# Different types of CSS.

There are three types of CSS as mentioned below.

* **External –**These are written in separate files.
* **Internal –**These are cited at the top of the web page code document.
* **Inline –** These are written right next to the text.

# **Explain the term Responsive web design.**

It is a method in which we design and develop a web page according to the user activities and conditions which are based on various components like the size of the screen, portability of the web page on the different devices, etc.

Hence it is done by using different flexible layouts and grids.

# How do we make a rounded corner by using CSS?

We can make a rounded corner by using the property “border-radius”. We can apply this property to any element.

# **What is the use of the Box Model in CSS?**

In CSS, the box model is a box that binds all the HTML elements and it includes features like margins, border, padding, and the actual content.

By using a box model we will get the authority to add the borders all around the elements and we can also define the space between the elements.

# **What is a CSS pseudo-class?**

It is a class that is used to define a **special state of an HTML element**.

This class can be used by styling an element when a user snooped over it and also it can style an HTML element when it gets the focus.

selector:pseudo-class {

property:value;

}

# Pseudo Element

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

# **What are the differences between relative and absolute in CSS?**

The main difference between relative and absolute is that “relative” is used for the same tag in CSS and it means that if we write the left:10px then the padding will shift to 10px in the left while absolute is totally relative to the non-static parent.

It means if we write left:10px then the result will be 10px far from the left edge of the parent element.

# The position Property

The position property specifies the type of positioning method used for an element.

There are five different position values:

* static
* relative
* fixed
* absolute
* sticky

Elements are then positioned using the top, bottom, left, and right properties. However, these properties will not work unless the position property is set first. They also work differently depending on the position value.

## position: static;

HTML elements are positioned static by default.

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

## position: relative;

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

## 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. The top, right, bottom, and left properties are used to position the element.

## position: absolute;

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

## position: sticky;

An element with position: sticky; is positioned based on the user's scroll position.

# **Differentiate between inline and block element.**

A block-level element always starts on a new line and takes up the full width available (stretches out to the left and right as far as it can).

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

# **Differentiate between display:none and visibility:hidden**

display:none means that the tag in question will not appear on the page at all (although you can still interact with it through the dom). There will be no space allocated for it between the other tags.

visibility:hidden means that unlike display:none, the tag is not visible, but space is allocated for it on the page. The tag is rendered, it just isn't seen on the page.

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

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

**Margin**: the top most layer, the overall structure is shown  
**Border**: the padding and content option with a border around it is shown.  Background color affects the border.  
**Padding**: Space is shown. Background colour affects the border.  
**Content**: Actual content is shown.

# **Define float property of CSS?**

By float property, the image can be moved to the right or the left along with the text to be wrapped around it. Elements before this property is applied do not change their properties.

# **How does Z index function?**

Overlapping may occur while using CSS for positioning HTML elements. Z index helps in specifying the overlapping element. It is a number which can be positive or negative, the default value being zero.

# What are functions/mixins?

Mixins are a very handy way of adding a number of styles, based on a particular input parameter. For example, you might always want to add fallback styles when adding border-radius, but you don’t necessarily know what value you might want.

@mixin border-radius($radius) {

-webkit-border-radius: $radius;

-moz-border-radius: $radius;

-ms-border-radius: $radius;

border-radius: $radius;

}

.box {

@include border-radius(10px);

}