

**Question:** What is a CSS selector? Provide examples of element, class, and ID selectors.

**Answer:**

CSS selectors are used to find the HTML elements you want to style.

Element Selector:

```
p {  
  text-align: center;  
  color: red;  
}
```

Class Selector:

```
.center {  
  text-align: center;  
  color: red;  
}
```

ID Selector:

```
#para1 {  
  text-align: center;  
  color: red;  
}
```

**Question:** Explain the concept of CSS specificity. How do conflicts between multiple styles get resolved?

**Answer:**

If there are two or more CSS rules that point to the same element, the selector with the highest specificity will "win", and its style declaration will be applied to that HTML element.

If two selectors have the same specificity, the one that is defined last in the CSS file will be applied. This is known as the "cascade" aspect of CSS.

**Question:** What is the difference between internal, external, and inline CSS? Discuss the advantages and disadvantages of each approach.

**Answer:**

- **Inline CSS:**

Applied directly within an HTML element's style attribute, affecting only that specific element.

Advantages:

Inline styles have the highest specificity, meaning they will override styles defined in internal or external CSS.

Useful for quick testing or debugging styles on specific elements without modifying external or internal styles.

Disadvantages:

Inline styles can make HTML cluttered and harder to read. They also lead to duplication if the same styles are applied to multiple elements.

Styles defined inline cannot be reused across multiple elements or pages, leading to redundancy.

Mixing styles directly in HTML goes against the principle of separating content (HTML) from presentation (CSS).

- **Internal CSS:**

Defined within a <style> tag in the <head> section of an HTML document, influencing the entire page.

Advantages:

Useful for single-page websites or when styles are only needed for one specific page.

All styles are contained within the same document, making it easy to see and edit styles related to that page.

Disadvantages:

Styles defined in internal CSS cannot be reused across multiple pages, leading to duplication if the same styles are needed elsewhere.

Each page with internal CSS will have its own styles, which can increase the overall size of the HTML document and slow down loading times.

- External CSS:

Contained in a separate .css file linked to the HTML document, allowing for consistent styling across multiple pages.

Advantages:

The same CSS file can be linked to multiple HTML documents, promoting consistency and reducing redundancy.

Keeps HTML and CSS separate, making both easier to read and maintain.

Browsers cache external CSS files, which can lead to faster load times for subsequent page visits.

Disadvantages:

Linking to an external CSS file requires an additional HTTP request, which can slow down the initial page load if the file is large or the server is slow.

If the external CSS file is not loaded (e.g., due to a broken link), the styles will not be applied, potentially leading to a poor user experience.

**Question:** Explain the CSS box model and its components (content, padding, border, margin). How does each affect the size of an element?

**Answer:**

The CSS Box Model defines how elements are sized, positioned, and rendered on a webpage.

- Content - The content of the box, where text and images appear
- Padding - Clears an area around the content. The padding is transparent
- Border - A border that goes around the padding and content
- Margin - Clears an area outside the border. The margin is transparent

**Question:** What is the difference between border-box and content-box box-sizing in CSS? Which is the default?

**Answer:**

border-box and content-box are the two different values of box sizing.

- content-box: This is the default value of box-sizing. The dimension of element only includes height and width and does not include border and padding given to element. Padding and Border take space outside the element.
- border-box: In this value, not only width and height properties are included but you will find padding and border inside of the box

content box is default value of box sizing.

**Question:** What is CSS Flexbox, and how is it useful for layout design? Explain the terms flex-container and flex-item.

**Answer:**

The Flexible Box Layout module introduces a one-dimensional layout system that handles space distribution and item alignment effectively. It works seamlessly for horizontal or vertical arrangements, making it a go-to solution for responsive designs.

- Flex Container: The parent div containing various divisions.
- Flex Items: The items inside the container div.

**Question:** Describe the properties justify-content, align-items, and flex-direction used in Flexbox.

**Answer:**

- justify-content: Aligns items along the main axis.
- align-items: Aligns multiple lines of items on the cross axis
- flex-direction: Defines the main axis direction

**Question:** Explain CSS Grid and how it differs from Flexbox. When would you use Grid over Flexbox?

**Answer:**

Flexbox which is primarily one-dimensional, CSS Grid allows for the arrangement of items in both rows and columns simultaneously. This makes it ideal for more complex layouts.

CSS Grid introduces the concept of grid lines and grid areas. You can explicitly place items in specific grid areas.

If you need to create a complex layout with both rows and columns, such as a magazine-style layout, a dashboard, or a grid of cards, CSS Grid is the better choice.

If your design requires items to overlap or span multiple rows and columns, CSS Grid allows for this kind of layout easily.

**Question:** Describe the grid-template-columns, grid-template-rows, and grid-gap properties. Provide examples of how to use them.

**Answer:**

Grid-template-column: Specifies the size of the columns, and how many columns in a grid layout.

```
.container {  
    display: grid;  
    grid-template-columns: 100px 200px auto;  
}
```

Grid-template-row: Specifies the size of the rows in a grid layout.

```
.container {  
    display: grid;  
    grid-template-rows: 150px 100px;  
}
```

Grid-gap: A shorthand property for the *row-gap* and the *column-gap* properties.

```
.container {  
    display: grid;  
    gap: 20px 50px;  
}
```

**Question:** What are media queries in CSS, and why are they important for responsive design?

**Answer:**

media queries are a technique that allows developers to apply different styles to a website based on specific device conditions like screen size, orientation, resolution

Media queries allow web pages to adapt to different screen sizes and resolutions. This is crucial in today's multi-device environment, where users access websites on smartphones, tablets, laptops, and desktops.

By using media queries, developers can create layouts that are optimized for various devices, ensuring that content is easily readable and navigable. This leads to a better user experience, as users do not have to zoom in or scroll horizontally to view content.

**Question:** Write a basic media query that adjusts the font size of a webpage for screens smaller than 600px.

**Answer:**

```
body
{
    font-size: 25px;
}

@media (max-width: 600px)
{
    body
    {
        font-size: 14px;
    }
}
```

**Question:** Explain the difference between web-safe fonts and custom web fonts. Why might you use a web-safe font over a custom font?

**Answer:**

Web-safe fonts are a set of fonts that are widely available and pre-installed on most operating systems and devices. These fonts are considered safe because they are likely to be present on the user's device, ensuring consistent rendering across different platforms and browsers.

Custom web fonts are fonts that are not typically installed on users' devices. They are often loaded from a web font service and can include a wide variety of styles and designs.

Web-safe fonts do not require additional HTTP requests to load, which can improve page load times, especially on mobile devices or slower connections.

**Question:** What is the font-family property in CSS? How do you apply a custom Google Font to a webpage?

**Answer:**

The font-family property specifies the font for an element. The font-family property can hold several font names as a "fallback" system. If the browser does not support the first font, it tries the next font.

```
<style>
```

```
@importurl('https://fonts.googleapis.com/css2?family=Roboto&family=Pacifico&family=Lobster&
display=swap');
```

```
</style>
```