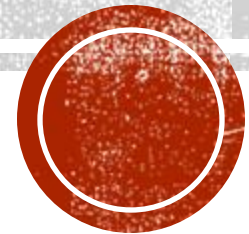


CSS STYLING

Unit-III

SCS181 Web Technologies-I



CSS STYLING (BACKGROUND, TEXT FORMAT, CONTROLLING FONTS)

CSS (Cascading Style Sheets) allows developers to style web pages by controlling aspects such as background, text formatting, and fonts.

1. Background Styling: CSS offers a variety of properties to control the background of HTML elements. These properties allow you to add colors, images, and control their position and behavior.

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

Values: Color names (`red`), hexadecimal values (`#ff0000`), RGB (`rgb(255, 0, 0)`), HSL (`hsl(0, 100%, 50%)`).

Example:

```
body {  
  
    background-color: lightblue;  
  
}
```



CSS STYLING (BACKGROUND, TEXT FORMAT, CONTROLLING FONTS)

b. `background-image`: Sets an image as the background of an element.

Values: URL path to the image (`url('image.png')`).

Example:

```
div {  
  
    background-image: url('background.jpg');  
  
}
```

c. `background-repeat`: Controls if/how the background image is repeated.

Values: `repeat` (default), `no-repeat`, `repeat-x` (only horizontally), `repeat-y` (only vertically).

Example:

```
div {  
  
    background-image: url('pattern.png');  
  
    background-repeat: no-repeat;  
  
}
```



CSS STYLING (BACKGROUND, TEXT FORMAT, CONTROLLING FONTS)

d. `background-position`: Sets the initial position of the background image.

Values: Keywords (`left`, `right`, `center`), or specific values (`20px 30px`).

Example:

```
div {  
  
    background-image: url('banner.png');  
  
    background-position: center;  
  
}
```

e. `background-size`: Defines the size of the background image.

Values: `cover` (scales the image to cover the entire element), `contain` (scales the image to fit inside the element), or specific values (`100px 200px`).

Example:

```
section {  
  
    background-image: url('large-image.jpg');  
  
    background-size: cover;  
  
}
```



CSS STYLING (BACKGROUND, TEXT FORMAT, CONTROLLING FONTS)

f. `background-attachment`: Controls the scrolling behavior of the background image.

Values: `scroll` (default, background scrolls with the page), `fixed` (background stays fixed while scrolling), `local` (scrolls within an element).

Example:

```
div {  
  
    background-image: url('scroll-image.jpg');  
  
    background-attachment: fixed;  
  
}
```



2. TEXT FORMATTING

CSS text formatting allows you to control the appearance and layout of text within elements.

a. color

Sets the color of text.

Values: Color names, hexadecimal, RGB, HSL.

Example:

```
p {  
    color: #333;  
}
```

b. text-align

Aligns text horizontally within its container.

Values: left, right, center, justify.

Example:

```
h1 {  
    text-align: center;  
}
```



2. TEXT FORMATTING

c. text-decoration

Adds or removes decorations such as underlining, overlining, or strikethrough on text.

Values: none, underline, overline, line-through.

Example:

```
a { text-decoration: none; }
```

d. text-transform

Controls the capitalization of text.

Values: none, capitalize, uppercase, lowercase.

Example:

```
h2 { text-transform: uppercase; }
```



2. TEXT FORMATTING

e. letter-spacing

Sets the space between characters (letter spacing).

Values: Length values (normal, 0.1em, 2px).

Example: `h1 { letter-spacing: 3px; }`

f. line-height

Defines the amount of space between lines of text.

Values: Numeric values (1.5), length (px), percentage (%).

Example: `p { line-height: 1.8; }`

g. text-shadow

Applies a shadow effect to the text.

Values: Horizontal offset, vertical offset, blur radius, and color.

Example:

`h1 { text-shadow: 2px 2px 5px
 rgba(0, 0, 0, 0.3); }`



3. FONT CONTROL

CSS provides properties to control the style, size, and appearance of fonts, allowing for creative and structured text designs.

a. font-family

Specifies the font(s) to be applied to text. You can list multiple fonts as fallbacks, where the browser will use the next available font in the list if the first one isn't available.

Values: Font names (e.g., Arial, Verdana), or generic family names (serif, sans-serif, monospace).

Example:

```
body { font-family: 'Arial', sans-serif; }
```

b. font-size

Sets the size of the font. You can use absolute units (px, pt), relative units (em, %), or keywords (small, medium, large).

Values: Length (16px, 1.2em), percentage (100%), or keywords (small, medium, large).

Example:

```
h1 { font-size: 36px; }
```



3. FONT CONTROL

c. font-weight

Controls the thickness of the font, making text bold or light.

Values: normal, bold, bolder, lighter, or numeric values (from 100 to 900).

Example:

```
strong { font-weight: bold; }
```

d. font-style

Defines the style of the font, usually used to italicize text.

Values: normal, italic, oblique.

Example: `em { font-style: italic; }`

e. font-variant

Controls the appearance of small-caps text.

Values: normal, small-caps.

Example: `p { font-variant: small-caps; }`



COMBINING PROPERTIES:

You can combine background, text, and font properties to create unique and visually appealing designs.

Example:

```
body {  
    background-color: #f0f0f0; font-family: 'Arial', sans-serif; color: #333; line-height: 1.6;  
}  
h1 {  
    font-size: 48px; color: #444; text-align: center; text-transform: uppercase; letter-spacing: 2px;  
}  
p {  
    font-size: 18px; margin: 20px 0;  
}
```

