CSS

**What is the Box model in CSS? Which CSS properties are a part of it?**

ANS :

* **Content:**  Actual Content of the box where the text or image placed.
* **Padding:** Area surrounding the content (Space between the border and content).
* **Border:** Area surrounding the padding.
* **Margin:** Area surrounding the border.

### What are the different types of Selectors in CSS?

**ANS :**

**Universal Selector:** The universal selector works like a wildcard character, selecting all elements on a page. In the given example, the provided styles will get applied to all the elements on the page.

\* {

**color**: "green";

**font-size**: 20px;

**line-height**: 25px;

}

**Element Type Selector:** This selector matches one or more HTML elements of the same name. In the given example, the provided styles will get applied to all the ul elements on the page.

**ul** {

line-style: none;

**border**: solid 1px #ccc;

}

**ID Selector:** This selector matches any HTML element that has an ID attribute with the same value as that of the selector. In the given example, the provided styles will get applied to all the elements having ID as a container on the page.

#container {

**width**: 960px;

**margin**: 0 auto;

}

<**div** id="container"></**div**>

**Class Selector:** The class selector also matches all elements on the page that have their class attribute set to the same value as the class.  In the given example, the provided styles will get applied to all the elements having ID as the box on the page.

.box {

**padding**: 10px;

**margin**: 10px;

**width**: 240px;

}

<**div** class="box"></**div**>

**Descendant Combinator:** The descendant selector or, more accurately, the descendant combinator lets you combine two or more selectors so you can be more specific in your selection method.

#container .box {

**float**: left;

**padding-bottom**: 15px;

}

<**div** id="container">

<**div** class="box"></**div**>

<**div** class="box-2"></**div**>

</**div**>

<**div** class=”box”></**div**>

This declaration block will apply to all elements that have a class of box that is inside an element with an ID of the container. It’s worth noting that the .box element doesn’t have to be an immediate child: there could be another element wrapping .box, and the styles would still apply.

### What is the difference between inline, inline-block, and block?

**Block Element:** The block elements always start on a new line. They will also take space for an entire row or width. List of block elements are <div>, <p>.

**Inline Elements:** Inline elements don't start on a new line, they appear on the same line as the content and tags beside them. Some examples of inline elements are <a>, <span> , <strong>, and <img> tags.

**Inline Block Elements:** Inline-block elements are similar to inline elements, except they can have padding and margins added on all four sides.

**What is a Pseudo element? What is pseudo-class?**

Pseudo-classes select regular elements but under certain conditions like when the user is hovering over the link.

* :link
* :visited
* :hover
* :active
* :focus

A pseudo-element however allows us to create items that do not normally exist in the document tree, for example ::after.

* ::before
* ::after
* ::first-letter
* ::first-line
* ::selection

### Explain CSS position property?

**Absolute**

To place an element exactly where you want to place it. absolute position is actually set relative to the element's parent. if no parent is available then the relative place to the page itself (it will default all the way back up to the element).

**Relative**

"Relative to itself". Setting position: relative; on an element and no other positioning attributes, it will no effect on its positioning. It allows the use of z-index on the element and it limits the scope of absolutely positioned child elements. Any child element will be absolutely positioned within that block.

**Fixed**

The element is positioned relative to the viewport or the browser window itself. viewport doesn't change if you scroll and hence the fixed element will stay right in the same position.

**Static**

Static default for every single page element. The only reason you would ever set an element to position: static is to forcefully-remove some positioning that got applied to an element outside of your control.

**Sticky**

Sticky positioning is a hybrid of relative and fixed positioning. The element is treated as relative positioned until it crosses a specified threshold, at which point it is treated as fixed positioned.

### What are the different ways to hide the element using CSS?

Using display property(display: none). It’s not available for screen readers. The element will not exist in the DOM if display: none is used.

Using visibility property(visibility: hidden), will take up the space of the element. It will be available to screen reader users. The element will actually be present in the DOM, but not shown on the screen.