



## Experiment No 1-3

Aim - create web page using navigation bar by using CSS

### Theory -

#### a) Explain text properties - i) Text color

The color property is used to set the color of the text

i) a color name - like "red"

ii) a HEX value - like "#ff0000"

iii) an RGB value - like "rgb(255,0,0)"

e.g  $\Rightarrow$  body { color: blue; }

#### b) Background color

e.g  $\Rightarrow$  body { background-color: lightgrey; }

#### c) Text Alignment

A text can be left or right aligned, centered or justified.

e.g  $\Rightarrow$  h1 { text-align: center; }

#### d) Text Align last

The text-align-last property specifies how to align the last line of a text.

e.g  $\Rightarrow$  p.a { text-align-last: right; }

#### e) Text Direction

The direction & unicode-bidi properties can be used to change the text direction of an element.

e.g  $\Rightarrow$  p { direction: rtl;  
unicode-bidi: bidi-override; }





### e) Vertical Alignment

The vertical-align property sets the vertical alignment of an element.

e.g. `img {vertical-align: text-bottom;}`

### 2) Text Decoration

a) Add a Decoration line to Text - used to add a decoration line to text.

e.g. `h1 {text-decoration-line: overline;}`

### b) Specify a color for the Decoration line

It is used to set the color of the decoration line

e.g. `h1 {text-decoration-color: red;}`

### c) Specify a style for the decoration line

It is used to set the style of the decoration line

e.g. `h1 {text-decoration-style: solid;}`

### 3) Text Transformation

It is used to specify uppercase & lowercase letters in text

e.g. `p.uppercase {text-transform: uppercase;}`

### 4) Text Spacing

#### a) Text Indentation

It is used to specify the indentation of the first line of a text.

e.g. `p {text-indent: 50px;}`

#### b) letter spacing

It is used to specify the space between the characters in a text

e.g  $\Rightarrow$  h1 {letter-spacing: 5px; }

c) Line Height

It is used to specify the space between lines.

e.g  $\Rightarrow$  p.small {line-height: 0.8; }

d) Word spacing

It is used to specify the space between the words in text

e.g  $\Rightarrow$  p.two {word-spacing: -2px; }

• Explain list properties

Two types of list - i) Unordered list

ii) Ordered list

i) Unordered list (<ul>) - the list items are marked with bullets.

e.g  $\Rightarrow$ 

- coffee
- tea

ii) Ordered list (<ol>) - the list items are marked with numbers or letters.

e.g  $\Rightarrow$ 

1. coffee
2. tea

CSS List Properties -

1) list-style - sets all the properties for a list in one declaration

2) list-style-image - specifies an image as the list-item marker

3) list-style-position - specifies the position of the list-item markers (bullet points)

4) list-style-type - specifies the type of list-item marker.

### • Explain Font Properties

Font is type of serif, sans-serif, monospace, cursive, fantasy etc.

#### 1) Font-family

We use the font-family property to specify the font of a text.

e.g. `p {font-family: "Times New Roman", Times, serif;}`

- a) Arial (sans-serif) - font for both online and printed media.
- b) Verdana (sans-serif) - Easily readable even for small font sizes.
- c) Tahoma (sans-serif) - font has less space between the characters.
- d) Trebuchet MS (sans-serif) - Not supported by all mobile os.
- e) Times New Roman (serif) - In many newspapers font used.
- f) Georgia (serif) - Is an elegant serif font.
- g) Garamond (serif) - for many printed books used in font.
- h) Courier New (monospace) - used with coding & email display.
- i) Brush Script MT (cursive) - font was designed to mimic handwriting.

#### 2) Font style

The font-style property is mostly used to specify italic text.

- i) normal - The text is shown normally.
- ii) italic - The text is shown in italic.
- iii) oblique - The text is leaning.

e.g. `p.normal { font-style: normal; }`

#### 3) Font weight

The font-weight property specifies the weight of font.

e.g. `p.normal { font-weight: normal; }`

WHAT IS



#### 4) font-variant

It is used to specify whether or not a text should be displayed in a small-caps font.

e.g. `p { font-variant: normal; }`

#### 5) font-size

The font-size property sets the size of the text.

e.g. `h1 { font-size: 40px; }`

### • Explain Position Properties

The position property specifies the type of positioning method used for an element (static, relative, fixed, absolute & sticky).

#### 1) position: static;

static positioned elements are not affected by the top, bottom, left & right properties. An element with position: static; is not positioned in any special way; it is always positioned according to the normal flow of the page.

This `<div>` element has position: static;

e.g. `div.static { position: static;  
border: 3px solid #73AD21; }`

#### 2) position: relative;

An element with position: relative; is positioned relative to its normal position. Other content will not be adjusted to fit into any gap left by the element. The top, right, bottom & left properties in relatively-positioned from its normal position. This `<div>` element has position: relative;

e.g. `div.relative { position: relative; left: 30px;  
border: 3px solid #73AD21; }`



### 3) position: fixed;

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 `<div>` element has `position: fixed;`

e.g ⇒ `div.fixed { position: fixed; bottom: 0; right: 0; width: 300px; border: 3px solid #73AD21; }`

### 4) position: absolute;

If an absolute positioned element has no positioned ancestors (instead of positioned relative to the viewport, fixed). This `<div>` element has `position: relative;` and `position: absolute;`

e.g ⇒ `div.relative { position: relative; width: 400px; height: 200px; border: 3px solid #73AD21; }`

`div.absolute { position: absolute; top: 80px; right: 0; width: 100px; height: 100px; border: 3px solid #73AD21; }`

### 5) position: sticky;

An element with `position: sticky;` is positioned based on the user's scroll position.

e.g ⇒ `div.sticky { position: -webkit-sticky; /* safari */ position: sticky; top: 0; background-color: red; border: 2px solid #4CAF50; }`

### Define Navigation bar & its types.

Having easy-to-use navigation is important for any website.

Navigation Bar = List of links.

A navigation bar is basically, a list of links, so using the `<ul>` and `<li>` elements makes perfect sense.

With CSS you can transform boring HTML menus into





good-looking navigations bars.

e.g ⇒ <ul>

```
<li><a href = "#home"> Home </a></li>
<li><a href = "#contact"> Contact </a></li>
</ul>
```

Types of Navigation Bar

1) Vertical Navigation Bar

2) Horizontal Navigation Bar.

1) Vertical Navigation Bar

The style <a> elements inside the list, in addition to the code from the previous page.

e.g ⇒ li · a { display: block;  
width: 60px; }

a) display: block; → Displaying the links as block elements makes the whole link area clickable ↗ width, height.

b) width: 60px; → Block elements takes up the full width available by default. We want to specify a 60 pixels width.

O/P ⇒ Home

News

Contact

About

2) Horizontal Navigation Bar

Two ways to create a horizontal navigation bar,  
Using inline or floating list items



### a) Inline List Items

One way to build a horizontal navigation bar the `<li>` elements as inline, in addition "standard" code from previous page.

e.g. `→ li { display: inline; }`

`display: inline;` → We remove the line breaks before `</li>` after each list item; to display them on one line.

### b) Floating List Items

An horizontal navigation bar to float `<li>` elements & specify a layout for the navigation links.

e.g. `→ li { float: left; }`

`a { display: block; padding: 8px; background-color: #cccccc; }`

`float: left;` → block elements to float next to each other.

`display: block;` → allows us to specify padding.

`padding: 8px;` → `<a>` element, to make them look good.

`background-color: #cccccc` → add gray background-color to each `<a>` element.

### Conclusion -

In this practice, we using HTML tags and CSS properties are using text properties, list properties, font & position properties. and having easy to use navigation is important for any website.