**ASSIGNMENT-1**

(Basic Understanding of Web Development)

**TASK-4: Comprehensive CSS Properties Showcase**

* CSS property is used to set the style or assign behavior of HTML elements. The CSS property contains two parts, property\_name, and property\_value. The property\_value is enclosed within double quotes (" ").
* **background**: Sets the background properties in one declaration. Combines color, image, position, size, etc.
  + background: #fff url('image.jpg') no-repeat center/cover;
* **border**: Sets border width, style, and color in one declaration. Shorthand for border-width, border-style, and border-color.
  + border: 1px solid #000;
* **box-shadow**: Adds shadow effects to an element. Specifies horizontal and vertical offsets, blur radius, spread radius, and color.
  + box-shadow: 10px 10px 5px #888888;
* **color**: Sets the text color of an element. Accepts named colors, hex, RGB, RGBA, HSL, and HSLA values.
  + color: #333;
* **display**: Specifies the display behavior of an element. Common values include block, inline, inline-block, flex, grid, and none.
  + display: block;
* **flex**: Defines how a flex item will grow or shrink within a flex container. Shorthand for flex-grow, flex-shrink, and flex-basis.
  + flex: 1 0 auto;
* **font**: Sets font style, variant, weight, size, line height, and family in one declaration. Shorthand for various font- properties.
  + font: italic small-caps bold 16px/1.5 'Arial';
* **grid**: Defines grid properties in one declaration. Shorthand for grid-template-rows, grid-template-columns, grid-template-areas, etc.
  + grid: "header header" 50px "main aside" 1fr / 1fr 200px;
* **height**: Sets the height of an element. Accepts length units, percentages, or the keywords auto and inherit.
  + height: 100px;
* **margin**: Sets the margin on all four sides of an element in one declaration. Accepts length units, percentages, or auto.
  + margin: 10px 20px 30px 40px;
* **padding**: Sets the padding on all four sides of an element in one declaration. Accepts length units, percentages, or inherit.
  + padding: 10px 20px 30px 40px;
* **position**: Specifies how an element is positioned in the document. Values include static, relative, absolute, fixed, and sticky.
  + position: absolute;
* **text-align**: Sets the horizontal alignment of text within an element. Common values are left, right, center, and justify.
  + text-align: center;
* **transform**: Applies a 2D or 3D transformation to an element. Functions include translate, rotate, scale, skew, etc.
  + transform: rotate(45deg);
* **transition**: Defines the transition effect between two states of an element. Shorthand for transition-property, transition-duration, transition-timing-function, and transition-delay.
  + transition: all 0.3s ease;
* **width**: Sets the width of an element. Accepts length units, percentages, or the keywords auto and inherit.
  + width: 50%;
* **z-index**: Sets the stack order of an element. Only works on positioned elements (those with position other than static).
  + z-index: 10;
* **align-items**: Aligns flex items along the cross axis in a flex container. Values include stretch, center, flex-start, flex-end, and baseline.
  + align-items: center;
* **background-color**: Sets the background color of an element. Accepts named colors, hex, RGB, RGBA, HSL, and HSLA values.
  + background-color: #f0f0f0;
* **border-radius**: Rounds the corners of an element. Accepts length units or percentages.
  + border-radius: 10px;
* **cursor**: Specifies the type of cursor to be displayed when pointing over an element. Common values are default, pointer, move, text, etc.
  + cursor: pointer;
* **flex-direction**: Defines the direction of the main axis in a flex container. Values include row, row-reverse, column, and column-reverse.
  + flex-direction: column;
* **font-family**: Specifies a prioritized list of font family names for text content. Fallback fonts ensure text is displayed if a preferred font is unavailable.
  + font-family: 'Helvetica', sans-serif;
* **font-size**: Sets the size of the font. Accepts length units, percentages, or keywords like small, medium, large.
  + font-size: 16px;
* **font-weight**: Sets the weight (boldness) of the font. Accepts values like normal, bold, or numeric values from 100 to 900.
  + font-weight: bold;
* **justify-content**: Aligns flex items along the main axis in a flex container. Values include flex-start, flex-end, center, space-between, space-around, and space-evenly.
  + justify-content: space-between;
* **line-height**: Sets the height of a line of text. Accepts length units, percentages, or numbers (which act as a multiplier of the font size).
  + line-height: 1.5;
* **list-style**: Sets all the list properties in one declaration. Combines list-style-type, list-style-position, and list-style-image.
  + list-style: square inside url('image.png');
* **max-height**: Sets the maximum height of an element. Accepts length units, percentages, or none.
  + max-height: 200px;
* **max-width**: Sets the maximum width of an element. Accepts length units, percentages, or none.
  + max-width: 100%;
* **min-height**: Sets the minimum height of an element. Accepts length units, percentages, or none.
  + min-height: 100px;
* **min-width**: Sets the minimum width of an element. Accepts length units, percentages, or none.
  + min-width: 50px;
* **opacity**: Sets the opacity level of an element. Accepts values from 0 (fully transparent) to 1 (fully opaque).
  + opacity: 0.75;
* **outline**: Sets the outline properties in one declaration. Combines outline-width, outline-style, and outline-color.
  + outline: 2px dashed #00f;
* **overflow**: Controls what happens to content that overflows an element's box. Values include visible, hidden, scroll, and auto.
  + overflow: hidden;
* **text-align**: Sets the horizontal alignment of text within an element. Common values are left, right, center, and justify.
  + text-align: center;
* **text-decoration**: Adds decoration to text. Values include none, underline, overline, line-through, and combinations thereof.
  + text-decoration: underline;
* **text-transform**: Controls the capitalization of text. Values include none, capitalize, uppercase, and lowercase.
  + text-transform: uppercase;
* **visibility**: Controls the visibility of an element. Values include visible, hidden, and collapse (for table rows/columns).
  + visibility: hidden;
* **white-space**: Controls how whitespace inside an element is handled. Values include normal, nowrap, pre, pre-line, and pre-wrap.
  + white-space: nowrap;
* **width**: Sets the width of an element. Accepts length units, percentages, or the keywords auto and inherit.
  + width: 50%;
* **word-wrap**: Allows long words to be broken and wrap onto the next line. Common value is break-word.
  + word-wrap: break-word;
* **z-index**: Sets the stack order of an element. Only works on positioned elements (those with position other than static).
  + z-index: 10;
* **align-content**: Aligns the flex container's lines within the flex container when there is extra space. Values include stretch, center, flex-start, flex-end, space-between, and space-around.
  + align-content: space-between;
* **align-self**: Allows the default alignment (or the one specified by align-items) to be overridden for individual flex items. Values include auto, flex-start, flex-end, center, baseline, and stretch.
  + align-self: center;
* **animation**: A shorthand property for animation-name, animation-duration, animation-timing-function, animation-delay, animation-iteration-count, animation-direction, animation-fill-mode, and animation-play-state.
  + animation: slidein 3s ease-in-out infinite;
* **background-clip:** The **background-clip** property in CSS is a powerful tool that dictates how a background (be it color or image) extends within an element. It’s all about controlling the reach of your background.
* background-clip:border-box|padding-box|content-box|text|initial|inherit;
* **background-image:** The **background-image** Property is used to set one or more background images for an element. By default, it places the image in the top left corner. To specify two or more images, we need to specify the separate URLs with a comma for both images.
* background-image: url (‘url’) | none | initial |inherit;