

MODULE 02

CSS Basics

CSS

- CSS stands for Cascading Style Sheets
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media
- CSS saves a lot of work. It can control the layout of multiple web pages all at once
- External stylesheets are stored in CSS files

HOW TO ADD CSS?

- There are three ways of inserting a style sheet:
 - External CSS
 - With this type, you can change the look of an entire website by changing just one file.
 - Each HTML page must include a reference to the external style sheet file inside the `<link>` element, inside the head section. (`<link rel="stylesheet" href="firstPageStyle.css">`)
 - Internal CSS
 - An internal style sheet may be used if one single HTML page has a unique style.
 - The internal style is defined inside the `<style>` element, inside the head section.
 - Inline CSS
 - An inline style may be used to apply a unique style for a single element.
 - To use inline styles, add the `style` attribute to the relevant element. The `style` attribute can contain any CSS property.

CSS COLOR

- There are 6 different ways for applying color in CSS
 - Using 140 Standard Color Name (red, lime, cyan)
 - RGB (rgb(52, 199, 71))
 - HSL (hsl(52, 30%, 60%))
 - HEX (#FFFF00)
 - RGBA (rgb(25, 79, 91, 0.8))
 - HSLA (hsl(55, 9%, 14%, 0.4))

CSS COLOR

- CSS Background Color
 - `background-color:#00bb99;`
- CSS Text Color
 - `color:blue;`
- CSS Border Color
 - `border-color: rgb(214,147,21);`

CSS BACKGROUND

- The following properties are used to add background effects for elements.
 - `background-color` (specifies the background color of an element)
 - `background-image` (specifies an image to use as the background of an element)
 - `background-repeat` (helps to control the repetition of image vertically & horizontally)
 - `background-attachment` (specifies whether the background image should scroll or be fixed)
 - `background-position` (specify the position of the background image)
 - `Background-size` (specifies the size of the background images)
 - `background` (shorthand property)

CSS BORDERS

- The following properties are used to add border effects for elements.
 - `border-width`
 - `border-style`
 - `border-color`
 - `border-radius`
 - `border` (shorthand property)

CSS BORDER SIDE

- In CSS, there are also properties for specifying each of the borders (top, right, bottom, and left):
 - If the border-style property has four values:
 - border-style: dotted solid double dashed;
 - top border is dotted, right border is solid, bottom border is double, left border is dashed
 - If the border-style property has three values:
 - border-style: dotted solid double;
 - top border is dotted, right and left borders are solid, bottom border is double
 - If the border-style property has two values:
 - border-style: dotted solid;
 - top and bottom borders are dotted, right and left borders are solid
 - If the border-style property has one value:
 - border-style: dotted;
 - all four borders are dotted
- This is applicable to border-width & color also.

CSS TEXT

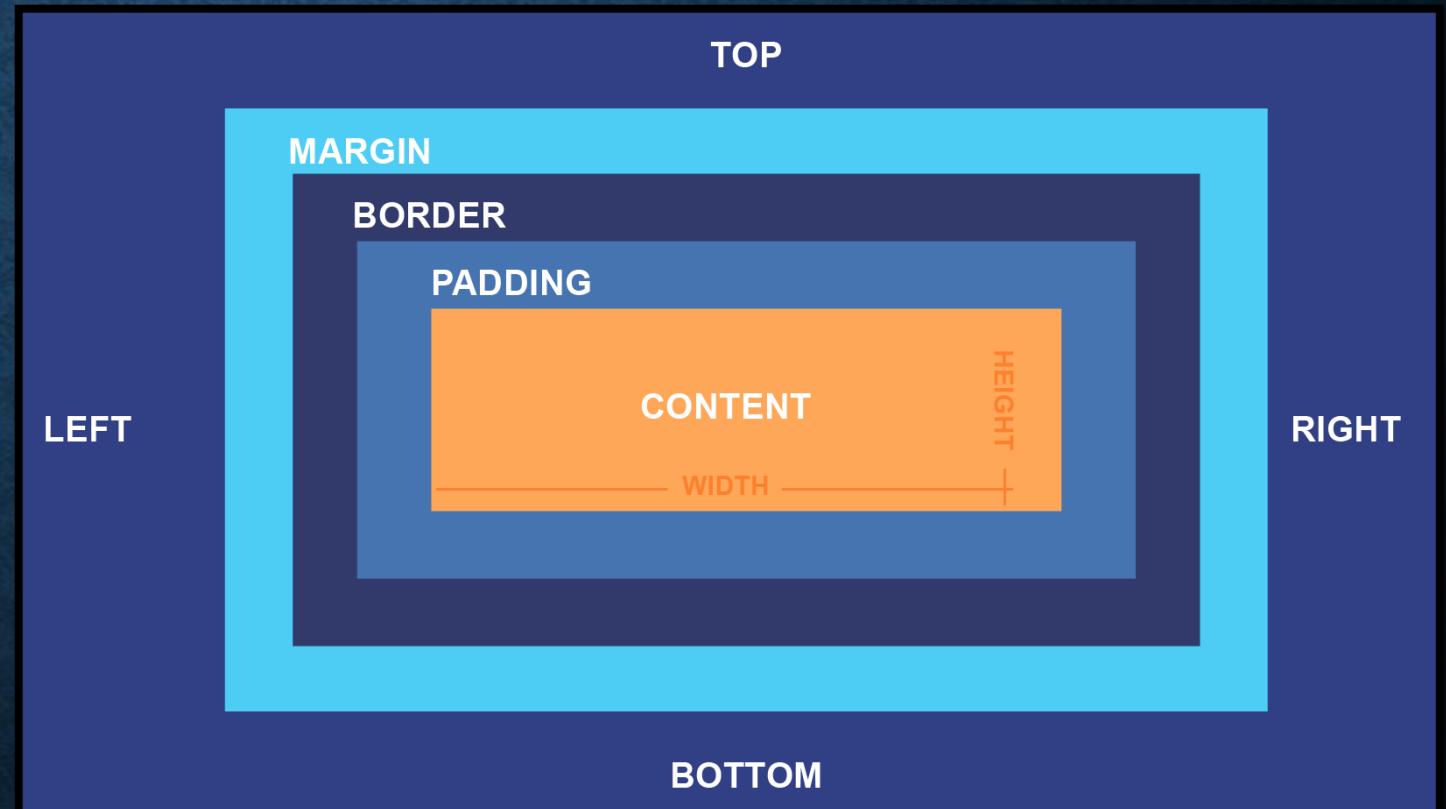
- **Text Color**
 - `background-color`, `color`
- **Text Alignment**
 - `text-align`, `text-align-last`, `direction`, `unicode-bidi`, `vertical-align`
- **Text Decoration**
 - `text-decoration-line`, `text-decoration-color`, `text-decoration-style`, `text-decoration-thickness`, `text-decoration`
- **Text Transformation**
 - `text-transform`
- **Text Spacing**
 - `text-indent`, `letter-spacing`, `line-height`, `word-spacing`, `white-space`
- **Text Shadow**
 - `text-shadow`

CSS FONT

- Font Family
 - `font-family`
- Font Style
 - `font-style`
- Font Weight
 - `font-weight`
- Font Size
 - `font-size`
- Font-variant
 - `Font-varient`

CSS BOX MODEL

- The CSS box model is essentially a box that wraps around every HTML element.
- Basically 4 element are there
 - Content
 - Padding
 - Border
 - Margin
- How width & height are calculated?



CSS COMBINATORS SELECTORS

- A CSS selector can contain more than one simple selector. Between the simple selectors, we can include a combinator.
- There are four different combinators in CSS:
 - descendant selector (space) (Selects all 2nd elements inside 1st element)
 - child selector (>) (Selects all 2nd elements where the parent is 1st element)
 - adjacent sibling selector (+) (Selects the 2nd element that is placed immediately after 1st element)
 - general sibling selector (~) (selects all 2nd elements that are next siblings of 1st element.)

CSS PSEUDO CLASS SELECTORS

- There are different pseudo classes in CSS, some are:
 - :hover (Selects elements on mouse over)
 - :first-child (Selects every element that is the first child of its parent)
 - :nth-child(n) (Selects every element that is the nth child of its parent)
 - :visited (Selects all visited links)
 - :last-child (Selects every element that is the last child of its parent)

CSS PSEUDO ELEMENT SELECTORS

- There are different pseudo classes in CSS, some are:
 - ::after (Insert something after the content of each given element)
 - ::before (Insert something before the content of each given element)
 - ::first-letter (Selects the first letter of each given element)
 - ::first-line (Selects the first line of each given element)
 - ::marker (Selects the markers of list items)
 - ::selection (Selects the portion of an element that is selected by a user)

CSS ATTRIBUTE SELECTORS

- There are different pseudo classes in CSS, some are:
 1. [attribute] (Select all element with the given attribute)
 2. [attribute=value] (Select all element having attribute with given value)
 3. [attribute~=value] (Select all element having attribute containing a word with given value)
 4. [attribute|=value] (Select element having attribute having given value or have value-something)
 5. [attribute^=value] (Selects every element whose attribute value begins with given value)
 6. [attribute\$=value] (Selects every element whose attribute value ends with given value)
 7. [attribute*=value] (Selects every element whose attribute value contains the given value)

Note :- Selectors 3 & 4 works with whole word not with letters.

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">`