CSS POSITION

- The position property specifies the type of positioning method used for an element
- Basically 5 types are there
  - Static
  - Relative
  - Fixed
  - Absolute
  - Sticky

CSS STATIC POSITION

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

CSS RELATIVE POSITION

- An element with position: relative; is positioned relative to its normal position.
- Setting the top, right, bottom, and left properties of a relatively-positioned element will cause it to be adjusted away from its normal position. Other content will not be adjusted to fit into any gap left by the element.

CSS FIXED POSITION

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

CSS ABSOLUTE POSITION

- An element with position: absolute; is positioned relative to the nearest positioned ancestor (instead of positioned relative to the viewport, like fixed).
- However; if an absolute positioned element has no positioned ancestors, it uses the document body, and moves along with page scrolling.

CSS STICKY POSITION

- An element with position: sticky; is positioned based on the user's scroll position.
- A sticky element toggles between relative and fixed, depending on the scroll position. It is positioned relative until a given offset position is met in the viewport - then it "sticks" in place (like position:fixed).

CSS DISPLAY

- The display property is the most important CSS property for controlling layout.
- The display property specifies if/how an element is displayed.
- Every HTML element has a default display value depending on what type of element it is. The default display value for most elements is block or inline.
- Some block level element are :- div, h1, p
- Some inline elements are :- span, a, img

CSS FLOAT & CLEAR

- The float property is used for positioning and formatting content.
- The float property can have one of the following values:
  - left - The element floats to the left of its container
  - right - The element floats to the right of its container
  - none - The element does not float (will be displayed just where it occurs in the text). This is default
  - inherit - The element inherits the float value of its parent

- The clear property specifies what should happen with the element that is next to a floating element.
- The clear property can have one of the following values:
  - none - The element is not pushed below left or right floated elements. This is default
  - left - The element is pushed below left floated elements
  - right - The element is pushed below right floated elements
  - both - The element is pushed below both left and right floated elements
  - inherit - The element inherits the clear value from its parent

CSS BOX SIZING

- The CSS box-sizing property allows us to include the padding and border in an element's total width and height.
- This property is used when you want to clearly define perfect width & height for your elements.

CSS MEDIA QUERY

- Media queries in CSS3 extended the CSS2 media types idea: Instead of looking for a type of device, they look at the capability of the device.
- 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
- Using media queries are a popular technique for delivering a tailored style sheet to desktops, laptops, tablets, and mobile phones

CSS MEDIA QUERY SYNTAX

- The result of the query is true if the specified media type matches the type of device the document is being displayed on and all expressions in the media query are true. When a media query is true, the corresponding style sheet or style rules are applied, following the normal cascading rules.
- Unless you use the not or only operators, the media type is optional and the all type will be implied.
  - @media not|only mediatype and (expressions) { CSS-Code;}
- You can also have different stylesheets for different media:
  - <link rel="stylesheet" media="mediatype and|not|only (expressions)" href="print.css">