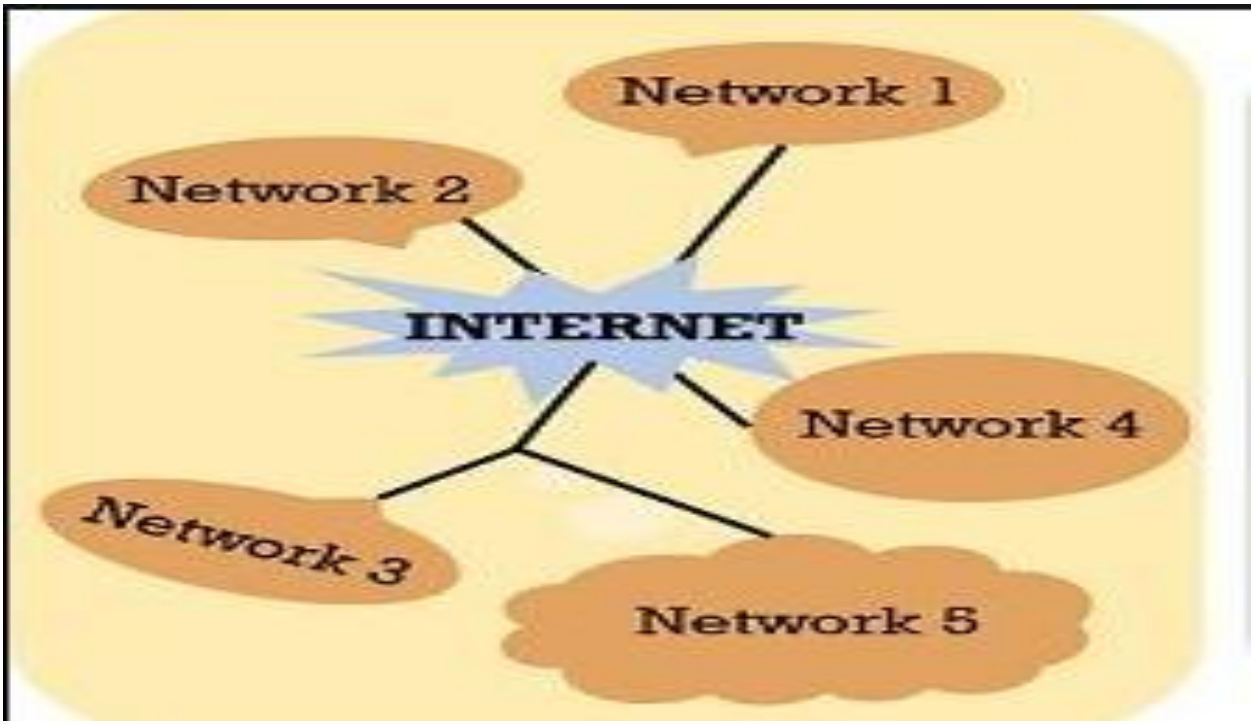
Chapter: One
HTML & CSS

Introduction to internet and Web Technology

- The internet is a global network of networks.
- It is a global system that consist of millions of public, private, academic, business and government networks.
- It connects millions of computer together globally, forming a network in which any computer can communicate with any other computer as long as they are both connected to the internet.
- The internet is a global system of interconnected Computer networks that uses the standard internet protocol suit TCP/IP to serve billions of user worldwide.



Web Technology

- Web technology is development of mechanism that allows two or more computer devices to communicate over a network.
- Web technology that relate to the interface between web servers and their client called web technology.
- It includes markup language, programming interface and language and standards for document identification and display.

- Web technology are playing the leading role in www include many latest evolutions in it like web services, web 2.0. table less design, HTML, XHTML, XML, CSS etc.
- Web technology aims to enhance creativity, secure information, sharing, collaboration and functionality of web services

- Web 0.0: The Development of the Web
- Web 1.0 – The Read-Only Web
- Web 2.0 – The Social (Read-Write) Web
- Web 3.0 – The Semantic (Read-Write-Execute) Web
- Web 4.0 – The Mobile Web?
- Web 5.0 – The Intelligent / Emotional (Symbiotic) Web

❖ **Client side web technology:**
HTML,CSS, JavaScript, VBScript, XHTML, DHTML,WML, AJAX etc.

❖ **Server Side Web technology**
ASP, PHP, perl, JSP, ASP.net, MYSQL, SQL SERVER, Access.

❖ **Some more advances Technology**
XML,XSLT, RSS,X-Path, XQuery, RDF, Ruby and Rails etc

Concept of WWW

- WWW stands for world wide web.
- WWW is a collection of information which is accessed with the help of internet.
- WWW is collection of text pages, digital photographs, music files, videos animation you can access over the internet.
- WWW is a system of internet servers that support specially formatted documents (Web page)
- These document are formatted in a markup language called HTML that supports link to other documents as well as graphics, audio and video files.
- User access the www facilities with help of web browser, which provides transparent access to the www servers
- WWW uses HTTP protocol to transmit data (Web page)

Web Terminology

Client

– Any computer on network that request services from another computer on the network.

Server

– Any computer that receives request from client computers processes and sends the response.

Web page:

– Any page that is hosted on the internet.

Web site

- Collection of interlinked web pages that is hosted on the internet.

Web Development

- The process of creating, modifying web pages.

Web browser:

- A program that receives information from the web. E.g. IE, Chrome, Mozilla etc

World Wide Web Consortium

- The World Wide Web Consortium (W3C) is the main international standards organization for the World Wide Web.
- W3C was created to ensure compatibility and agreement among industry members in the adoption of new standards.
- Prior to its creation, incompatible versions of HTML were offered by different vendors, increasing the potential for inconsistency between web pages.
- The consortium was created to get all those vendors to agree on a set of core principles and components which would be supported by everyone

Overview of HTML

- HTML stands for hypertext markup language. It is not a programming language.
- HTML is markup language that web browser used to interpret and compose text, image and other material into visual or audio and video form web page.
- HTML uses markup tags to describe web pages.
- HTML tags are keywords surrounded by angle brackets like <html> and they are case insensitive
- Most HTML tags normally come in pairs like and **and** . The first tag is called the start tag (or opening tag) and the second tag is called the end tag (or closing tag)
- The HTML file must have an extension ".htm" or ".html"

Versions of HTML

HTML 2.0

- It set the standard for core HTML features based upon current practice in 1994.

HTML 3.2

- W3C's first Recommendation for HTML which represented the consensus on HTML features for 1996.
- HTML 3.2 added widely-deployed features such as tables, applets, text-flow around images, superscripts and subscripts, while providing backwards compatibility with the existing HTML 2.0 standard.

HTML 4.0

- First released as a W3C Recommendation on 18 December 1997.

- A second release was issued on 24 April 1998 with changes limited to editorial corrections.
- This specification has now been superseded by HTML 4.01.

HTML 4.01

- HTML 4.01 is the current official standard.
- It includes support for most of the proprietary extensions, plus support for extra features (Internationalized documents, support for Cascading Style Sheets, extra TABLE, FORM, and JavaSctipt enhancements), that are not universally supported.
- After this XHTML was released which stands for eXtensible HyperText Markup Language.

HTML 5.0

- This is the new version of HTML with many exciting new features and published officially in 2012

HTML5

- HTML5 is newest version of HTML, which incorporates all feature from earlier version of HTML.
- HTML5 provides new semantic elements like <header>, <footer> and <section > etc.
- It includes enhance form controls and attribute (i.e, web Form 2.0).
- The canvas which supports two dimensional drawing directly on web page.
- HTML5 supports built-in audio and video, drag and drop functionality.
- It supports more advanced feature for web developers such as data storage and offline application.
- HTML5 should be device independent

HTML4

☞ HTML4 is the fourth version of the Hypertext Markup Language (HTML), which is the standard markup language for creating web pages and applications. It was developed by the World Wide Web Consortium (W3C) and was officially released in December 1997.

☞ HTML4 introduced several important features and improvements over HTML3.2. some important key features of HTML4:

❖ **Improved structure and semantics:** HTML4 introduced structural elements such as , , and <blockquote>, which allowed developers to create more organized and meaningful web page structures. It also included semantic markup elements like , , <cite>, and <code> for emphasizing and marking up specific parts of the content.

- **Tables and frames:** HTML4 added support for complex tables and frames, allowing developers to create more advanced layouts and structures. Tables were commonly used for tabular data presentation, while frames allowed for the division of a web page into multiple independent sections.

- **Cascading Style Sheets (CSS):** While CSS was introduced in HTML3.2, HTML4 expanded its support for CSS and encouraged its use for styling web pages. CSS allowed developers to separate the presentation (styling) from the structure and content of a web page, providing greater flexibility and control over the design.

- **Scripting:** HTML4 introduced the <script> element, which allowed developers to embed scripts within web pages. This facilitated the use of client-side scripting languages like JavaScript for interactive and dynamic web page behavior.

- **Forms and input elements:** HTML4 introduced various form-related elements and attributes, such as <form>, <input>, <select>, <textarea>, and <button>. These elements provided a standardized way to create input fields, checkboxes, radio buttons, dropdown lists, and buttons within web forms.

- **Multimedia support:** HTML4 introduced the <img> element for embedding images in web pages. It also supported plugins like Adobe Flash and Java applets for more advanced multimedia capabilities.

- **Accessibility:** HTML4 introduced accessibility features, such as the <label> element for associating labels with form controls, and the alt attribute for providing alternative text descriptions for images, which improves web accessibility for users with disabilities.

HTML5

- HTML5 is the fifth and latest version of the Hypertext Markup Language (HTML), which is the standard markup language for creating web pages and applications. It was developed by the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG), and it was officially released in October 2014.

☞ HTML5 introduced many new features over HTML4. Some of the key features of HTML5 are as follows:

- **Semantic elements:** HTML5 introduced several new semantic elements such as <header>, <footer>, <nav>, <article>, <section>, etc., which provide a more meaningful structure to web documents, making it easier for search engines and assistive technologies to understand the content.

- **Multimedia support:** HTML5 includes built-in support for audio and video elements, allowing developers to embed media content directly into web pages without the need for third-party plugins like Flash. This has greatly simplified the process of adding multimedia to websites and made it more accessible across different devices and platforms.

- **Canvas and SVG:** HTML5 introduced the <canvas> element, which provides a drawing surface for dynamic graphics and animations. Additionally, HTML5 supports Scalable Vector Graphics (SVG), which allows for the creation of vector-based graphics that can be scaled without losing quality.

- **Offline web applications:** HTML5 introduced the Application Cache (AppCache) feature, which allows web developers to create offline web applications. By specifying which files should be cached, web applications can still function even when the user is offline or experiencing a poor internet connection.
- **Improved forms and input types:** HTML5 introduced new form input types such as email, URL, date, time, number, range, etc. It also introduced form validation and the <datalist> element, which provides a list of predefined options for input fields.

- **Geolocation:** HTML5 includes a Geolocation API that allows web applications to access a user's geographic location information if the user provides permission. This feature enables location-based services and applications.

- **Web storage:** HTML5 introduced the localStorage and sessionStorage APIs, which provide a way to store data locally on the client's browser. This allows web applications to store data persistently or temporarily, depending on the needs of the application.

Difference between HTML4 and HTML5:

I. **Doctype Declaration:** In HTML4, the doctype declaration was more complex and required a Document Type Definition (DTD) to specify the version. HTML5 simplified this by introducing a simpler, shorter doctype declaration: <!DOCTYPE html>, which is compatible with all HTML5 features.

II. **Syntax:** HTML5 allows for looser syntax rules and is forgiving of minor errors, making it easier to write and maintain code. HTML4 had stricter syntax rules and required well-formed markup.

III. **Structure:** HTML4 primarily focused on defining the structure of a webpage using a limited set of elements. HTML5 introduced new semantic elements like <header>, <nav>, <section>, <article>, and <footer>, allowing developers to define the structure and meaning of content more accurately.

IV. **Multimedia Support:** HTML5 introduced built-in support for multimedia elements, such as <video> and <audio>, eliminating the need for third-party plugins like Adobe Flash Player. HTML4 relied on plugins for multimedia playback.

V. **Form Input Types:** HTML5 introduced new form input types like email, url, date, number, range, and color, providing better user experience and input validation. HTML4 had limited form input types.

VI. **Canvas and Scalable Vector Graphics (SVG):** HTML5 introduced the <canvas> element, allowing dynamic rendering of 2D graphics and animations. It also included native support for SVG, enabling the use of vector-based graphics directly in the HTML code.

VII. **Local Storage and Offline Support:** HTML5 introduced the Web Storage API, which provides a more efficient and extensive client-side storage solution compared to cookies in HTML4. HTML5 also added offline support through the use of application cache, allowing web applications to work offline and load faster.

VIII. **Mobile Support:** HTML5 includes features that improve mobile device support, such as the ability to handle touch events, geolocation APIs, and better integration with mobile browsers.

New Elements

<article>	Defines an article
<aside>	Defines content aside from the page content
<bdi>	Isolates a part of text formatted in a different direction from other text outside it
<command>	Defines a command button that a user can invoke
<details>	Defines additional details that the user can view or hide
<dialog>	Defines a dialog box or window
<summary>	Defines a visible heading for a <details> element
<figure>	Specifies self-contained content, like illustrations, diagrams, photos, code listings, etc.
<figcaption>	Defines a caption for a <figure> element
<footer>	Defines a footer for a document or section
<header>	Defines a header for a document or section
<hgroup>	Groups a set of <h1> to <h6> elements when a heading has multiple levels
<mark>	Defines marked/highlighted text
<meter>	Defines a scalar measurement within a known range (a gauge)

<nav>	Defines navigation links
<progress>	Represents the progress of a task
<ruby>	Defines a ruby annotation (for East Asian typography)
<rt>	Defines an explanation/pronunciation of characters (for East Asian typography)
<rp>	Defines what to show in browsers that do not support ruby annotations
<section>	Defines a section in a document
<time>	Defines a date/time
<wbr>	Defines a possible line-break
<audio>	Defines sound content
<video>	Defines a video or movie
<source>	Defines multiple media resources for <video> and <audio>
<embed>	Defines a container for an external application or interactive content (a plug-in)
<track>	Defines text tracks for <video> and <audio>
<canvas>	Used to draw graphics, on the fly, via scripting (usually JavaScript)

Removed Elements

- <acronym>
- <applet>
- <basefont>
- <big>
-
- <dir>
- <font>
- <frame>
- <frameset>
- <noframes>
- <strike>
- <tt>

Global Attributes

HTML attributes provide additional information about an element and define its properties. Global attributes are special types of attributes that can be used with any HTML element, offering common functionality to enhance behavior and presentation.

- ❖ Global attributes can be applied to any HTML element, such as , , <img>, and more.
- ❖ They control various aspects like styling, identification, accessibility, and interaction, improving web page design and functionality.
- ❖ Common global attributes include id, class, style, title, and lang.

1. accesskey

Defines a shortcut key to activate or focus an element.
Example: <button accesskey="s">Submit</button>

2. class

Specifies one or more class names for an element (used for styling and JavaScript).
Example:

3. contenteditable

Indicates whether the element's content is editable.
Values: true, false
Example: Edit me!

4. data-*
Custom data attributes for storing extra information.
Example:

5. dir
Specifies the text direction.
Values: ltr, rtl, auto
Example: Welcome

6. draggable
Specifies whether the element is draggable.
Values: true, false
Example: <img src="image.jpg" draggable="true">

7. hidden
Hides the element from view.
Example: <div hidden>This is hidden

8. id
Specifies a unique identifier for the element.
Example: <h1 id="main-title">Hello, World!</h1>

9. lang
Specifies the language of the element's content.
Example: This is English.

10. style
Adds inline CSS to an element.
Example: This is red text.

11. tabindex
Defines the tab order for an element.
Example: <button tabindex="1">First</button>

12. title
Provides additional information about the element (shown as a tooltip on hover).
Example: <abbr title="HyperText Markup Language">HTML</abbr>

13. translate
Specifies whether the content of the element should be translated.
Values: yes, no
Example: BrandName

HTML Document

- An HTML document is a file containing Hypertext Markup Language.
- HTML code is based on tags, or hidden keywords, which provides instruction for formatting the document.
- A tag starts with angular bracket i.e. '<' (less than sign) and end with an angular bracket i.e. '>' (greater than sign).
- Tags tell the web browser what to do with the text.
- HTML is defined by the WWC (World Wide Web Consortium), an organization that regulates standards for the Internet. It was developed by Tim Berners-Lee in 1991.
- It is not a programming language but tag oriented language. Programming on an HTML document is done with JavaScript.

Structure of an HTML document is shown below

```
<!DOCTYPE html>
<html>
    <head>
        place processing information and meta data here
    </head>
    <body>
        This is the place where the information to be displayed
    in a web page is written.
    </body>
</html>
```

HTML Elements/Tag References

- The HTML instructions, along with the text to which the instructions apply, are called HTML elements.
- The HTML instructions are themselves called tags, and look like -- that is, they are simply the element name surrounded by left and right angle brackets.
- The content in the web-page is written after the starting tag, and closed with the end tag.
- E.g**:<element_name>** text to be written HERE**</element_name>**
- The end tag has slash character in front of it.
- HTML tags are not case sensitive;<b>**means same as <B> .**

HTML Attributes

- Attributes provide additional information about HTML elements. Attributes are always specified in the start tag.
- Attributes come in name/value pair like name = "value".
- For example, HTML links are defined with <a> tag and the link address is provided as an attribute href like

Saraswati Multiple Campus.

- **Note**: Always quote attribute values and use lowercase attributes.

Organizing Text in HTML5

- HTML5 has many element or tag that allows us to organize the text of our page.
    - Paragraph tag
    - Heading
    - Line break
    - Horizontal Rule
    - List tag

Paragraphs:

- Paragraphs are defined with and tag. Think of paragraph as a block of text .
- You can use the align attribute with paragraph tag as well.

This is a paragraph
This another Paragraph

Headings:-

- Heading are defined with <h1> to <h6> tags.

- <h1> tag defines the largest heading while <h6> defines the smallest.

<h1> This is heading 1</h1>
<h2> This is heading 2</h2>
<h3> This is heading 3</h3>
<h4> This is heading 4</h4>
<h5> This is heading 5</h5>
<h6> This is heading 6</h6>

Line Breaks:

- The
 tag is used when you want to start a new line, but do not want to start a new paragraph.
- The
 tag forces a line break whenever you place it. It is single line spacing in a document.

This Code	output
This is a para graph with line breaks	This is a para graph with line breaks

Horizontal Rule The element is used for horizontal rules that act as dividers between sections like this:

The horizontal rule does not have a closing tag. It takes attributes such as align and width

Code	Output
<hr width="50%" align="center">	

//Simple.html (Example)

```
<!DOCTYPE html>
<html>
<head>
        <title>First Example</title>
</head>
<body>
        <h1>This is heading</h1>
        <p><strong>This is paragraph</strong></p><br><hr>
        <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. <mark>Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book</mark>. It has survived not only five centuries, but also the leap into electronic typesetting, remaining
```

essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.</p>
</body>
</html>

List:

- HTML offers web authors three ways for specifying lists of information. All lists must contain one or more list element.
- HTML5 provides three type of list
  - **Un ordered List**
  - **Ordered List**
  - **Definition List**

HTML Unordered Lists:

- 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>Beetroot</li>
                        <li>Ginger</li><li>Potato</li>
                        <li>Radish</li>
                </ul>
        </body>
</html>
```

HTML Ordered List:

- items are numbered list instead of bulleted, This list is created by using <ol> tag

```
<!DOCTYPE html>
<html>
        <head>
                <title>HTML Ordered List</title>
        </head>
        <body>
                <ol>
                        <li>Beetroot</li>
                        <li>Ginger</li>
                        <li>Potato</li>
                        <li>Radish</li>
                </ol>
        </body>
</html>
```

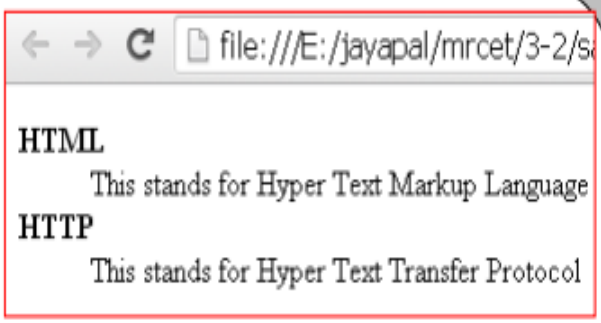
1. Beetroot
2. Ginger
3. Potato
4. Radish

HTML Definition Lists:-

- HTML and XHTML supports a list style which is called definition lists where entries are listed like in a dictionary or encyclopedia.
- The definition list is the ideal way to present a glossary, list of terms, or other name/value list.
- Definition List makes use of following three tags.

1). <dl>- Defines the start of the list
2). <dt>- A term
3). <dd>- Termdefinition
4). </dl>- Defines the end of thelist

```
<!DOCTYPE html>
<html>
  <head>
     <title>HTML Definition List</title>
  </head>
  <body>
   <dl>
        <dt><b>HTML</b></dt><dd>This stands for Hyper Text Markup Language</dd>
        <dt><b>HTTP</b></dt><dd>This stands for Hyper Text Transfer Protocol</dd>
  </dl>
</body>
</html>
```



HTML
This stands for Hyper Text Markup Language
HTTP
This stands for Hyper Text Transfer Protocol

Working with Links and URLs

- Links allow users to click their way from page to page.
- HTML links are hyperlinks.
- You can click on links and jump to another document.
- When you move mouse over a link, the mouse arrow will turn into little hand.
- In HTML, Links are defined with **<a> tag.**

*** link text ***

- Here **href** attributes specifies the destination address
- **link text** is the visible part.
- Target attribute specifies where to open linked document.
  - _blank: open linked document in new window or tab
  - _self : open the linked document in same window/tab
  - _parent: open the linked document in parent frame.
  - _top: open linked document in the fully body of window.

```
<!DOCTYPE html>
<html>
<body>
<h2>The target Attribute</h2>
<a href="https://www.w3schools.com/" target="_blank">Visit
W3Schools!</a>
<p>If target="_blank", the link will open in a new browser window or tab.</p>
</body>
</html>
```

Creating Tables

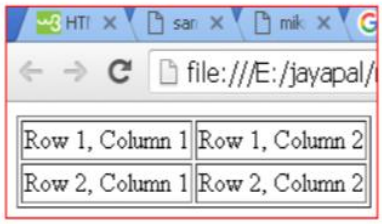
- The HTML tables allow web authors to arrange data like text, images, links, other tables, etc. into rows and columns of cells.
- . The HTML tables are created using the**<table>** tag .
- In which the **<tr>** tag is used to create table rows and <td>tag is used to create data cells.
- **<th>**elements defines the heading of table.
- **<caption>**elements defines table caption.

Attributes	**Description**
Colspan	Make a cell space may columns
Rowspan	Make a cell space may rows
Id	Uniquely define one table

Border	Defines border of table
Border-collapse	To collapse cell border
Text-align	Align cell text

Example:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Tables</title>
</head>
<body>
        <table border="1">
                <tr>
                        <td>Row 1, Column 1</td><td>Row 1, Column 2</td>
                </tr>
                <tr><td>Row 2, Column 1</td><td>Row 2, Column 2</td>
                </tr>
        </table>
</body>
</html>
```



Write HTML code to demonstrate following out put?

BCA		Student Information			
		2077	2078	2079	2080
Category	Boys	95	105	200	250
	Girls	75	85	150	225

Code:

```
 <tableborder="1px"width="50%"align="center">
    <tr><thcolspan="2"rowspan="2">BCA</th><thcolspan="4">Student
Information</th></tr>
    <tr><td>2077</td><td>2078</td><td>2079</td><td>2080</td></tr>

<tr><throwspan="2">Category</th><th>Boys</th><td>95</td><td>105</td><td>200</td><
td>250</td></tr>
    <tr><th>Girls</th><td>75</td><td>85</td><td>150</td><td>225</td></tr>

  </table>
```

Write HTML tag to generate the following table.

Routine			
Sunday	**Monday**	**Break**	**Tuesday**
WT	DSA		SAD
Java	Statics		Maths

Code:

```
<h3>Routine</h3>
  <tablewidth="50%"align="center"border="1"cellspacing="0">
 <tr><thcolspan="4">Routine</th></tr>
 <tr><th>Sunday</th><th>Monday</th><throwspan="4">Break</th><th>Tuesday</th></tr>
    <tr><td>WT</td><td>DSA</td><td>SAD</td></tr>
    <tr><td>Java</td><td>Statics</td><td>Maths</td></tr>
    <tr><td> </td><td> </td><td> </td></tr>
  </table>
```

Working with Image, Color and Canvas

- There are several image format suitable for web uses like jpeg, png and GIF etc.
- An image is inserted with <img/> tag.
- It must include at least two attributes: **src**(It is a necessary attribute that describes the source or path of the image.) and **alt(T**he alt attribute defines an alternate text for the image, if it can't be displayed**).**
- If image illustrates the text,it is best place it inside a <figure> tag. The <figurecaption> tag is used to enter the image caption.

Example

```
<!DOCTYPE html>
<html>
    <head>
        <title>Using Image in Webpage</title>
    </head>
    <body><p>Simple Image Insert</p>
        <img src="test.png" alt="Test Image" />
    </body>
</html>
```

HTML Colors

- HTML colors are specified with predefined color names, or with RGB, HEX, HSL, RGBA, or HSLA values.
- HTML supports 140 standards color names.

HTML RGB and RGBA Colors

- An RGB color value represents RED, GREEN, and BLUE light sources.
- An RGBA color value is an extension of RGB with an Alpha channel (opacity).
- In HTML, a color can be specified as an RGB value, using this formula: **rgb(*red, green, blue*)**
- Each parameter (red, green, and blue) defines the intensity of the color with a value between 0 and 255.
  Example*: **<h1 style="background-color:rgb(255, 0, 0);">rgb(255, 0, 0)</h1>***
- RGBA color values are an extension of RGB color values with an Alpha channel - which specifies the opacity for a color.
- An RGBA color value is specified with:
  rgba(*red, green, blue, alpha*)
- The alpha parameter is a number between 0.0 (fully transparent) and 1.0 (not transparent at all):
  Example**: *<h1 style="background-color:rgba(255, 99, 71, 0);">rgba(255, 99, 71, 0)</h1>***

HTML HEX Colors

- A hexadecimal color is specified with: #RRGGBB, where the RR (red), GG (green) and BB (blue) hexadecimal integers specify the components of the color.
- In HTML, a color can be specified using a hexadecimal value in the form:
- ***#rrggbb***
- Where rr (red), gg (green) and bb (blue) are hexadecimal values between 00 and ff (same as decimal 0-255).
  Example: <h1 style="background-color:#ff0000;">#ff0000</h1>

HTML HSL and HSLA Colors

- HSL stands for hue, saturation, and lightness.
- HSLA color values are an extension of HSL with an Alpha channel (opacity).
- In HTML, a color can be specified using hue, saturation, and lightness (HSL) in the form:
  hsl(*hue*, *saturation*, *lightness*)
- Hue is a degree on the color wheel from 0 to 360. 0 is red, 120 is green, and 240 is blue.
- Saturation is a percentage value, 0% means a shade of gray, and 100% is the full color.
- Lightness is also a percentage value, 0% is black, and 100% is white.
  Example: <h1 style="background-color:hsl(0, 100%, 50%);">hsl(0, 100%, 50%)</h1>
- An HSLA color value is specified with:
  hsla(*hue, saturation*, *lightness, alpha*)
- The alpha parameter is a number between 0.0 (fully transparent) and 1.0 (not transparent at all):
  Example: <h1 style="background-color:hsla(9, 100%, 64%, 0);">hsla(9, 100%, 64%, 0)</h1>

Creating Hyperlink

- ☞ Hyperlink means it create a link. Whenever you click on the link, then you will reach where it is linked.
- ☞ Generally there are two types of link:
  1. **Internal link**
  2. **External link**
- ☞ You can use any types of text, image, symbols or any part of an image for hyperlink.

- ☞ Internal links only work inside the single page and external links connect more than one individual page.

The <A NAME> Tag:

- ☞ This tag is used to define marking for hyperlinks within a single page.
- ☞ We can define any text within double quote ("....") for book marking.

Note: Don't forget to define bookmark with hash (#) key for hyperlinks.

The <A HREF> Tag:

- ☞ This tag (Anchor Hyperlink REFerence) is used to create required internal and external hyperlinks. In case of internal link, you have to define Anchor Name at first.

Uses of LINK, ALINK, VLINK attribute of <BODY> tag

LINK:

- ☞ This attribute of <BODY> tag defines the color of Hyperlink, which have not yet been visited.
  E.g. <BODY link="NAVY">

ALINK:

- ☞ This attribute of <BODY> tag defines the color of Hyperlink, when you click on it.
  E.g. <BODY ALINK="RED">

VLINK:

- ☞ This attribute of <BODY> tag defines the color of Hyperlink, which you have already visited.
  E.g. <BODY VLINK="CYAN">

Email And Web Link

Internal link Using an Image

Direct linking with an image

Inserting Special Character

- ☞ Apart from all alphanumeric and numeric characters, you can add special characters and symbols.

Description	Entry Number	Result
Non-breaking space		^s

Less than	&#60	<
Greater than	&#62	>
Ampersand	&#38	&
Cent	&#162	C
Pound	&#163	£
Yen	&#165	¥
Euro	&#8364	€
Copyright	&#169	©
Registered trademark	&#174	®
Trademark	&#153	™
	&#177	+ -

How to creating a Form

- ☞ A form is an area that can contain form elements.
- ☞ When you create a form, you must set up a form before adding in run to the form.
- ☞ To setup a form, you need to specify two important information: ACTION, METHOD.

METHOD:

- ☞ Method property tells the form how to transfer the data to the form processor.
- ☞ The value for method can be "POST" or "GET". POST means that you are going to post the information and GET means that you are just going to get another page.

  Note: GET is almost never used.

ACTION:

- ☞ Action property tells what action the form should take when the user precess the SUBMIT Button.
- ☞ Action is the URL (Uniform Resource Locator) to which you are posting the information.
- ☞ This normally has a CGI, ASP, Mailto: etc.

The <INPUT> Tag:

- ☞ It is the most used form tag. The type of INPUT is specified with the TYPE attribute.
- ☞ This tag doesn't have its ending tag.
- ☞ The most commonly used INPUT TYPES are as below:

- TEXT(NAME, VALUE, SIZE, MAXLENGTH,PLACEHOLDER)
- RADIO(NAME, VALUE, CHECKED)
- CHECKBOX(Name, Value)
- IMAGE(SRC)
- RESET(Value)
- SUBMIT(value)
- PASSWORD/HIDDEN(Name, Value, Size, Maxlength)

Where NAME specify the name of input tag and value specify that its value.

<TEXTAREA> Tag:

- This tag is used to specify the area of the text. This tag makes two dimensional text area, in which the viewer can type from a short sentence to many paragraph. This tag has its ending tag.
- This tag has following attributes:
  - NAME(Name of the tag)
  - VALUE(Editable information of the tag)
  - COLS(width of the textarea)
  - ROWS(Height of the textarea)

The <SELECT> Tag

- This tag is used to create a menu that offers visitors a list of option to choose from.
- It has ending tag called </SELECT>.
- It has following attributes and values:
  - NAME (It specify name of the tag)
  - MULTIPLE(If more than one option shown at a time)
  - SIZE(Number options you want to display at a time)

The <OPTION> Tag

- This is the mini-tag of <SELECT> tag. Normally, it comes with <SELECT> tag. This tag is used to define the options to choose from the dropdown list. This tag mainly has one attribute:
  - SELECTED
- This attribute specifies the default option that appears first in the non-multiple SELECT tag. It is not really needed since the first <OPTION> tag is assumed to be default.

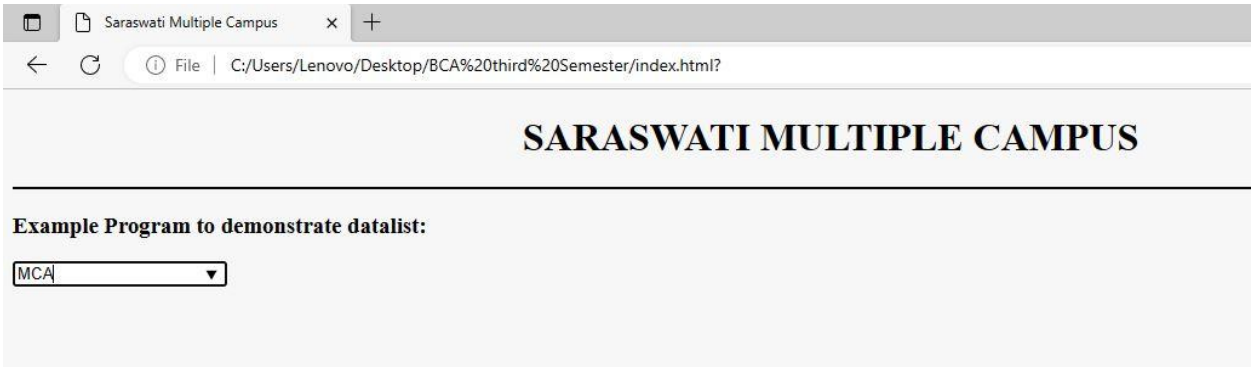
How to use <datalist> Tag:

```
<!DOCTYPEhtml>
<html>
    <head><title>Saraswati Multiple Campus</title></head>
    <bodybgcolor="#f7f7f7">
        <center><h1>SARASWATI MULTIPLE CAMPUS</h1></center>
```

```
        <hrcolor="#060807">
<h3>Example Program to demonstrate datalist:</h3>

  <FORMaction=""method="">
    <inputlist="course">
    <datalistid="course">
<optionvalue="BBA">
  <optionvalue="BBS">
    <optionvalue="BCA">
      <optionvalue="MCA">
    </datalist>
  </FORM>
    </body>
</html>
```

Output:



How to Using Frames

- ☞ Generally, a browser will display a single HTML document in its window. Using the frames extensions to HTML, you can divide the main browser window into a number of sub windows, which is referred to as "Frame".
- ☞ Each frame contains a different HTML documents and can be linked to other frames.
- ☞ A framed document is like any other HTML document but it has a <FRAMESET> tag instead of <BODY> tag.

The <FRAMESET> Tag

- ☞ This is the main container of a framed document. We use only <FRAME> tag and <NOFRAME> tag between the <FRAMESET> tags.
- ☞ This tag has only two types of attributes:
  - ❖ Rows
  - ❖ Cols

Rows:

- ☞ The values of the ROWS attribute determine how the screen is to be divided up between different frames.
- ☞ There are three ways to express the value:
  <FRAMESET ROWS="250,*">
  Where, 250 pixel is the height of the first row and the other take what is left out.
  <FRAMESET ROWS="25%,75%"> OR <FRAMESET ROWS="25%,*">

Cols:

- ☞ This attribute has the same syntax as the rows attribute, but divides the screen up horizontally.
- ☞ Frameset tag can have either the rows or the cols attribute but not both. There can be nested framesets (combination of rows and cols attributes).

The <FRAME> Tag:

- ☞ This tag describes the individual frames within a <FRAMESET> tag.
- ☞ It is not a container, so there is no its ending tag.
- ☞ The <Frame> tag has following attributes and values:

SRC:

- ☞ This attributes specifies the source of the HTML file to place in the same frame. If you have no SRC attribute, then the browser will display the blank frame.

NAME:

- ☞ This attribute is used to give a frame name, so that it can be addressed by links in other documents.

SCROLLING:

- ☞ This attribute determines whether the frame should have scrollbars or not.
- ☞ This attribute has the following values:
  Scrolling = Yes
  Scrolling = No
  Scrolling = Auto (By default, but it holds vertical scrollbar)

NORESIZE:

- ☞ Normally, the user can change the sizes of individual frames by dragging them with the mouse, if NORESIZE attribute is used then the user can't change the sizes of the frames.

MARGINWITDTH:

- ☞ This attribute specifies the horizontal distance in pixel between the contents and left/right edges of the frame.

MARGINHEIGHT:

- ☞ This attribute specifies the vertical distance in pixels between the contents and top/bottom edges of the frame.

TARGET:

- ☞ This attributes specifies the target location of the URL of the HTML files.
  Eg. <FRAME NAME="contents" TARGET="main" SRC="content.html">

Banner	
Contents Hyperlink Here	Main Target of contents

The <NOFRAME>

- ☞ Some browser cannot handle or support frames. So, this tag is used to display message to inform use about it.
- ☞ It ends with </NOFRAME> tag. A browser that can handle frames will ignore anything contained inside these tags.

Cascade Style Sheet (CSS)

What is CSS?

CSS stands for Cascade Style Sheet is a simple design language intended to simplify the process of making web page presentable. CSS handles the look and feel part of web page. Using CSS, you can control the color of text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used as well as variety of other effects. CSS is easy to learn and understand but it provides a powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.

Advantage of CSS

1. **CSS save time**

You can write CSS once and then reuse the same sheet in multiple HTML pages. You can define a style sheet for each HTML element and apply it to as many web pages as you want.

2. **Page load faster**
   If you are using CSS, you do not need to write HTML tag attributes every time just write one CSS rules of a tag and apply it to all the occurrences of that tag. So, less code means faster download times.

3. **Easy maintenance**
   To make a global change, simply change the style, and all the elements in all the web pages will be updated automatically.
4. **Superior styles to HTML**
   CSS has a much wider array of attributes than HTML, So you can give a far better look to your HTML page in comparison to HTML attributes.
5. **Multiple Devices Compatibility**
   Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cellphones or for printing.
6. **Global web standards**
   Now HTML attributes are being deprecated and it is being recommended to use CSS. So, it'sa good idea to start using CSS in all the HTML pages to make them compatible with future browsers.

CSS Syntax

A CSS comprises of style rules that are interpreted by the browser and then applied to the corresponding elements in your document. A style rule is made of three parts:

i. **Selector**

A selector is an HTML tag at which a style will be applied. This could be any tag like <H1> or<TABLE> etc.

ii. **Property**
A property is a type of attribute of HTML tag. Put simply, all the HTML attributes are converted into CSS properties. They could be color, border, etc.

iii. **Value**
Values are assigned to properties. For example, color property can have the value either red or #F1F1F1 etc.

Syntax:

Selector { property: value;}

Eg. you can define a table border as follws

Table{border:1px solid #C00;}

Where , table is a selector and border is a property and the given value 1px solid #C00 is the value of that property.

Type of Selector

you can define selectors in various simple ways base on your comfort.

1. **The Type Selectors**

The type selector is used to apply the specify type of HTML element or tag.

Eg

H1 {

Color:#CCC;

}

2. **The Universal Selector**
   Rather than selecting elements of a specific type, the universal selector quite simply matches the name of any element type. It is denoted by '*' operator.
   Syntax:
   - * { propeeryt:values;}

   Eg. *

   {

   margin:0px;

   padding:0px;

   }

Note: This rule renders the content of every element in our document margin and padding zero.

The Class Selector

You can define style rules based on the class attribute of the elements. All the elements having that class will be formatted according to the defined rule. It is denoted by '. 'operator.

Syntax

. clsaa_nam { property:values;}

Eg. black { color:#000000;}

This class can also use in particular element

Eg. h1.black

{ color:#000000;}

Note: This rule renders the content in black for only <h1> elements with class attribute set to black.

The ID Selectors

You can define style rules based on the id attribute of the elements. All the elements having that id will be formatted according to the defined rule.

Syntax:

#id_name

{

property:value;

}

Eg. #black

{

Color:#000000;

}

This rule renders the content in black for every element with id attribute set to black in our document. You can also use id as well

H1#black

{

Color:#000000;

}

The Child Selector

Eg

Body>p{

color:#000000;

}

The attribute selector

You can also apply styles to HTML elements with particular attributes. The style rule below will match all the input elements having a type attribute with a value of text.

Eg.

```
Input[type=”text”]
{
        Color:#000000;
}
```

Multiple Style Rules

You may need to define multiple style rules for a single element. You can define these rules to combine multiple properties and corresponding values into a single block as defined in the following example

```
H1{
color:#36C;
font-weight:normal;
letter-spacing: .4em;
margin-bottom:1em;
text-transform:lowercase;
}
```

Grouping Selectors

You can apply a style to many selectors if you like. Just separate the selectors with a comma, as given in the following example

Eg.

```
h1,h2,h3,p,div
{
color:#36C;
font-weight: normal;
letter-spacing: .4em;
margin-bottom: 1em;
```

text-transform: lowercase;

}

Note:

You can combine the various class selectors together as shown below:

Eg.

#content, #footer, #supplement {

Position: absolute;

Left:510px;

Width:200px;

}

How to use CSS (CSS - Inclusion)

There are four ways to associate style with HTML document. Most commonly used methods are inline CSS and External CSS.

1. Internal CSS (Embedded CSS)
2. Inline CSS
3. External CSS

Internal CSS (Embedded CSS)

Internal CSS can be use into an HTML document using the <style> element. This tag is placed inside the <head>……</head> tags.

Syntax:

<head>

<style type="text/css" media="…">

Style Rules

……

</style>

</head>

Attributes & Values of Style element

Attributes	Values	Description
Type	Text/css	Specifies the style sheet language as a content-type. This is required attribute.
Media	Screen	Specifies the devise, the document will be displayed on. Default value is all. This is an optional attribute.
	Tty	
	Tv	
	Projection	
	Handheld	
	Print	
	Braille	
	Aural	
	all	

Inline CSS

Inline style rule is only use with HTML element. This rule will be applied to that element only

Syntax:

<element style=”properties and values;”

Eg. <h1 style=”color:#36c;”>This is inline CSS</h1>

External CSS

The< link > element is used to include an external style sheet . It has following attributes and values

Rel=”stylesheet”

Type=”txt/css”

Href=”stylesheet.css”

Note: It is always included between <head></head> tag.

Eg

<head>

<link rel=”stylesheet” type=”txt/css” href=”stylesheet.css”>

</head>