1. CSS BASICS

- What is CSS :- CSS (Cascading Style Sheets) is used to style and design web pages.

- CSS Syntax :- CSS syntax consists of a selector, a property, and a value.

- How CSS works :- CSS works by applying style rules to HTML elements in the browser.

- CSS Comments :- CSS comments are used to explain code and are written inside /* */.

- CSS Version History :- CSS evolved from CSS1 to CSS2, CSS2.1, and CSS3.

2. TYPES OF CSS

- Inline CSS :- Inline CSS is written directly inside an HTML element using the style attribute.

- Internal CSS :- Internal CSS is written inside the <style> tag in the <head> section of an HTML file.

- External CSS :- External CSS is written in a separate .css file and linked to HTML.

- CSS Priority & Cascade :- CSS priority follows the order: Inline > Internal > External, and the last rule overrides others.

3. CSS SELECTORS

- Universal Selector (*) :- Selects all elements on the webpage.

- Element Selector :- Selects HTML elements by their tag name.

- Class Selector (.) :- Selects elements that have a specific class attribute.

- ID Selector (#) :- Selects a unique element with a specific ID.

- Group Selector :- Selects multiple elements at the same time using commas.

- Descendant Selector :- Selects elements that are inside another specified element.

- Child Selector (>) :- Selects elements that are direct children of a specified element.

- Adjacent Sibling Selector (+) :- Selects the element immediately after a specified element.

- General Sibling Selector (~) :- Selects all sibling elements after a specified element.

- Attribute Selectors :- Select elements based on their attributes or attribute values.

- Pseudo-classes (:hover, :focus, etc.) :- Apply styles based on element states like hover or focus.

- Pseudo-elements (::before, ::after) :- Style specific parts of an element like before or after content.

4. CSS TYPOGRAPHY

- font-family :- Specifies the font style for text.

- font-size :- Defines the size of the text.

- font-style :- Sets text style like normal, italic, or oblique.

- font-weight :- Controls the thickness or boldness of text.

- line-height :- Sets the space between lines of text.

- letter-spacing :- Controls the space between letters.

- word-spacing :- Controls the space between words.

- text-align :- Aligns text left, right, center, or justify.

- text-decoration :- Adds decoration like underline or line-through.

- text-transform :- Changes text to uppercase, lowercase, or capitalize.

- white-space :- Controls how whitespace and line breaks are handled.

- web safe fonts :- Fonts that are supported across all browsers and devices.

- Google Fonts :- Free web fonts provided by Google that can be used online.

5. CSS COLORS & BACKGROUNDS

- Color formats (name, hex, rgb, rgba, hsl) :- Different ways to define colors in CSS.

- background-color :- Sets the background color of an element.

- background-image :- Adds an image as the background.

- background-repeat :- Controls whether the background image repeats.

- background-position :- Sets the position of the background image.

- background-size :- Controls the size of the background image.

- background-attachment :- Defines whether the background scrolls with the page or not.

- gradients (linear, radial) :- Creates smooth color transitions without images.

6. CSS BOX MODEL

- content :- The actual text or image inside an element.

- padding :- Space between the content and the border.

- Border :- A line that surrounds the padding and content.

- Margin :- Space outside the border between elements.

- box-sizing :- Defines how width and height of elements are calculated.

- Outline :- A line drawn outside the border without affecting layout.

7. CSS UNITS

- Absolute Units (px, cm, mm, in) **:-** Fixed units that do not change based on screen size.

- Relative Units (%, em, rem) **:-** Units that depend on parent or root element size.

- Viewport Units (vw, vh, vmin, vmax) **:-** Units based on the size of the browser window.

8. CSS DISPLAY & VISIBILITY

- display: block **:-** Element takes full width and starts on a new line.

- display: inline **:-** Element takes only required width and stays in line.

- display: inline-block **:-** Inline element that allows width and height.

- display: none **:-** Completely removes the element from the layout.

- visibility: hidden **:-** Hides the element but keeps its space.

- Opacity **:-** Controls the transparency level of an element.

9. CSS POSITIONING

- Static **:-** Default position, elements follow normal document flow.

- Relative **:-** Positioned relative to its normal position.

- Absolute **:-** Positioned relative to the nearest positioned ancestor.

- Fixed **:-** Positioned relative to the browser window.

- Sticky **:-** Switches between relative and fixed based on scroll.

- z-index **:-** Controls the vertical stacking order of elements.

10. CSS FLOAT & CLEAR

- float **:-** Moves an element to the left or right of its container.

- Clear **:-** Prevents elements from floating next to floated elements.

- Clearfix **:-** Technique to fix parent height issues caused by float.

- layout issues with float **:-** Float can break layouts if not cleared properly.

11. CSS OVERFLOW

- overflow **:-** Controls content overflow inside an element.

- overflow-x **:-** Controls horizontal overflow.

- overflow-y **:-** Controls vertical overflow.

- Hidden **:-** Hides overflowing content.

- Scroll **:-** Always shows scrollbars.
- Auto **:-** Shows scrollbars only when needed.

---

12. CSS WIDTH & HEIGHT

- Width **:-** Sets the width of an element.
- min-width **:-** Sets the minimum width of an element.
- max-width **:-** Sets the maximum width of an element.
- Height **:-** Sets the height of an element.
- min-height **:-** Sets the minimum height of an element.
- max-height **:-** Sets the maximum height of an element.

---

13. CSS FLEXBOX

- display: flex **:-** Enables flexbox layout.
- flex-direction **:-** Defines the direction of flex items.
- flex-wrap **:-** Controls whether items wrap to next line.
- flex-flow **:-** Shorthand for flex-direction and flex-wrap.
- justify-content **:-** Aligns items horizontally.
- align-items **:-** Aligns items vertically.
- align-content **:-** Aligns multiple rows of items.
- Gap **:-** Sets space between flex items.
- Order **:-** Controls the order of flex items.
- flex-grow **:-** Defines how much an item grows.
- flex-shrink **:-** Defines how much an item shrinks.
- flex-basis **:-** Sets the initial size of an item.
- align-self **:-** Aligns a single flex item independently.

---

14. CSS GRID

- display: grid **:-** Enables grid layout.
- grid-template-columns **:-** Defines column structure.
- grid-template-rows **:-** Defines row structure.
- grid-gap **:-** Sets space between grid items.
- grid-area **:-** Defines a grid item's position.

- grid-column **:-** Controls column placement.

- grid-row **:-** Controls row placement.

- justify-items **:-** Aligns items horizontally in grid cells.

- align-items **:-** Aligns items vertically in grid cells.

- place-items **:-** Shorthand for justify-items and align-items.

15. CSS MEDIA QUERIES

- Responsive design **:-** Design that adapts to different screen sizes.

- min-width **:-** Applies styles for screens wider than given width.

- max-width **:-** Applies styles for screens smaller than given width.

- Orientation **:-** Targets portrait or landscape mode.

- Breakpoints **:-** Screen widths where layout changes.

- mobile-first approach **:-** Designing for mobile first, then larger screens.

16. CSS VARIABLES

- :root **:-** Defines global CSS variables.

- var() **:-** Used to apply CSS variables.

- scope of variables **:-** Defines where variables can be accessed.

- dynamic theming **:-** Allows changing theme using variables.

17. CSS SHADOWS

- box-shadow **:-** Adds shadow to elements.

- text-shadow **:-** Adds shadow to text.

- multiple shadows **:-** Allows applying more than one shadow.

18. CSS TRANSITIONS

- transition-property **:-** Specifies which property to animate.

- transition-duration **:-** Sets animation duration.

- transition-timing-function **:-** Controls animation speed curve.

- transition-delay **:-** Delays the start of transition.

- hover effects **:-** Smooth effects triggered on hover.

19. CSS TRANSFORMS (2D & 3D)

- translate() **:-** Moves an element.

- scale() **:-** Resizes an element.

- rotate() **:-** Rotates an element.

- skew() **:-** Tilts an element.

- Perspective **:-** Adds 3D depth effect.

- transform-origin **:-** Defines the rotation origin point.

---

20. CSS ANIMATIONS

- @keyframes **:-** Defines animation steps.

- animation-name **:-** Specifies the animation name.

- animation-duration **:-** Sets animation time.

- animation-delay **:-** Delays animation start.

- animation-iteration-count **:-** Defines number of repetitions.

- animation-direction **:-** Controls animation direction.

- animation-fill-mode **:-** Defines element state after animation.

- animation-timing-function **:-** Controls animation speed curve.