**CSS Interview Questions**

1. **What Is CSS?**

**Ans:** Cascading Style Sheet is a web designing language simple for HTML elements.  It is basically used to simplify the process and make the web page look beautiful.

### What are the advantages of CSS?

Ans: There are a number of advantages of CSS,

* It gives lots of flexibility for setting the properties of the element
* Easy maintenance
* It allows separation of content of the HTML document from the style and layout of the content basically
* Loading of pages at a faster pace
* Compatibility with multiple device
* Increases the website’s adaptability and makes it compatible to future browsers.

1. **How Many ways can a CSS integrated as a web page?**

**Ans:**

**Inline**: term is used when the CSS code have attribute of HTML elements

<p style="colour:skyblue;"> hello world!</p>

**External:** Separate CSS file is created in the workspace and later linking them in every web page that is created.

<head>

<link rel="stylesheet" type="text/css" href="your\_CSS\_file\_location"/>

</head>

**Internal (Embeded)**: the head element of the web page has internal CSS implemented in it1

<head>

<style>

p {

color: lime;

background-color: black;

}

</style>

</head>

## What is The Viewport?

The viewport is the user's visible area of a web page.

The viewport varies with the device, and will be smaller on a mobile phone than on a computer screen.

HTML5 introduced a method to let web designers take control over the viewport, through the <meta> tag.

A <meta> viewport element gives the browser instructions on how to control the page's dimensions and scaling.

**<meta name="viewport" content="width=device-width, initial-scale=1.0">**

### Discuss the merits and demerits of Embedded Style Sheets?

### Ans:

There are several merits and demerits of embedded style sheets:

**Merits of Embedded Style Sheets:**

Multiple tag types can be created in a single document.

Styles, in complex situations, can be applied by using Selector and Grouping methods.

Extra download is unnecessary.

**Demerits of Embedded Style Sheets:**

Multiple documents cannot be controlled.

### 5. What is CSS Box Model and what are its elements?

Ans: In CSS, the term "box model" is used when talking about design and layout.

The CSS box model is essentially a box that wraps around every HTML element.

The CSS box defines the design and the layout of elements of CSS. The several elements are:

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

div {

width: 300px;

border: 25px solid yellow;

padding: 25px;

margin: 25px;

}

### What is the purpose of the z-index and how is it used?

### Ans: The z-index property in CSS controls the vertical stacking order of elements that overlap.

### When elements are positioned, they can overlap other elements.

### An element with a higher z-index is always stacked above than a lower index.

Z-Index can take the following values:

**Auto:** Sets the stack order equal to its parents.

**Number:** Orders the stack order.

**Initial:** Sets this property to its default value (0).

**Inherit:** Inherits this property from its parent element.

**8. What is display: block?**

Display: block;

Span {

display: block;

}

</style>

</head>

<body>

<span>A display property with a value of "block" results in</span> <span>a line break between the two elements.</span>

</body>

</html>

A block-level element always starts on a new line and takes up the full width available.

A display property with a value of "block" results in a line break between the two elements**.**

**9. What is display : inline?**

display: inline; An inline element does not start on a new line and only takes up as much width as necessary.

**10. What is display: none?**

display: none; is commonly used with JavaScript to hide and show elements without deleting and recreating them.

**Position Property:**

Elements are positioned using top, right, bottom and left properties.

However these properties will not work unless we declare first Position property then it will work properly.

**11. Position in CSS?**

**Position: static**

Static positioned elements are not affected by the top, bottom, left, and right properties.

An element with position: static; is not positioned in any special way; it is always positioned according to the normal flow of the page:

## position: relative;

An element with position: relative; is positioned relative to its normal position.

Setting the top, right, bottom, and left properties

## position: absolute;

an **absolute**positioned element is **relative to its parent**.

An element with position: absolute; is positioned relative to the nearest positioned ancestor (instead of positioned relative to the viewport, like fixed).

## position: fixed;

An element with position: fixed; is positioned relative to the viewport, which means it always stays in the same place even if the page is scrolled:

**12. Width Vs Max-Width?**

**Width:** The problem with the <div> above occurs when the browser window is smaller than the width of the element. The browser then adds a horizontal scrollbar to the page.

**Max-width: U**sing max-width instead, in this situation, will improve the browser's handling of small windows. This is important when making a site usable on small devices:

## The display: inline-block Value

Compared to display: inline, the major difference is that display: inline-block allows to set a width and height on the element.

Also, with display: inline-block, the top and bottom margins/paddings are respected, but with display: inline they are not.

## What are Pseudo-Elements?

**A CSS pseudo-element is used to style specified parts of an element.**

For example, it can be used to:

* Style the first letter, or line, of an element
* Insert content before, or after, the content of an element

**p::first-line {**

**color: #ff0000;**

**font-variant: small-caps;**

**}**

**Example:**

**1.First-letter**

**2.first-line**

**3.p: before**

**4.p.after**

**5 ::selection**

**What are Pseudo-classes?**

A pseudo-class is used to define a special state of an element.

For example, it can be used to:

* Style an element when a user mouses over it
* Style visited and unvisited links differently
* Style an element when it gets focus

a.highlight:hover {

color: #ff0000;

}

**Example:**

**1. :hover**

**2. :link**

**3. :first-child**

**4. : focus**

**5 : nth-child()**

What is CSS Grid?

The CSS Grid Layout Module offers a grid-based layout system, with rows and columns, making it easier to design web pages without having to use floats and positioning.

## CSS Box Sizing

The CSS box-sizing property allows us to include the padding and border in an element's total width and height.

**box-sizing: border-box**  on an element padding and border are included in the width and height

**Content-box:** Default. The width and height properties (and min/max properties) includes only the content. Border and padding are not included.

**What is FlexBox?**

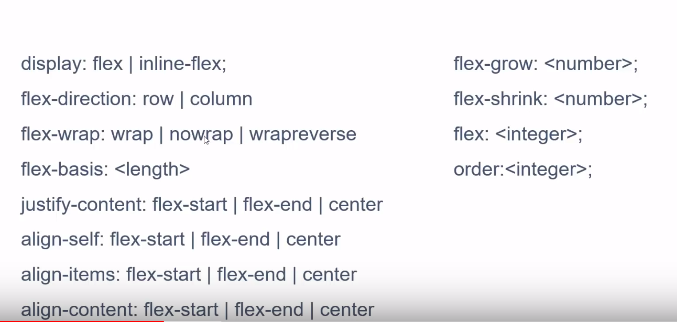
**The Flexible Box Layout Module, makes it easier to design flexible responsive layout structure without using float or positioning.**

**Examples:** align-content: flex-start;

align-content: center;

align-content: space-between;

align-content: space-around;



**Media queries**

Using media queries are a popular technique for delivering a tailored style sheet to desktops, laptops, tablets, and mobile phones (such as iPhone and Android phones).

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

## CSS3 Media Types

|  |  |
| --- | --- |
| **Value** | **Description** |
| **All** | Used for all media type devices |
| **print** | Used for printers |
| **screen** | Used for computer screens, tablets, smart-phones etc. |
| **speech** | Used for screen readers that "reads" the page out loud |

**Example:**

@media only screen and (max-width: 600px) {  
  body {  
    background-color: lightblue;  
  }  
}

# ****rem****

rem values are relative to the root html element, not to the parent element. That is, If font-size of the root element is 16px then 1 rem = 16px for all elements.

**What is Cross Browser Compatibility?**

Browser Compatibility is the manner in which a web page looks in different web browsers. Different browsers read the website code differently. In other words, Chrome will render a website Differently than FireFox or Internet Explorer will.

**TO over Come this issue following ways**

**1. DocType error.**

**2. Html/CSS validators**

**3. add a vendor to the specific functions**

**4. Layout Compatibility**

**5. Website feature compatibility**

**Css validators:**

Prefixes for different browsers are:

* -ms for Microsoft (Internet Explorer)
* -moz for Mozilla Foundation (Firefox)
* -o for Opera Software
* -webkit for Safari and Chrome

A full declaration of Opacity may look like:

.test{

-moz-opacity: 0.6;

-ms-opacity: 0.6;

-webkit-opacity: 0.6;

opacity: 0.6;

}