# CSS

Shristi Technology Labs

# Agenda

- What is CSS?
- Style and structure
- Types of CSS
- Handling Backgrounds
- Working with Fonts
- Styling text
- Using class and Id
- Styling Links
- Working with Lists

# What is CSS?

- CSS means “Cascading Style Sheet”
- Handles the look and feel part of a web page.
- Change the text color, font style, the paragraph spacing
- Can be combined with the markup languages HTML or XHTML
- Easy to learn and understand
- Provides powerful control over the presentation of a HTML document

# Style and structure

- Has style rules that are interpreted by the browser.

**Selector**

- is a HTML tag at which style will be applied.
- eg. **<h1>** or **<table>**

**Property :**

- Is a type of attribute of HTML tag like color, font etc

**Value :**

- Are assigned to properties

```
<style type="text/css">
    selector { property :value; }
</style>
```

# Types of CSS

- Internal CSS
- External CSS
- Inline CSS

# Internal Stylesheet

- is used in the same html page
- Is defined in head section and inside the <style> tags.

```
<html>
    <head>
        <style type="text/css">
        body {
            background-color: red;
        }
        h1{
            color: black;
            margin-left: 100px;
        }
        </style>
    </head>
    <body>
    <h1> This is a heading </h1>
    <p> This content of pragraph </p>
    </body>
</html>
```

# External Stylesheet

- Is written in a separate file and saved in ***anyname.css***
- is ideal when same style has to be applied to many pages.
- Can change the look of an entire web site by changing a single css file

```
<html>
<head>

    <link rel="stylesheet" type="text/css" href="mystyle.css">

</head>
<body>

<p> This is the main method of the content</p>
</body>
</html>
```

# Inline Stylesheet

- Is used inside any tag using **style** attribute
- **style** attribute can contain any CSS property.

```
<h1 style="color:red;background-color:aqua">Heading </h1>
```

# CSS Property

- Background
- Text
- Fonts
- Links
- Lists
- Display
- Floating

# CSS -Background

**background-color:** to set the background color of an element.

- background-color : pink;

**background-image:** to set the background image of an element.

- background-image : url (“butterfly.gif");

**background-position**: to control the position of an image in the background.

- background-position : top left;

**background-attachment** property is used to control the scrolling of an image

- background-attachment : scroll;

**background-repeat** is used to control the repetition of an image.

- background-repeat : repeat;

**background** is shorthand to specify a number of other background property

- background: #FFCC66 url("butterfly.gif") repeat scroll ;

# Example

```
body{

background-color:blue;
background-image:url("butterfly.jpg");
background-attachment:scroll;
background-position:top left;
background-repeat:repeat;
}
 h1{

    background: blue url("butterfly.jpg") scroll top left
         repeat;
 }
```

# CSS - Fonts

- **font-family** is used to change the face of a font.
  - font-family: sans-serif;
- **font-style** used to make a font italic or oblique.
  - font-style: italic;
- **font-variant** used to create a small-caps effect.
  - font-variant : small – caps ;
- **font-weight** used to increase or decrease the boldness for a font .
  - font-weight: bold;
- **font-size** used to increase or decrease the size of a font.
  - font-size: 20px;
- **font** is a shorthand to specify a number of other font properties.
  - font: sans-serif italic bold 20px;

# Example

```
h1{
    font-family:  sans-serif;
    font-style: italic;
    font-variant: small-caps;
    font-size: 20px;
    font-weight: bolder;

    }
p{
    font: sans-serif italic small-caps 20px bolder ;
}
```

# CSS – Text

- **color** is used to set the color of a text.
  - color: red;
- **Direction** is used to set the text direction.
  - direction : rlt;
- **letter-spacing** is used to add or subtract space between the letters in a word.
  - letter-spacing: 5px;
- **word-spacing** is used to add/subtract space between the words of a sentence.
  - word-spacing : 5px;
- **text-indent** is used to indent the text of a paragraph.
  - text-indent: 1cm;

# CSS – Text

- **text-align** is used to align the text of a document.
  - text-align : right;
- **text-decoration** is used to underline, overline, or strikethrough text.
  - text-decoration : underline;
- **text-transform** is used to capitalize text or convert text to uppercase or lowercase letters.
  - text-transform: uppercase;
- **white-space** is used to control the flow and formatting of text.
  - white-space : pre;
- **text-shadow** is used to set the text shadow around a text.
  - text-shadow : 4px 4px blue;

# Example

```
h1{

    color: red;
    direction: ltr;
    letter-spacing: 2px;
    word-spacing: 2px;
    text-indent: 2cm;
    text-decoration: underline;
    text-transform: lowercase;
    text-align: right;
    text-shadow: 4px 2px blue;
}
```

# Using class

- To give same formatting to different tags.
- It is referred by .

```
<p class="check" >Hello in p tag</p>
<div class="check">Hello I am inside a DIV</div>
<h1 class="check" >Hello I am inside a h1 tag</h1>
```

```
.check {
    background-color: blue;
    color: yellow;
}
p {
    color: green;
    font-variant: small-caps;
    font-family: sans-serif;
    font-style: italic;
}
p.check{
    background-color: teal;
    color: white;
    text-decoration: underline;
}
```

# Using id

- To give unique formatting for an individual tag.
- It is referred by **#**

```
<p id="myid"> This is  paragraph - 1 with myid</p>
<p> This is a paragraph - 2. welcome</p>
<h1 > This is a header</h1>
<h1 id = "header"> This is a header with id</h1>
```

```
p{
 background-color: pink;
 border:solid 1px;
 font-weight:bold;
 color: maroon;
}
h1{
background-color: green;
color:fuchsia;
 }
#myclass{
color:yellow;
text-transform: uppercase;
}
#header{
    letter-spacing: 10px;
    color: red;
}
```

# CSS - Links

- **:link** for unvisited hyperlinks.
  - **a : link {color: orange}**
- **:visited** for visited hyperlinks.
  - **a : visited { color: black}**
- **:hover** done on an element that currently has the user's mouse pointer hovering over it.
  - **a : hover { color: yellow}**
- **:active** done on an element on which the user is currently clicking.
  - **a : active {color: green}**

# Example

```
a:link{
    color:green;
    }
a:visited{
    color:navy;

    }
a: active{
    color:fuchsia;
    }
a: hover {
    text-transform: uppercase;
    font-stretch: wider;
    color: orange;
}
```

# CSS - List

- **list-style-type** control the shape or appearance of the marker.
  - **list- style- type: circle;**
- **list-style-image** an image for the marker rather than a bullet point or number.
  - **list- style-image: url (butterfly.gif);**

# CSS - Floating

- Elements are floated horizontally, either left or right only.
- Floating an image to the right/left of text.
- Do  not float  up or down in CSS.
- Elements before the floating element will not be affected.
  - float: right; float: left;
- Turning off float – using clear
  - clear: both;

# Example

```
<style type="text/css">
img{
    float: right;
}
h1{
    clear: right;
}
</style>
</head>
<body>
     This is in body
    <span>Hello how are U</span>
    <img src="panda1.jpg" width="200px" height="200px">
    <h1>welcome to css</h1>
    <h1>Hello world</h1>
```

# CSS - Display

- Hiding an element can be done by setting the property
  - **display :none, visibility :hidden**

**visibility: hidden**

- Hides an element but it takes up the same space as before the element will be hidden

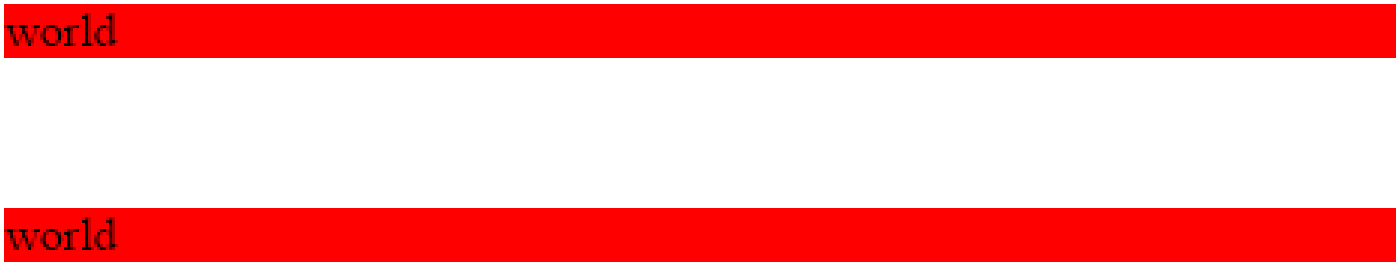
**display :none**

- Hides an element, and it will not any up space

# Example - Visiblity

```
<body>
    <p style="background-color:red">world</p>
    <p style="visibility:hidden;background-color:red">Hello</p>
    <p style="background-color:red">world</p>
</body>
```

**Output in Browser**



# Example - Display

```
<body>
    <p style="background-color:red">world</p>
    <p style="display:none;background-color:red">Hello</p>
    <p style="background-color:red">world</p>
</body>
```

**Output in Browser**

world

world

# CSS - block and inline elements

- A block element is an element that takes up the full width available.
- **Example for block elements**
  - <h1>, <p>, <li>,
- **Example of inline elements**
  - , <a>

# CSS – Positioning

- static
- fixed
- relative
- absolute

# Positioning - static

- Static positioning is the default positioning model for elements.
- They are displayed in the page where they rendered as part of normal HTML flow.
- They don't obey left, top, right and bottom rules

# Positioning - relative

- Relative positioning allows to specify a specific offset (left, top etc) which is relative to the element's normal position in HTML flow.

# Positioning - absolute

- The element is positioned relative to the nearest positioned ancestor.
- They are still bound by the viewport and will cause scrolling

```
#div3 {
    position: relative
}
#header3 {
    position: absolute;
    left: 200px;
    top: 100px;
}
```

- If there are no ancestors, then positioned relative to the entire document body.

```
#header3 {
    position: absolute;
    left: 200px;
    top: 100px;
}
```

```
<div id="div3">hello
    <h1 id="header3">This is heading3</h1>
</div>
```

# Positioning - fixed

- Fixed positioning restricts an element to a specific position in the viewport, which stays in place during scroll

- They do not cause scroll because they are not considered to be bound by the viewport:

# Advantages

- Saves time
- Pages load faster
- Easy maintenance
- Superior style to html
- Multiple device compatibility
- Global web standards

# Thank You