relevant page as needed.

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

**The advantages of External Style Sheets are as follows :**

* With the help of External Style Sheets, the styles of numerous documents can be organized from one single file.
* In External Style Sheets, Classes can be made for use on numerous HTML element types in many forms of the site.
* In complex contexts, Methods like selector and grouping can be implemented to apply styles.

**The disadvantages of External Style Sheets are as follows :**

* An extra download is essential to import style information for each file.
* The execution of the file may be deferred till the external style sheet is loaded.
* While implementing style sheets, we need to test Web pages with multiple browsers in order to check compatibility issues.

16. What is the meaning of the CSS selector?

CSS selectors are used to define the elements you want to style with CSS. There are many different types of CSS selectors, each with their own unique syntax. These tell the browser which elements to apply CSS property values to.

The element or elements targeted by a CSS selector are referred to as the “subject of the selector.” A subject can be selected based on its element type, class, ID name, given attribute, or pseudo-state.

With so many different types available, you are not only able to customize your site faster — you’re also able to maintain granular control over your code.

Below we’ll explain how you can use CSS selectors on your site and then explore the different types of selectors. Let’s get started.

## How to Use Selectors in CSS

There are two major ways you can use selectors in CSS. If you have your HTML and CSS in one doc, then you simply have to add CSS selectors into the <head> section of your webpage. You’ll see this method in the examples below.

However, you can also keep your HTML and CSS in separate documents. In that case, you might have an HTML document labelled index.html and a CSS file labelled style.css. The index.html file must include a line of code referencing the CSS file so that these styles are rendered on your webpage.

16. What are the media types allowed by CSS?

## Specifying media-dependent style sheets

There are currently two ways to specify media dependencies for style sheets:

* Specify the target medium from a style sheet with the @media or @import at-rules.
* @import url("fancyfonts.css") screen;
* @media print {
* /\* style sheet for print goes here \*/
* }
* Specify the target medium within the document language. For example, in HTML 4 ([[HTML4]](https://www.w3.org/TR/CSS21/refs.html#ref-HTML4)), the "media" attribute on the LINK element specifies the target media of an external style sheet:
* <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN">
* <HTML>
* <HEAD>
* <TITLE>Link to a target medium</TITLE>
* <LINK REL="stylesheet" TYPE="text/css"
* MEDIA="print, handheld" HREF="foo.css">
* </HEAD>
* <BODY>
* <P>The body...
* </BODY>
* </HTML>

The [@import](https://www.w3.org/TR/CSS21/cascade.html#at-import) rule is defined in the [chapter on the cascade](https://www.w3.org/TR/CSS21/cascade.html).

### The @media rule

An @media rule specifies the target [media types](https://www.w3.org/TR/CSS21/media.html%23media-types#media-types) (separated by commas) of a set of [statements](https://www.w3.org/TR/CSS21/syndata.html#tokenization) (delimited by curly braces). Invalid statements must be ignored per [4.1.7 "Rule sets, declaration blocks, and selectors"](https://www.w3.org/TR/CSS21/syndata.html#rule-sets) and [4.2 "Rules for handling parsing errors."](https://www.w3.org/TR/CSS21/syndata.html#parsing-errors) The @media construct allows style sheet rules for various media in the same style sheet:

@media print {

body { font-size: 10pt }

}

@media screen {

body { font-size: 13px }

}

@media screen, print {

body { line-height: 1.2 }

}

Style rules outside of @media rules apply to all media types that the style sheet applies to. At-rules inside @media are invalid in CSS2.1.

## Recognized media types

The names chosen for CSS media types reflect target devices for which the relevant properties make sense. In the following list of CSS media types the names of media types are normative, but the descriptions are informative. Likewise, the "Media" field in the description of each property is informative.

**all**

Suitable for all devices.

**braille**

Intended for braille tactile feedback devices.

**embossed**

Intended for paged braille printers.

**handheld**

Intended for handheld devices (typically small screen, limited bandwidth).

**print**

Intended for paged material and for documents viewed on screen in print preview mode. Please consult the section on [paged media](https://www.w3.org/TR/CSS21/page.html) for information about formatting issues that are specific to paged media.

**projection**

Intended for projected presentations, for example projectors. Please consult the section on [paged media](https://www.w3.org/TR/CSS21/page.html) for information about formatting issues that are specific to paged media.

**screen**

Intended primarily for color computer screens.

**speech**

Intended for speech synthesizers. Note: CSS2 had a similar media type called 'aural' for this purpose. See the appendix on [aural style sheets](https://www.w3.org/TR/CSS21/aural.html) for details.

**tty**

Intended for media using a fixed-pitch character grid (such as teletypes, terminals, or portable devices with limited display capabilities). Authors should not use [pixel units](https://www.w3.org/TR/CSS21/syndata.html#length-units) with the "tty" media type.

**tv**

Intended for television-type devices (low resolution, color, limited-scrollability screens, sound available).

Media type names are case-insensitive.

Media types are mutually exclusive in the sense that a user agent can only support one media type when rendering a document. However, user agents may use different media types on different canvases. For example, a document may (simultaneously) be shown in 'screen' mode on one canvas and 'print' mode on another canvas.

Note that a multimodal media type is still only one media type. The 'tv' media type, for example, is a multimodal media type that renders both visually and aurally to a single canvas.

@media and @import rules with unknown media types (that are nonetheless valid identifiers) are treated as if the unknown media types are not present. If an @media/@import rule contains a malformed media type (not an identifier) then the statement is invalid.

***Note:*** Media Queries supercedes this error handling.

For example, in the following snippet, the rule on the P element applies in 'screen' mode (even though the '3D' media type is not known).

@media screen, 3D {

P { color: green; }

}

***Note.*** Future updates of CSS may extend the list of media types. Authors should not rely on media type names that are not yet defined by a CSS specification.

### Media groups

This section is informative, not normative.

Each CSS property definition specifies which media types the property applies to. Since properties generally apply to several media types, the "Applies to media" section of each property definition lists media groups rather than individual media types. Each property applies to all media types in the media groups listed in its definition.

defines the following media groups:

* **continuous** or **paged**.
* **visual**, **audio**, **speech**, or **tactile**.
* **grid** (for character grid devices), or **bitmap**.
* **interactive** (for devices that allow user interaction), or **static** (for those that do not).
* **all** (includes all media types)

17. What is the rule set?

Each (valid) declaration block is preceded by one or more comma-separated selectors, which are conditions selecting some elements of the page. **A selector group and an associated declarations block, together, are called a ruleset, or often a rule**.

Rule sets **provide the capability to achieve this broader, more holistic, view of a data source and its records by executing and evaluating multiple rules together against individual records**. The output from rule sets provides a view into your data at several levels: Rule level.

18.Create Layouts

https://github.com/Amanmansuri2001/Task/commit/c849075d5130f92fdf5e36b5d58339f448c68ea4