

# Web Technologies

# Unit 1: Contents

**UNIT-I: HTML** :Basic Syntax, Standard HTML Document Structure, Basic Text Markup, Images, Hypertext Links, Lists, Tables, Frames Forms.

**CSS: Cascading style sheets** : Levels of Style Sheets, Style Specification Formats, Selector Forms, Property value forms, Font Properties, List Properties, color, Alignment of Text

# Introduction to HTML

- HTML is the standard markup language for Web pages.
- With HTML you can create your own Website.
- HTML is easy to learn - You will enjoy it!
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

# HTML tags

- HTML markup tags are usually called HTML tags
- HTML tags are keywords (tag names) surrounded by angle brackets like `<html>`
- HTML tags normally come in pairs like `<b>` and `</b>`
- The first tag in a pair is the start tag, the second tag is the end tag
- The end tag is written like the start tag, with a forward slash before the tag name
- Start and end tags are also called opening tags and closing tags

# History of HTML

Year	Version
1989	Tim Berners-Lee invented www
1991	Tim Berners-Lee invented HTML
1993	Dave Raggett drafted HTML+
1995	HTML Working Group defined HTML 2.0
1997	W3C Recommendation: HTML 3.2
1999	W3C Recommendation: HTML 4.01
2000	W3C Recommendation: XHTML 1.0
2008	WHATWG HTML5 First Public Draft
2012	<u><a href="#">WHATWG HTML5 Living Standard</a></u>
2014	<u><a href="#">W3C Recommendation: HTML5</a></u>
2016	W3C Candidate Recommendation: HTML 5.1
2017	<u><a href="#">W3C Recommendation: HTML5.1 2nd Edition</a></u>
2017	<u><a href="#">W3C Recommendation: HTML5.2</a></u>

# Structure of HTML Page

```
<html>
```

```
<head>
```

```
<title>Page title</title>
```

```
</head>
```

```
<body>
```

```
<h1>This is a heading</h1>
```

```
<p>This is a paragraph.</p>
```

```
<p>This is another paragraph.</p>
```

```
</body>
```

```
</html>
```

# Some Important HTML Tags

Tag	Description
<code>&lt;html&gt; ... &lt;/html&gt;</code>	Declares the Web page to be written in HTML
<code>&lt;head&gt; ... &lt;/head&gt;</code>	Delimits the page's head
<code>&lt;title&gt; ... &lt;/title&gt;</code>	Defines the title (not displayed on the page)
<code>&lt;body&gt; ... &lt;/body&gt;</code>	Delimits the page's body
<code>&lt;h <i>n</i>&gt; ... &lt;/h<i>n</i>&gt;</code>	Delimits a level <i>n</i> heading
<code>&lt;b&gt; ... &lt;/b&gt;</code>	Set ... in boldface
<code>&lt;i&gt; ... &lt;/i&gt;</code>	Set ... in italics
<code>&lt;center&gt; ... &lt;/center&gt;</code>	Center ... on the page horizontally
<code>&lt;ul&gt; ... &lt;/ul&gt;</code>	Brackets an unordered (bulleted) list
<code>&lt;ol&gt; ... &lt;/ol&gt;</code>	Brackets a numbered list
<code>&lt;li&gt; ... &lt;/li&gt;</code>	Brackets an item in an ordered or numbered list
<code>&lt;br&gt;</code>	Forces a line break here
<code>&lt;p&gt;</code>	Starts a paragraph
<code>&lt;hr&gt;</code>	Inserts a horizontal rule
<code>&lt;img src="..."&gt;</code>	Displays an image here
<code>&lt;a href="..."&gt; ... &lt;/a&gt;</code>	Defines a hyperlink

Listing of the HTML tags found during the survey, distinguishing between the correct and incorrect ones.

Correct HTML Tags	Correct HTML Tags	Correct HTML Tags	Incorrect HTML Tags	Incorrect HTML Tags	Incorrect HTML Tags
a	Em	option	abcsbmit	gave	ps
address	Font	p	adress	grin	rfpicr
align	Form	param	ahref	hmshade	shift
applet	Frame	pre	aired	htm	silence
area	Frameset	right	aircheck	http	stations
b	h[1..6]	script	and	il	svd
base	head	select	are	imgsrc	tdalign
basefont	hr	small	bgsound	inputtype	tdcolspan
blink	html	strong	bodybgcolor	is	tdnowrap
blockquote	i	sub	border	it	tdwidth
e	img	table	brt	jberg	textareawrap
body	input	td	by	jpd	thank
br	left	textarea	ceneter	krone	tsfinfo
caption	li	th	centre	m	valign
center	kbd	title	clear	marquee	vk
cite	link	tr	color	means	w
code	map	tt	colw	moore	wbr
dd	menu	u	embed	name	were
dir	meta	ul	emp	nobr	width
div	noframes	width	front	of	wireless
dl	ol		fontsize	palign	www
dt					



# HTML Example

```
<!DOCTYPE html>  
<html>  
  <body>  
    <h1>My First Heading</h1>  
    <p>My first paragraph.</p>  
  </body>  
</html>
```

# Explanation

- The DOCTYPE declaration defines the document type
- The text between `<html>` and `</html>` describes the web page
- The text between `<body>` and `</body>` is the visible page content
- The text between `<h1>` and `</h1>` is displayed as a heading
- The text between `<p>` and `</p>` is displayed as a paragraph

# What is HTML?

- HTML, otherwise known as **HyperText Markup Language**, is the language used to **create Web pages**
- Using HTML, you can create a Web page with **text, graphics, sound, and video**.
- Hyper means we can navigate from one web page to another page where they need not be linear pages.

# Tags

- HTML is written in the form of tags
- A tag is a keyword enclosed by pair of angle brackets ( Example: < > )
- Where some text is placed between tags.
- HTML elements have two basic properties

➤ Attributes

➤ contents

# More Tags...

- The opening and closing tags use the same command except the closing tag contains an additional forward slash /
- For example, the expression `<b>Warning</b>` would cause the word 'Warning' to appear in bold face on a Web page.
- There are some tags which have an opening tag but not a closing tag, also known as Empty HTML Elements.

Eg - `<br>`, `<hr>` i.e Break and Horizontal rule

# Nested Tags

- Whenever you have HTML tags within other HTML tags, you must close the nearest tag first
- Example:  
`<h1> <i> The Nation </i> </h1>`

# Structure of a Web Page

- All Web pages share a common structure
- All Web pages should contain a pair of `<HTML>`, `<HEAD>`, `<TITLE>`, and `<BODY>` tags.

`<html>`

`<head>`

`<title> Example </title>`

`</head>`

`<body>`

This is where you would include the text and images on your Web page.

`</body>`

`</html>`

# Comment Statements

- Comment statements are notes in the HTML code that explain the important features of the code
- The comments do not appear on the Web page itself but are a useful reference to the author of the page and other programmers
- To create a comment statement use the `<!-- Write ur comment here -->` tags



# HTML Elements

An HTML element is defined by a start tag, some content, and an end tag.

## HTML Elements

The HTML **element** is everything from the start tag to the end tag:

**<tagname>Content goes here...</tagname>**

Examples of some HTML elements:

`<h1>My First Heading</h1>`

`<p>My first paragraph.</p>`

# HTML Attributes

- All HTML elements can have **attributes**
- Attributes provide **additional information** about elements
- Attributes are always specified in **the start tag**
- Attributes usually come in name/value pairs like: **name="value"**

For example:

The href Attribute

The <a> tag defines a hyperlink.

The href attribute specifies the URL of the page the link goes to:

```
<a href="https://www.w3schools.com">Visit W3Schools</a>
```

# The <TITLE> Tag

- Choose the title of your Web page carefully; The title of a Web page determines its ranking in certain search engines
- The title will also appear on Favorite lists, History lists, and Bookmark lists to identify your page

# Headings

- Web pages are typically organized into sections with headings; To create a heading we use the expression
- `<Hn>....</Hn>` where n is a number between 1 and 6
- In this case, the 1 corresponds to the largest size heading while the 6 corresponds to the smallest size

# Text Formatting

- Manipulating text in HTML can be tricky; Oftentimes, what you see is NOT what you get
- For instance, special HTML tags are needed to create paragraphs, move to the next line, and create headings

# Text Formatting Tags

`<b> Bold Face </b>`

`<strong> This is also same like bold but shows importance </strong>`

`<i> Italics </i>`

`<u> Underline </u>`

`<p> New Paragraph </p>`

`<br> Next Line`

# Example on Text Formatting Tags

```
<!DOCTYPE html">
<head>
<title>Example on Text Formatting Tags</title>
</head>
<body>
<p><b> Bold Face </b> <br>
  <strong> this is also same like bold but shows importance </strong> </p>
<p><i>The text will appear as Italics </i> </p>
<p><u>This text will appear in Underline </u> </p>
<p> New Paragraph will be started from this tag,
```

we can write the paragraph as

```
many number of lines and sentences </p>
<br>
</body>
</html>
```

# Other Formatting Tags

<pre>- Preformatted Text

<mark> - Marked text

<small> - Smaller text

<del> - Deleted text

<ins> - Inserted text

<sub> - Subscript text

<sup> - Superscript text



## pre: Defines preformatted text

The `<pre>` tag defines preformatted text.

- Text in a `<pre>` element is displayed in a fixed-width font, and the text preserves both spaces and line breaks. The text will be displayed exactly as written in the HTML source code.

# `<mark>`- Marked text

The HTML `<mark>` element defines text that should be marked or highlighted:

## **Example**

```
<p> Please come with your <mark>  
  Observation and Record </mark> for  
  your WT Lab  
</p>
```

# <small> Tag

- The HTML <small> element defines smaller text:

Example:

<h6> this will display heading is small font </h6>

<small> This will display text in very small  
</small>

**Note:** h6 is smaller than compared with small attribute.

# `<del>` - Deleted Text tag

The HTML **`<del>`** element defines text that has been deleted from a document. Browsers will usually strike a line through deleted text:

## **Example:**

```
<p>My    Interested    Job    is    <del>  
Software</del> Government. </p>
```

# `<ins>` - Inserted text

The HTML `<ins>` element defines a text that has been inserted into a document. Browsers will usually underline inserted text:

## **Example:**

```
<p>
```

```
My      Interested      Job      is      <del>  
Software</del>      <ins>Government      </ins>.  
</p>
```

# <sub> - Subscript text

The HTML <sub> element defines subscript text. Subscript text appears half a character below the normal line, and is sometimes rendered in a smaller font. Subscript text can be used for chemical formulas, like H<sub>2</sub>O:

## Example:

```
<p>This      is      an      example      on  
  <sub>subscripted</sub> text.</p>
```

# HTML <sup> Element

The HTML <sup> element defines superscript text. Superscript text appears half a character above the normal line, and is sometimes rendered in a smaller font. Superscript text can be used for footnotes, like WW W<sup>[1]</sup>:

<p>This is an example on  
<sup>superscripted</sup> text.</p>

# Example on Text Formatting Tags

```
<!DOCTYPE html>
```

```
<head>
```

```
<title>Text Formatting tags </title>
```

```
</head>
```

```
<body>
```

```
<pre>Please come with your  
    Observation and Record  
        for your WT Lab </pre>
```

```
<p> Please come with your <mark> Observation and Record </mark> for your WT Lab  
</p>
```

```
<h6> this will display heading is small font </h6>
```

```
<small> This will display text in very small </small>
```

```
<p>My Interested Job is <del> Software</del> Government. </p>
```

```
<p>
```

```
My Interested Job is <del> Software</del> <ins>Government </ins>. </p>
```

```
<p>This is an example on <sub>subscripted</sub> text.</p>
```

```
<p>This is an example on <sup>superscripted</sup> text.</p>
```

```
</body>
```

```
</html>
```



# Tables

- Tables can be used to display rows and columns of data, create multi-column text, captions for images, and sidebars
- The `<table>` tag is used to create a table;  
the `<tr>` tag defines the beginning of a row while  
the `<td>` tag defines the beginning of a cell  
`<th>` is used to define the table heading for a cell  
`<thead>` is used to define the heading for the entire  
table

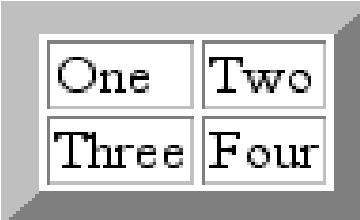
# Adding a Border

- The `BORDER=n` attribute allows you to add a border `n` pixels thick around the table
- To make a solid border color, use the `BORDERCOLOR="color"` attribute

# Creating Simple Table

```
<TABLE BORDER=10>  
  <TR>  
    <TD>One</TD>  
    <TD>Two</TD>  
  </TR>  
  <TR>  
    <TD>Three</TD>  
    <TD>Four</TD>  
  </TR>  
</TABLE>
```

- Here's how it would look on the Web:



One	Two
Three	Four

# Adjusting the Width

- When a Web browser displays a table, it often adds extra space. To eliminate this space use the `WIDTH =n` attribute in the `<TABLE>` and `<TD>` tags
- Keep in mind - a cell cannot be smaller than its contents, and if you make a table wider than the browser window, users will not be able to see parts of it.

# Centering a Table

- There are two ways to center a table
  - Type `<TABLE ALIGN=CENTER>`
  - Enclose the `<TABLE>` tags in opening and closing `<CENTER>` tags

# Wrapping Text around a Table

- It is possible to wrap text around a table. This technique is often used to keep images and captions together within an article.
- To wrap text around a table, type `<TABLE ALIGN = LEFT>` to align the table to the left while the text flows to the right.
- Create the table using the `<TR>`, `<TD>`, and `</TABLE>` tags as you normally would

# Adding Space around a Table

- To add space around a table, use the HSPACE=n and VSPACE=n attributes in the <TABLE> tag
- Example:  
    <TABLE HSPACE=20 VSPACE=20>

# Nesting Tables

- Create the inner table
- Create the outer table and determine which cell of the outer table will hold the inner table
- Test both tables separately to make sure they work
- Copy the inner table into the cell of the outer table
- Don't nest too many tables. If you find yourself doing that, find an easier way to lay out your Web page



# Changing a Cell's Color

- To change a cell's color, add the BGCOLOR="color" attribute to the <TD> tag
- Example:  
    <TD BGCOLOR="blue">

# Colspan and Rowspan Attributes

You will use **colspan** attribute if you want to merge two or more columns into a single column. Similar way you will use **rowspan** if you want to merge two or more rows.

# Changing the Font

- The expression `<font face = "fontname"> ... </font>`  
can be used to change the font of the enclosed text
- To change the size of text use the expression `<font size=n> .... </font>` where n is a number between 1 and 7

# Changing the Font

- To change the color, use `<font color="red">.... </font>`; The color can also be defined using hexadecimal representation ( Example: #ffffff )
- These attributes can be combined to change the font, size, and color of the text all at once; For example, `<font size=4 face="Courier" color="red"> .... </font>`

# Aligning Text

- The **align** attribute can be inserted in the `<p>` and `<h1>` tags to right justify, center, or left justify the text
- For example, `<h1 align=center> The New York Times </h1>` would create a centered heading of the largest size

# Page Formatting

- To define the background color, use the BGCOLOR attribute in the <BODY> tag
- To define the text color, use the TEXT attribute in the <BODY> tag
- To define the size of the text, type <BASEFONT SIZE=n>

# Example

```
<html>
```

```
<head>
```

```
<title> First Example Program </title>
```

```
</head>
```

```
<body>
```

```
<h1 style="background-color:tomato;">
```

This is where you would include the text and images on your Web page.

```
</h1>
```

```
</body>
```

```
</html>
```

# HTML Lists

HTML offers web authors three ways for specifying lists of information. All lists must contain one or more list elements. Lists may contain –

**<ul>** – An unordered list. This will list items using plain bullets.

**<ol>** – An ordered list. This will use different schemes of numbers to list your items.

**<dl>** – A definition list. This arranges your items in the same way as they are arranged in a dictionary.



# HTML Un-Ordered List

An unordered list is a collection of related items that have no special order or sequence. This list is created by using HTML `<ul>` tag. Each item in the list is marked with a bullet.

# Example

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Unordered List</title>
</head>
<body>
<ul>
  <li>HTML</li>
  <li>CSS</li>
  <li>Java Script</li>
  <li>Angular</li>
</ul>
</body>
</html>
```

# The type Attribute

You can use **type** attribute for <ul> tag to specify the type of bullet you like. By default, it is a disc. Following are the possible options

```
<ul type = "square">
```

```
<ul type = "disc">
```

```
<ul type = "circle">
```

# HTML Ordered Lists

If you are required to put your items in a numbered list instead of bulleted, then HTML ordered list will be used. This list is created by using **<ol>** tag. The numbering starts at one and is incremented by one for each successive ordered list element tagged with **<li>**

# Ordered List

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Unordered List</title>
</head>
<body>
<ol>
  <li>HTML</li>
  <li>CSS</li>
  <li>Java Script</li>
  <li>Angular</li>
</ol>
</body>
</html>
```

# The type Attribute

You can use **type** attribute for `<ol>` tag to specify the type of numbering you like. By default, it is a number. Following are the possible options –

`<ol type = "1">` - Default-Case Numerals.

`<ol type = "I">` - Upper-Case Numerals.

`<ol type = "i">` - Lower-Case Numerals.

`<ol type = "A">` - Upper-Case Letters.

`<ol type = "a">` - Lower-Case Letters.

# Html definition/description list

In order to create the definition or description list in html, we have 3 important tags. They are as follows:

`<dl>` this is root element of definition list `</dl>`

`<dt>` defines the definition type or name of data

`<dd>` defines the data of that definition type

# Example on definition list

```
<dl>  
  <dt>Programming</dt>  
    <dd>C</dd>  
    <dd>CPP</dd>  
    <dd>JAVA</dd>  
  <dt>Designing</dt>  
    <dd>HTML</dd>  
    <dd>CSS</dd>  
</dl>
```



# HTML FORMS

HTML Forms are required when you want to collect some data from the site visitor. For example during user registration you would like to collect information such as name, email address, credit card, etc. A form will take input from the site visitor and then will post it to a back-end application such as database, ASP Script or PHP script etc.

# Example

Form Elements Example - Windows Internet Explorer

C:\Users\praveen\Desktop\WEB TECHNOLOGIES\TMPhd

Bing

Favorites | Suggested Sites | Web Slice Gallery

Form Elements Example

Home | RSS | Mail | Print | Page | Safety | Tools | ?

## Registration Form

First Name :

Last Name :

Gender :  ▼

Branch : ☐ Btech ☐ M.Tech

Done

Computer | Protected Mode: Off

90%

# Html form tag

form -create an html form,  
contains form elements . form  
elements .

<form>

form elements

</form>

# Form elements

Form Elements are text fields, text area fields, drop-down menus, radio buttons, check boxes.

## Sub-Element

`<input>` element which contains main attribute type.

```
<input type="text">  
<input type="radio">  
<input type="submit">
```

# text type

## Text Input

**<form>**

**First name: <br>**

**<input type="text" name="firstname"><br>**

**Last name:<br>**

**<input type="text" name="lastname">**

**</form>**

# Radio Button Input

- Input type="radio" defines a radio button

```
<html>
```

```
<body>
```

```
<form>
```

```
First name: <br>
```

```
<input type="radio" name="gender" value="Male" > Male <br>
```

```
<input type="radio" name="gender" value="female"> Female<br>
```

```
<input type="radio" name="gender" value="other"> Other <br>
```

```
</form>
```

```
</body>
```

```
</html>
```

# Other tags

- `<input type =“ checkbox” >`
- `<input type =“ file”>`
- `<textarea rows=“5” cols=“100”>`
- `</textarea>`

# Type = Submit button attribute

```
<body>
```

```
<form action="a.txt">
```

```
First name:<br>
```

```
<input type="text" name="firstname"> <br>
```

```
Password:<br>
```

```
<input type="password" name="pass" ><br><br>
```

```
<input type="submit" value="Submit">
```

```
</form>
```



# File Upload Box

```
<body>
```

```
<form>
```

```
<input type="file" name="fileupload"/>
```

```
</form>
```

```
</body>
```

# Html form Elements

Some of the main elements in HTML form tag is:

- `<input>`
- `<label>`
- `<select>`
- `<textarea>`
- `<button>`
- `<fieldset>`
- `<legend>`
- `<datalist>`
- `<output>`
- `<option>`
- `<optgroup>`

# label element

The **<label>** element defines a label for several form elements. The **for** attribute of the **<label>** tag should be equal to the **id** attribute of the **<input>** element to bind them together.

Syntax:

```
<label for="fname">First name:</label>  
<input type="text" id="fname" >
```

## The select Element

The select element defines a drop-down list:

Choose a car:

Volvo

Saab

Fiat

Audi

# The select element

## The <select> Element

- The <select> element defines a drop-down list:

### Example

```
<label for="cars">Choose a car:</label>
<select id="cars" name="cars">
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="fiat">Fiat</option>
  <option value="audi">Audi</option>
</select>
```

# <option> element

- The <option> element defines an option that can be selected.
- By default, the first item in the drop-down list is selected.
- To define a pre-selected option, add the selected attribute to the option:

## Syntax:

```
<option value="fiat" selected>Fiat</option>
```

Choose a car:

# Allow multiple selections

- Use the multiple attribute to allow the user to select more than one value:

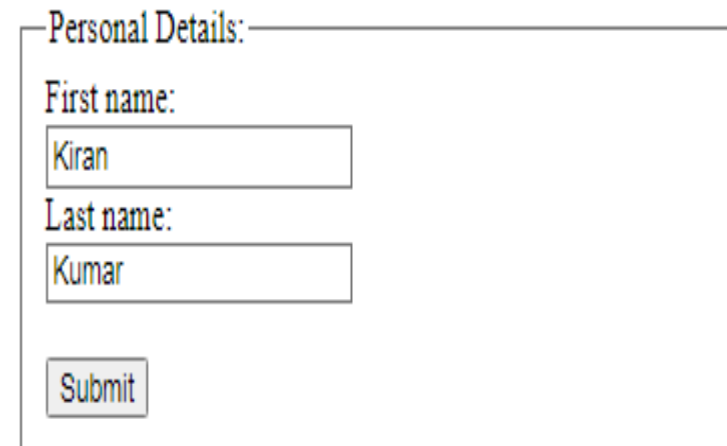
## Syntax:

```
<label for="cars">Choose a car:</label>  
<select id="cars" name="cars" size="4" multiple>  
  <option value="volvo">Volvo</option>  
  <option value="saab">Saab</option>  
  <option value="fiat">Fiat</option>  
  <option value="audi">Audi</option>  
</select>
```

# The <fieldset> and <legend> Elements

- The <fieldset> element is used to group related data in a form.
- The <legend> element defines a caption for the <fieldset> element.

```
<form action="/action_page.php">
  <fieldset>
    <legend>Personal Details:</legend>
    <label for="fname">First name:</label><br>
    <input type="text" id="fname" name="fname"
value="Kiran"><br>
    <label for="lname">Last name:</label><br>
    <input type="text" id="lname" name="lname"
value="Kumar"><br><br>
    <input type="submit" value="Submit">
  </fieldset>
</form>
```



Personal Details:

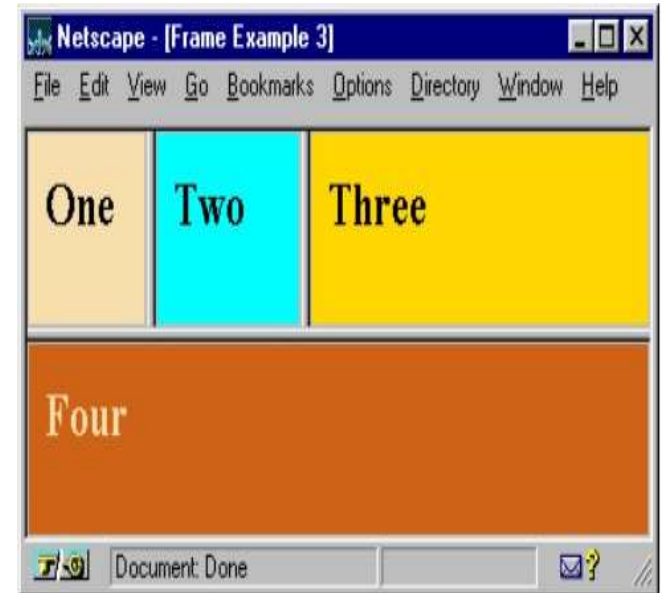
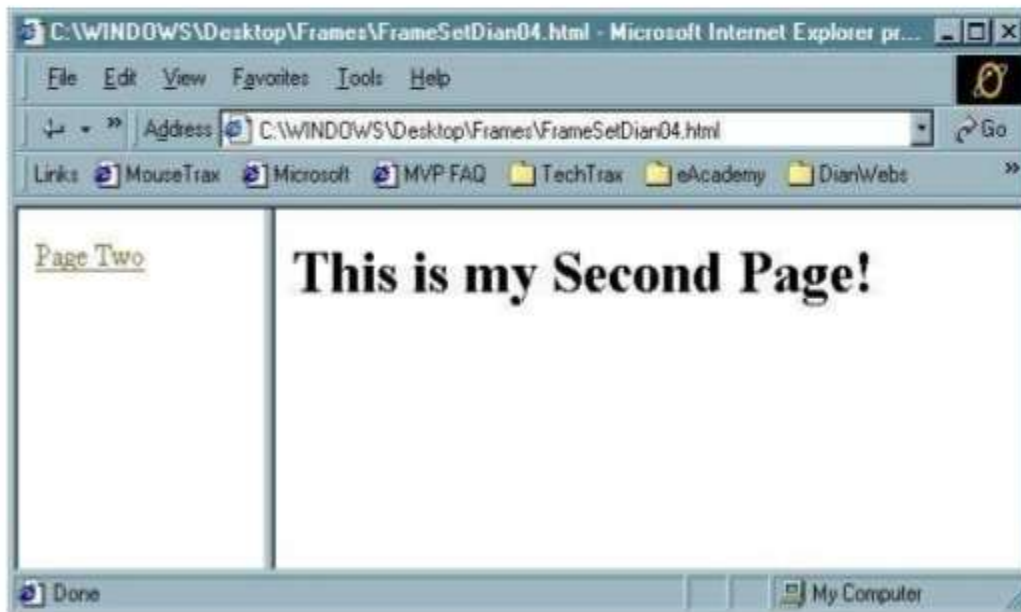
First name:  
Kiran

Last name:  
Kumar

Submit

# Html Frames

- Frames – enables the browser to display more than one HTML document on the same window.





# Frame Creation

## HTML Frames -Creating Frames

- Divide your browser window into multiple sections where each section can load a separate HTML document.
- A collection of frames in the browser window - **A frameset.**
- The window is divided into frames with rows and columns.

### **<frameset> tag instead of <body> tag**

- The rows attribute of <frameset> tag defines horizontal frames  
cols attribute defines vertical frames.
- Each frame is indicated by **<frame> tag and it defines which** HTML document shall open into the frame.

This <frame> tag is currently not used in HTML5, so we need to **use<iframe>** instead of that.

# Syntax

```
<html>
```

```
<frameset cols="70%,30%">
```

```
<frame src="https://developer.mozilla.org/en/HTML/Element/iframe" />
```

```
<frame src="https://developer.mozilla.org/en/HTML/Element/frame" />
```

```
</frameset>
```

```
</html>
```

# Cascading Style Sheets(CSS)

- CSS stands for Cascading Style Sheets
- Styles define how to display HTML elements
- Styles were added to HTML 4.0 to solve a problem
- External Style Sheets can save a lot of work
- External Style Sheets are stored in CSS files
- The extension for external style sheets is .css

# CSS Solved a Big Problem ?

- HTML was never intended to contain tags for formatting a document.
- HTML was intended to define the content of a document, like:

`<h1>This is a heading</h1>`

`<p>This is a paragraph.</p>`

When tags like <font>, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers.

Development of large web sites, where fonts and color information were added to every single page, became a long and expensive process.

To solve this problem, the World Wide Web Consortium (W3C) created CSS.

In HTML 4.0, all formatting could be removed from the HTML document, and stored in a separate CSS file. All browsers support CSS today.

# CSS Syntax

## CSS Syntax

The CSS syntax is made up of three parts: a selector, a property and a value:

```
selector { property : value ;Font-family:"times"  
;color:red ; size:10; }
```

The selector is normally the **HTML element/tag** you wish to define, the property is the **attribute** you wish to change, and each property can take a value. The property and value are separated by a **colon**, and surrounded by curly braces:

# For example

```
body {font-family: times new roman;  
color: blue;width:20pt;}
```

Or else we can also define as follows:

```
p  
{  
font-family : "sans serif"  
}  
H1{}  
H2{}
```

# Multiple Attributes in one selector

If you wish to specify more than one property, you must separate each property with a semicolon. The example below shows how to define a center aligned paragraph, with a red text color:

Example:

```
p {  
text-align:center;  
color:red;  
}
```



# Levels of Style Sheets

The CSS can be declared in 3 ways:

- 1) External style sheet
- 2) Internal style sheet (inside the <head> tag)
- 3) Inline style (inside an HTML element)

# 1)Internal Style Sheet

An internal style sheet should be used when a **single document** has a unique style. You define internal styles in the head section with the <style> tag.

## Example:

```
<head>  
<style type="text/css">  
body  
{background-color: red}  
p  
{margin-left: 20px}  
</style>  
</head>
```

## 2) Inline Styles

An inline style should be used when a unique style is to be applied to a single occurrence of an element.

To use inline styles you use the style attribute in the relevant tag. The style attribute can contain any CSS property. The example shows how to change the color and the left margin of a paragraph:

### **For Example**

```
<p style="color: red; margin-left: 20px">
```

```
    This is a paragraph</p>
```

### 3)External Style Sheet

An external style sheet is ideal, when the style is applied **to many pages**. Each page must link to the style sheet using the <link> tag. The <link> tag goes inside the head section.

```
<head>
```

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

```
</head>
```

# ***Multiple Style Sheets***

If some properties have been set for the same selector in different style sheets, the values will be inherited from the more specific style sheet. For example, an external style sheet has these properties for the h3 selector:

```
h3
{
color: red;
text-align: left;
font-size: 8pt
}
```

And an internal style sheet has these properties for the h3 selector:

```
h3
{
text-align: right;
font-size: 20pt
}
```

If the page with the internal style sheet also links to the external style sheet the properties for h3 will be:

Then the h3 properties will be as follows:

color: red;

text-align: right;

font-size: 20pt

The color is inherited from the external style sheet and the text-alignment and the font-size is replaced by the internal style sheet.

# Example on CSS

```
<html>
<head>
<style type="text/css">
body {background-color: yellow}
h1 {background-color: #00ff00}
h2 {background-color: transparent}
p {background-color: rgb(250,0,255)}
</style>
</head>
<body>
<h1> Java made simple </h1>
<h2> Introduction </h2>
  <p> Java was invented by SunMicro ... </p>
</body>
</html>
```



# ***CSS Selectors***

CSS selectors are used to "find" (or select) the HTML elements you want to style

- 1) CSS element selector
- 2) CSS id selector
- 3) CSS class selector
- 4) CSS Universal selector
- 5) CSS Generic selector

# 1) CSS Element Selector

The element selector selects HTML elements based on the element name.

***For example: If we want to align the paragraph to center and change font color to red.***

**Syntax:**

```
p
{
    text-align: center;
    color: red;
}
```

# Simple selector form

The selector is a tag name or a list of tag names, separated by commas

Consider the following examples, in which the property is font-size and the property value is a number of points :

h1, h3

{

font-size: 24pt ;

}

h2

{

font-size: 20pt ;

}

## 2. Css Id Selector

An id selector allow the application of a style to one specific element.

General form:

```
#specific-id  
{  
  property-value list  
}
```

**Example:**

```
#section14  
{  
font-size: 20;  
}
```

Specifies a font size of 20 points to the

**<h2 id = “section14” > Hello <h2>**

# 3. CSS Class Selector

Used to allow different occurrences of the same tag to use different style specifications .A style class has a name, which is attached to a tag name

p.normal {property/value list}

p.warning{property/value list}

The class you want on a particular occurrence of a tag is specified with the class attribute of the tag.

For example,

<p class = “normal”> A paragraph of text that we want to be presented in ‘normal’ presentation style </p>

<p class = “warning” > A paragraph of text that is a warning to the reader ,which should be presented in an especially noticeable style. </p>

## 4. CSS Universal Selector

The universal selector denoted by an asterisk(\*).

It applies its style to all elements in the document

For Eg:

```
*  
{  
  text-align: right;  
}
```

Makes all elements in the document align right side

## 5. CSS Generic Selector

- A generic class can be defined if you want a style to apply to more than one kind of tag. A generic class must be named, and the name must be.

Example,

```
.large {property : value list }
```

- Use it as if it were a normal style class

```
<h1 class = "large"> ... </h1>
```

...

```
<p class = "large"> ... </p> in with a period
```

# CSS Properties

CSS include 60 different properties in 7 categories:

- Fonts
- Lists
- Alignment of text
- Margins
- Colors
- Backgrounds
- Borders



# 1)Fonts

## 1.Font-Families

- The font-family property is used to specify a list of font name.
- The browser will use the first font in the list that it supports. For example, the following could be specified.

**Example:** font- family: Arial, Helvetica, Courier

## 2.font-size

Sets the size of fonts. There are two categories of font-size values, absolute and relative. In the case of absolute category the size value could be given as length value in points, picas or pixels or keywords from the list xx-small, x-small, small, medium, large and x-large.

Eg: **font-size: 10pt**

The relative size values are smaller and larger, which adjust the font size relative to the font size of the parent element.

Eg: **font-size: 1.2em**

This sets the font size to 1.2 times the font size of the parent element. 1.2em and 120% are same.

# 3.Font Variant

- The default value of the font-variant property is **normal**, which specifies the usual character font.
- This property can be set to **small-caps** to specify small capital letters.

## 4. font-style

Most commonly used to specify italic.

Eg:     font-style: italic

## 5. font-weight

- Used to specify degrees of boldness.

**Eg: font-weight: bold**

- Possible values are **bolder**, **lighter**, **bold**, **normal(default)**

## 6. Font Shorthands

*For specifying more than one font properties. The values can be stated in a list as value of the font property.*

**Eg: font: bold 14pt Arial Helvetica**

The order which browser follows is last must be font name, second last font size and then the font style, font variant and font weight can be in any order but before the font size and names.

# 7. The text-decoration property

Specifies some special features of text. Possible values of text-decoration property are : line-through, overline, underline, none

Suppose if we apply these for a paragraph tag

~~This illustrates line-through~~

This illustrates overline

This illustrates underline

## 8. List properties

- Property Name: **list-style-type** can applied to both ordered and unordered list.

### Unordered lists

- Bullet can be a disc (default), a square, or a circle. Set it on either the <ul> or <li> tag. On

<ul>, it applies to list items.

```
<style type = "text/CSS">
```

```
ul
```

```
{
```

```
list-style-type:square;
```

```
}
```

```
</style>
```

# ORDERED LIST

*On ordered lists - list-style-type property can be used to change the sequence values*

<b><i>Property value</i></b>	<b><i>Sequence type</i></b>	<b><i>First four</i></b>
Decimal	Arabic numerals	1, 2, 3, 4
upper-alpha	UC letters	A, B, C, D
lower-alpha	lc letters	a, b, c, d
upper-roman	Uc Roman	I, II, III, IV
lower-roman	Lc Roman	i, ii, iii, iv



# 9.CSS Color properties

*Color is a problem for the Web for two reasons:*

1. Monitors vary widely
2. Browsers vary widely

There are three color collections

- 1)color: color name;
- 2)color: rgb(100,255,125);
- 3)color: #01a2cf

For hexa decimal we need to enter the color range in 0 to f letters only.