Q.1 What's Box Model in CSS ?

The **CSS box model** is a container that contains multiple properties including borders, margins, padding, and the content itself. It is used to create the design and layout of web pages.The web browser renders every element as a rectangular box according to the CSS box model.

Eg.

- **content**: This contains the actual data in the form of text, images, or other media forms and it can be sized using the width & height property.
- **padding**: This property is used to create space around the element, inside any defined border.
- **border**: This property is used to cover the content & any padding, & also allows setting the style, color, and width of the border.
- **margin**: This property is used to create space around the element ie. around the border area.

Q.2 What are the Different Types of Selectors in CSS & what are the advantages of them?

CSS selectors are used *to select the content you want to style*.CSS selectors select HTML elements according to its id, class, type, attribute etc.

There are several different types of selectors in CSS.

- **Simple Selector:** It is used to select the HTML elements based on their element name, id, attributes, etc.
    - Element Selector : The element selector selects HTML elements based on the element name (or tag) for example p, h1, div, span, etc.
    - Id Selector : The **id (#)** selector uses the *id* attribute of an HTML element to select a specific element.
    - Class Selector : The class selector (.) selects HTML elements with a specific class attribute.
- **Group Selector : This selector is used to style all comma-separated elements with the same style.**
- **Universal Selector: The Universal selector (*) in CSS is used to select all the elements in an HTML document. It also includes other elements which are inside under another element.**
- **Attribute Selector: The attribute selector [attribute] is used to select the elements with a specified attribute or attribute value.**
- **Pseudo-Element Selector: It is used to style any specific part of the element. For Example- It is used to style the first letter or the first line of any element**
- **Pseudo-Class Selector: It is used to style a special type of state of any element. For example- It is used to style an element when a mouse cursor hovers over it.**

Q.3 What is VW/VH ?

The full form of VW is **viewport width**. It works like the **percentage unit.** Specifying **10vw** is equivalent to occupying 10% of entire visible screen width.

The full form of VH is **viewport height**. It works like the **percentage unit** as well. Specifying **10vh** is equivalent to occupying 10% of entire visible screen height.

Q.4 Whats difference between Inline, Inline Block and block ?

Block : Block elements begin from a new line by default and cover space to its left and right as far as it can go. The height that it covers is equal to the content height. Also, it covers the whole horizontal space of its parent element.

Inline : Inline elements never start from a new line and only cover the width according to the size of bounded tags in the HTML element.

Inline Block : The display value of inline-block works similarly to inline with one exception: the height and width of that element become modifiable.

Q.5 How is Border-box different from Content Box?

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 for example .box {width: 200px; border: 10px solid black;} renders a box that is 200px wide.

Q.6 What's z-index and How does it Function ?

Z Index (z-index) is a CSS property that defines the order of overlapping HTML elements. Elements with a higher index will be placed on top of elements with a lower index.

z-index only works on positioned elements (position: absolute, position: relative, position: fixed, or position: sticky) and flex items (elements that are direct children of display:flex elements).

Q.7 What's Grid & Flex and difference between them?

Grid : CSS Grid Layout, is a two-dimensional grid-based layout system with rows and columns, making it easier to design web pages without having to use floats and positioning. Like tables, grid layout allow us to align elements into columns and rows.

To get started you have to define a container element as a grid with display: grid, set the column and row sizes with grid-template-columns and grid-template-rows, and then place its child elements into the grid with grid-column and grid-row.

Flexbox : The CSS Flexbox offers a one-dimensional layout. It is helpful in allocating and aligning the space among items in a container (made of grids). It works with all kinds of display devices and screen sizes.

To get started you have to define a container element as a grid with display: flex;

Q.8 Difference between absolute and relative and sticky and position explain with example.

Absolute:

absolute position related to its closet positioned ancestor

absolute position does not maintain gap

we can do overlapping using absolute position

we can use top,left,right,bottom, z-index porpertise

Relative :

Relative position are used on current window

we can use top,left,right,bottom, z-index properties

in relative position it leave gap on its original position and any other content will not fit in that place

Sticky :

Sticky position show behaviour to its related parents container only

we can use top,left,right,bottom, z-index properties

Fixed :

fixed postionare used show content on the entire web page

we can use top,left,right,bottom, z-index porpertise

fixed element does not leave a gap in the page where it would normally have been located

Q.10 Build Responsive Layout both desktop and mobile and Tablet, see below image for reference ?