

Part-II  
Web-Technology (Optional)

Sastri/ B.A 2<sup>nd</sup> Year  
Course/Paper.2

**INTRODUCTION TO HTML  
&  
BASICS OF INTERNET TECHNOLOGY**



**CENTER OF DISTANCE & ONLINE EDUCATION**

*(Formerly Directorate of Distance Education)*

**NATIONAL SANSKRIT UNIVERSITY :: TIRUPATI-517 507 (A.P)**

*(Erstwhile Rashtriya Sanskrit Vidyapeetha, Tirupati)*



# **Web Technology (Optional)**

## **1.0. Introduction**

### **1.1. Objective**

### **1.2. Basic HTML**

#### **1.2.1. HTML**

#### **1.2.2. Advantages of HTML**

#### **1.2.3. Disadvantages of HTML**

### **1.3. Structure of HTML**

### **1.4. Working with Text**

### **1.5. Summary**

### **1.6. SAQ**

---

## **1.0 Introduction**

HTML stands for Hyper Text Markup Language. It is used to design web pages using markup language. HTML is the combination of Hypertext and Markup language. ... HTML was created by Tim Berners-Lee in 1991. The first ever version of HTML was HTML 1.0, but the first standard version was HTML 2.0, published in 1999. The latest version of HTML is **HTML 5**, which has changed the face of web page design.

### **1.1 Objective**

In this chapter we discuss all basic tags of HTML

### **1.2. Basic HTML**

HTML is the standard Markup language for creating Web pages.

- HTML stands for Hyper Text Markup Language
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML elements are represented by tags
- HTML tags label pieces of content such as "heading", "paragraph", "table", and so on
- Browsers do not display the HTML tags, but use them to render the content of the page.

### 1.2.1. HTML

HTML (Hypertext Markup Language) is a markup language, which consists of tags embedded in the text of a document.

- **Hyper** is the opposite of linear. It used to be that computer programs had to move in a linear fashion. This before this, this before this, and so on. HTML does not hold to that pattern and allows the person viewing the World Wide Web page to go anywhere, any time they want.
- **Text** is what you will use. Real, honest to goodness English letters.
- **Mark up** is what you will do. You will write in plain English and then mark up what you wrote. More to come on that in the next Primer.
- **Language** because they needed something that started with "L" to finish HTML and Hypertext Markup Louie didn't flow correctly. Because it's a language, really -- but the language is plain English

The browser reading the document interprets these markup tags to help format the document for subsequent display to a reader. However, the browser makes many of the decisions about layout. Remember, web browsers are available for a wide variety of computer systems.

The browser thus displays the document with regard to features that the viewer selects either explicitly or implicitly. Factors affecting the layout and presentation include:

- The markup tags used.
- The physical page width.
- The fonts used to display the text.
- The color depth of the display.

#### **Basic concepts of HTML**

The HTML document on the word processor, or Notepad, WordPad, or Simple Text. When you are finished creating the HTML document, you'll then open the document in a browser, like Netscape Navigator. The browser will interpret the HTML commands for you and display the Web page. HTML documents must be text only. When you save an HTML document, you must save only the text.

## **The Word Processor**

When you write to the word processor you will need to follow a few steps:

1. Write the page as you would any other document.
2. When you go to save the document (Here's the trick), ALWAYS choose SAVE AS.
3. When the SAVE AS box pops up, you will need to save the page in a specific format. Look at the SAVE AS dialogue box when it pops up: Usually at the bottom, you find where you will be able to change the file format.
4. If you have a PC, save your document as ASCII TEXT DOS or just TEXT. Either one will work.
5. If you have a MAC, save your document as TEXT.
6. When I started writing HTML, I saved pages by assigning every Web page its own floppy disc. It just helped me keep it all straight, but if you want to save right to your hard drive, do it. I only offer the floppy disc premise as a suggestion.

**Note:** It is very important to choose SAVE AS EVERY time you save your document. If you don't, the program won't save as TEXT, but rather in its default format. In layman's terms -- use SAVE AS or screw up your document.

## **Name Your Document**

What you name your document is very important. You must first give your document a name and then add a suffix to it. That's the way everything works in HTML. You give a name and then a suffix.

Follow this format to name your document:

1. Choose a name. Anything. If you have a PC not running Windows 95, you are limited to eight letters, however.
2. Add a suffix. For all HTML documents, you will add either ".htm" or ".html".

## **Running a web page**

Once you have your HTML document on the floppy disc or your hard drive, you'll need to open it up in the browser. It's easy enough. Since you're using a browser to look at this Primer, follow along.

1. Under the FILE menu at the very top left of this screen, you'll find OPEN, OPEN FILE, OPEN DOCUMENT, or words to that effect.
2. Click on it. Some browsers give you the dialogue box that allows you to find your document right away. Internet Explorer, and later versions of Netscape Navigator, requires you to click on a BROWSE button or OPEN FILE button to get the dialogue box. When the dialogue box opens up, switch to the A:\ drive (or the floppy disc for

MAC users) and open your document. If you saved the file to your hard drive, get it from there.

3. You might have to then click an OK button. The browser will do the rest.

### **1.2.2. Advantages of HTML**

- HTML is widely used.
- Every browser supports HTML Language.
- Easy to learn and use.
- HTML is light weighted and fast to load.
- Do not get to purchase any extra software because it's by default in every window.
- Easy to use
- Loose syntax (although, being too flexible won't suit standards).
- HTML is easy enough to write
- HTML is that it is easy to code even for novice programmers.
- HTML also allows the utilization of templates, which makes designing a webpage easy.
- Very useful for beginners in the web designing field.
- HTML can be supported to each and every browser, if not supported to all the browsers.
- HTML is built on almost every website, if not all websites.
- HTML is increasingly used for data storage as like XML syntax.
- Free – You need not buy any software.
- HTML is present in every window by default so you not need to buy the software which cost too much.
- HTML has many tag and attributes which can short your line of code.

### **1.2.3. Disadvantages of HTML**

- It cannot produce dynamic output alone, since it's a static language.
- Making the structure of HTML documents becomes tough to understand.
- Errors can be costly.
- It is the time consuming as the time it consume to maintain on the colour scheme of a page and to make lists, tables and forms.

- It can create only static and plain pages so if we'd like dynamic pages then HTML isn't useful.
- Required to write a lot of code for just creating a simple webpage.
- We have to check up the deprecated tags and confirm not to use them to appear because another language that works with HTML has replaced the first work of the tag, and hence the opposite language needs to be understood and learned.
- Security features offered by HTML are limited.
- If we need to write down long code for creating a webpage then it produces some complexity.
- HTML can create only static and plain pages so if we'd like dynamic pages then HTML isn't useful.
- I need to write down tons of code for creating an easy webpage.
- Security features are not good at HTML.
- If we'd like to write down long code for creating a webpage then it produces some complexity.

### 1.3. Structure of HTML

#### HTML Markup Tags

HTML markup tags are usually called HTML tags

- HTML tags are keywords 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

Note: The start and end tags are also called the opening and closing tags

#### Syntax and example

```
<html>
<body>
<h1>My First HTML header</h1>
<p>My first HTML paragraph</p>
</body>
</html>
```



When a browser displays a web page, it will not display the markup tags.

The text between the <html> and </html> tags describes a web page.

The text between the <body> and </body> tags is displayed in the web browser.

The text between the <p> and </p> tags is displayed as paragraphs.

The text between the <b> and </b> tags is displayed in a bold font.

#### 1.4. Working with text

**HTML Formatting** is a process of formatting text for better look and feel. HTML provides us ability to format text without using CSS. There are many formatting tags in HTML. These tags are used to make text bold, italicized, or underlined. There are almost 14 options available that how text appears in HTML and XHTML

In HTML the formatting tags are divided into two categories:

- Physical tag: These tags are used to provide the visual appearance to the text.
- Logical tag: These tags are used to add some logical or semantic value to the text.

Element name	Description
<b>	This is a physical tag, which is used to bold the text written between it.
<strong>	This is a logical tag, which tells the browser that the text is important.
<i>	This is a physical tag which is used to make text italic.
<em>	This is a logical tag which is used to display content in italic.
<mark>	This tag is used to highlight text.
<u>	This tag is used to underline text written between it.
<tt>	This tag is used to appear a text in teletype. (not supported in HTML5)
<strike>	This tag is used to draw a strikethrough on a section of text. (Not supported in HTML5)

<sup>	It displays the content slightly above the normal line.
<sub>	It displays the content slightly below the normal line.
<del>	This tag is used to display the deleted content.
<ins>	This tag displays the content which is added
<big>	This tag is used to increase the font size by one conventional unit.
<small>	This tag is used to decrease the font size by one unit from base font size.

### Example

```
<!DOCTYPE html>
<html>
<head>
  <title>formatting elements</title>
</head>
<body>
<h1>Explanation of formatting element</h1>
<p><strong>This is an important content</strong>, and this is normal content</p>
</body>
</html>
```

### SAQ

1. Write a short note on basic tags of HTML with examples?
2. List different tags used for text?

### 1.5. Summary

- **HTML** stands for Hyper Text Markup Language.
- It is used to design web pages using markup language. ... **HTML** is a markup language which is used by the browser to manipulate text, images and other content to display it in required format. **HTML** was created by Tim Berners-Lee in 1991

- **HTML Formatting** is a process of formatting text for better look and feel. HTML provides us ability to format text without using CSS.

### **1.6. SAQ**

1. What is HTML? Explain various features of HTML.
2. List the disadvantages of HTML?
3. Explain the basic structure of HTML with example.
4. Discuss about various elements for text formatting.

## **2.0. Objective**

### **2.1. Introduction**

### **2.2. Heading tag**

**2.2.1. Font tag, Address tag, Line break, Paragraph tag, Center tag, Preformatted tag, Marquee tag**

### **2.3. Formatting block text**

**2.3.1. Div tag**

**2.3.2. Span tag**

### **2.4. List tag**

**2.4.1. Unordered list**

**2.4.2. Ordered list**

### **2.5. Summary**

### **2.6. SAQ**

---

## **2.0. Objective**

In this chapter we discuss heading tags, formatting tags and list tags

### **2.1. Introduction**

Headings in HTML helps the search engine to understand and index the structure of web page.

### **2.2. Heading tag**

A HTML heading or HTML h tag can be defined as a title or a subtitle which you want to display on the webpage. When you place the text within the heading tags `<h1>.....</h1>`, it is displayed on the browser in the bold format and size of the text depends on the number of heading.

There are six different HTML headings which are defined with the `<h1>` to `<h6>` tags, from highest level h1 (main heading) to the least level h6 (least important heading).

h1 is the largest heading tag and h6 is the smallest one. So h1 is used for most important heading and h6 is used for least important.

Headings are defined with the `<h1>` to `<h6>` tags. `<h1>` defines the largest heading. `<h6>` defines the smallest heading.

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

`<h2>This is a heading</h2>`

`<h3>This is a heading</h3>`

## Out put

This is a heading  
This is a heading  
This is a heading

## Example

```
<html>  
  
<body>  
  
<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>  
  
<p>Use heading tags only for headings.  
  
Don't use them to make something <b>BIG or BOLD</b>.  
  
Use other tags for that. </p>  
  
</body>  
  
</html>
```

### 2.2.1. Font tag, Address tag, Line break, Paragraph tag, Center tag, Preformatted tag, Marquee tag

#### Font tag

The **<font>** tag in HTML plays an important role in the web page to create an attractive and readable web page. The font tag is used to change the color, size, and style of a text. The base font tag is used to set all the text to the same size, color and face.

### **Syntax:**

`< font attribute = “value”> content </font>`

### **Address tag**

The `<address>` tag in HTML indicates the contact information of a person or an organization. If `<address>` tag is used inside the `<body>` tag then it represents the contact information of the document and if the `<address>` tag is used inside the `<article>` tag, then it represents the contact information of the article. The text inside the `<address>` tag will display in italic format. Some browsers add a line break before and after the address element.

### **Syntax**

`<address> Address....</address>`

### **Line Breaks**

Use the `<br>` tag if you want a line break (a new line) without starting a new paragraph:

`<p>This is<br>a para<br>graph with line breaks</p>`

The `<br>` tag is an empty tag. It has no end tag like `</br>`.

You can read more about empty HTML tags in the next chapter of this tutorial.

### **Syntax**

`<br>` or `<br />`

Even if `<br>` works in all browsers, writing `<br />` instead is more future proof.

### **Example**

`<html>`

`<body>`

`<p>This is<br>a para<br>graph with line breaks</p>`

`</body>`

`</html>`

### **Output**

This is  
a para  
graph with line breaks

## Paragraph tags

In this article, we will know the **HTML Paragraphs**, & its basic implementation through the examples. The `<p>` tag in HTML defines a paragraph. These have both opening and closing tags. So anything mentioned within `<p>` and `</p>` is treated as a paragraph. Most browsers read a line as a paragraph even if we don't use the closing tag i.e, `</p>`, but this may raise unexpected results. So, it is both a good convention, and we **must** use the closing tag.

## Syntax

`<p> content </p>`

## Comments

Comments can be inserted in the HTML code to make it more readable and understandable. Comments are ignored by the browser and not displayed.

## Syntax

`<!-- This is a comment -->`

Note: There is an exclamation point after the opening bracket, but not before the closing bracket.

## Example

```
<html>
```

```
<body>
```

```
<!--This comment will not be displayed-->
```

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

```
</body>
```

```
</html>
```

## Horizontal tag

The <hr> tag inserts a horizontal rule. In HTML the <hr> tag has no end tag.

### Syntax

```
<hr>
```

### Example

```
<html>
```

```
<body>
```

```
<p>The hr element defines a horizontal rule:</p>
```

```
<hr />
```

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

```
<hr />
```

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

```
<hr />
```

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

```
</body>
```

```
</html>
```

### Output

The hr tag defines a horizontal rule:

---

This is a paragraph

---

This is a paragraph

---

This is a paragraph



## Center tag

The `<center>` tag in HTML is used to set the alignment of text into the center. This tag is not supported in HTML5. CSS's property is used to set the alignment of the element instead of the center tag in HTML5

### Syntax

```
<center> content....</center>
```

## Preformatted tag

The `<pre>` tag in HTML is used to define the **block of preformatted text** which preserves the text spaces, line breaks, tabs, and other formatting characters which are ignored by web browsers. Text in the `<pre>` element is displayed in a fixed-width font, but it can be changed using CSS. The `<pre>` tag requires a starting and end tag.

### Syntax

```
<pre> contents... </pre>
```

## Marquee tag

The `<center>` tag is used to center-align text.

## Browser Support

Element	Chrome	Internet explore	Firefor	opera
<code>&lt;center&gt;</code>	Yes	Yes	Yes	Yes

## Marquee tag:

The `<marquee>` is a non-standard HTML tag which was used to create a scrolling text or an image. It was used to make the text/image scroll horizontally across or vertically down the web page. The `<marquee>` element comes in pairs. It means that the tag has

opening `<marquee>` and closing `</marquee>` elements. Consider the following example for marquee tag

```
<! DOCTYPE HTML>
<html>
<head>
<title> title of the document </title>
</head>
<body>
<marquee> a scrolling text created with HTML marquee element.
</marquee>
</body>
</html>
```

Use direction attribute of `<marquee>` element to change the direction of the text/image. See another example where the text scrolls from up to down.

```
<! DOCTYPE HTML>
<html>
<head>
<title> title of the document </title>
</head>
<body>
<marquee direction="down"> a scrolling text created with HTML marquee element.
</marquee>
</body>
</html>
```

### **Let us do one more example**

```
<! DOCTYPE HTML>
<html>
<head>
<title> title of the document </title>
</head>
<body>
<marquee behavior="scroll" direction="up">

</marquee>
</body>
</html>
```

### **Attributes**

The following attributes can be used to adjust the appearance of the `<marquee>` element.

Attribute	Value	Description
behavior	scroll slide alternate	Defines the scrolling type.
bgcolor	rgb(x,x,x) #xxxxxx colorname	Is used to give a background color.
direction	up down left right	Sets the direction for the scrolling content.
Height	pixels %	Defines the marquee's height.
Hspace	Pixels	Defines horizontal space around the marquee.
Loop	Number	Defines how many times the content will scroll. If we don't define this, the content will scroll forever.
scrollamount	Number	Defines the scrolling amount at each interval in pixels. Default value is 6.
scrolldelay	Seconds	Defines how long delay will be between each jump. The default value is 85 and smaller amounts than 60 will be ignored.
truespeed	Seconds	Is used to delay the scroll lesser than 60.

Attribute	Value	Description
Vspace	Pixels	Defines vertical space around the marquee.
Width	pixels %	Defines the marquee's width.

### 2.3. Formatting block text

Every HTML element has a default display value, depending on what type of element it is. There are two display values: **block and inline**. Two commonly used block elements are: **<div> and <span>**.

Span and div are both **generic HTML elements that group together related parts of a web page**. However, they serve different functions. A div element is used for block-level organization and styling of page elements, whereas a span element is used for inline organization and styling. Let us see them in detail

#### 2.3.1. Div tag

This is the very important block level tag which plays a big role in grouping various other HTML tags and applying CSS on group of elements. Even now **<div>** tag can be used to create webpage layout where we define different parts (Left, Right, Top etc.) of the page using **<div>** tag. This tag does not provide any visual change on the block but this has more meaning when it is used with CSS.

The div tag is known as Division tag. The div tag is used in HTML to make divisions of content in the web page like (text, images, header, footer, navigation bar, etc). Div tag has both open(**<div>**) and closing (**</div>**) tag and it is mandatory to close the tag. The Div is the most usable tag in web development because it helps us to separate out data in the web page and we can create a particular section for particular data or function in the web pages.

- Div tag is Block level tag
- It is a generic container tag

- It is used to the group of various tags of HTML so that sections can be created and style can be applied to them.

Following example for <div> using CSS(Cascading Style Sheets which we discuss later)

**Example**

```
<html>
<head>
  <title>gfg</title>
<style type=text/css>

p{
background-color:gray;
margin: 10px;
}

div
{
color: white;
background-color: 009900;
margin: 2px;
font-size: 25px;
}
</style>

</head>

<body>
<div > div tag </div>
<div > div tag </div>
<div > div tag </div>
<div > div tag </div>
</body>
</html>
```

### 2.3.2. Span tag

The HTML <span> is an inline element and it can be used to group inline-elements in an HTML document. This tag also does not provide any visual change on the block but has more meaning when it is used with CSS.

The difference between the <span> tag and the <div> tag is that the <span> tag is used with inline elements whereas the <div> tag is used with block-level elements.

Following example for <span> using CSS(Cascading Style Sheets which we discuss later)

**Example**

```
<!DOCTYPE html>
<html>
```

```

<body>
  <h2>Welcome To NATIONAL SANSKRIT UNIVERSITY</h2>

  <p> NATIONAL SANSKRIT is a
    <span style="color:red;font-weight:bolder">
      computer science</span> portal for
    <span style="background-color: lightgreen;"> SANSKRIT </span>.
  </p>
</body>
</html>

```

## 2.4. List tag

HTML lists are used to present list of information in well formed and semantic way. There are three different types of list in HTML and each one has a specific purpose and meaning.

- **Unordered list** — Used to create a list of related items, in no particular order.
- **Ordered list** — Used to create a list of related items, in a specific order.
- **Description list** — Used to create a list of terms and their descriptions.

**Note: inside a list item you can put text, images, links, line breaks, etc. you can also place an entire list inside a list item to create the nested list.**

### 2.4.1. HTML Unordered Lists

An unordered list is a list of items. The list items are marked with bullets (typically small black circles).

An unordered list starts with the <ul> tag. Each list item starts with the <li> tag.

```

<ul>
<li>Coffee</li>
<li>Milk</li>
</ul>

```

Here is how it looks in a browser:

- Coffee
- Milk

Inside a list item you can put paragraphs, line breaks, images, links, other lists, etc.

### 2.4.2. Ordered Lists

An ordered list is also a list of items. The list items are marked with numbers.

An ordered list starts with the <ol> tag. Each list item starts with the <li> tag.

```
<ol>
<li>Coffee</li>
<li>Milk</li>
</ol>
```

Here is how it looks in a browser:

1. Coffee
2. Milk

Inside a list item you can put paragraphs, line breaks, images, links, other lists, etc.

### **Description Lists**

A definition list is not a list of items. This is a list of terms and explanation of the terms.

A definition list starts with the <dl> tag. Each definition-list term starts with the <dt> tag. Each definition-list definition starts with the <dd> tag.

```
<dl>
<dt>Coffee</dt>
<dd>Black hot drink</dd>
<dt>Milk</dt>
<dd>White cold drink</dd>
</dl>
```

Here is how it looks in a browser:

Coffee

Black hot drink

Milk

White cold drink

Inside a definition-list definition (the <dd> tag) you can put paragraphs, line breaks, images, links, other lists, etc.

### **Example**

```
<html>
<body>
<h4>Numbered list:</h4>
```

```
<ol>
  <li>Apples</li>
  <li>Bananas</li>
  <li>Lemons</li>
  <li>Oranges</li>
</ol>
```

#### <h4>Letters list:</h4>

```
<ol type="A">
  <li>Apples</li>
  <li>Bananas</li>
  <li>Lemons</li>
  <li>Oranges</li>
</ol>
```

#### <h4>Lowercase letters list:</h4>

```
<ol type="a">
  <li>Apples</li>
  <li>Bananas</li>
  <li>Lemons</li>
  <li>Oranges</li>
</ol>
```

#### <h4>Roman numbers list:</h4>

```
<ol type="I">
  <li>Apples</li>
  <li>Bananas</li>
  <li>Lemons</li>
  <li>Oranges</li>
</ol>
```



<h4>Lowercase Roman numbers list:</h4>

<ol type="i">

<li>Apples</li>

<li>Bananas</li>

<li>Lemons</li>

<li>Oranges</li>

</ol>

</body>

</html>

## Output

*Numbered list:*

1. Apples
2. Bananas
3. Lemons
4. Oranges

*Letters list:*

- A. Apples
- B. Bananas
- C. Lemons
- D. Oranges

*Lowercase letters list:*

- a. Apples
- b. Bananas
- c. Lemons
- d. Oranges

*Roman numbers list:*

- I. Apples
- II. Bananas
- III. Lemons
- IV. Oranges

*Lowercase Roman numbers list:*

- i. Apples
- ii. Bananas
- iii. Lemons
- iv. Oranges

SAQ

1. Create a web page which displays subject and courses in National Sanskrit University by using ordered and unordered list?

### List Tags

Tag	Description
<ol>	Defines an ordered list
<ul>	Defines an unordered list
<li>	Defines a list item
<dl>	Defines a definition list
<dt>	Defines a definition term
<dd>	Defines a definition description
<dir>	Deprecated. Use <ul> instead
<menu>	Deprecated. Use <ul> instead

### 2.5. Summary

- There are six different HTML headings which are defined with the <h1> to <h6> tags, from highest level h1 (main heading) to the least level h6 (least important heading).
- HTML supports ordered, unordered and definition lists.
- The <marquee> is a non-standard HTML tag which was used to create a scrolling text or an image
- The <center> tag in HTML is used to set the alignment of text into the center.

## **2.6. SAQ**

1. Discuss about various elements for text formatting.
2. Explain various heading tags that are used in HTML.
3. What is the use of marquee tag>
4. What is the purpose of span tag?
5. Discuss about various list tags available in HTML.

### **3.0 Objective**

#### **3.1. Introduction**

#### **3.2. Inserting Images in Web Pages**

#### **3.3. Creating Hyperlinks in a Web Pages**

#### **3.4. Summary**

#### **3.5. SAQ**

---

### **3.0. Objective**

This unit helps in inserting images and hyper link in web pages.

#### **3.1. Introduction**

Images are not technically inserted into a web page; images are linked to web pages. An electronic link providing direct access from one distinctively marked place in a hypertext or hypermedia document to another in the same or a different document which is known as hyper link.

#### **3.2. Inserting Image in Web Pages**

With HTML you can display images in a document

#### **The Image Tag and the Src Attribute**

In HTML, images are defined with the <img> tag.

The <img> tag is empty, which means that it contains attributes only and it has no closing tag.

To display an image on a page, you need to use the src attribute. Src stands for "source". The value of the src attribute is the URL of the image you want to display on your page.

The syntax of defining an image:

```

```

The URL points to the location where the image is stored. An image named "boat.gif" located in the directory "images" on "www.images.com" has the URL: <http://www.images.com/images/boat.gif>.

The browser puts the image where the image tag occurs in the document. If you put an image tag between two paragraphs, the browser shows the first paragraph, then the image, and then the second paragraph

### **The Alt Attribute**

The alt attribute is used to define an "alternate text" for an image. The value of the alt attribute is an author-defined text:

```

```

The "alt" attribute tells the reader what he or she is missing on a page if the browser can't load images. The browser will then display the alternate text instead of the image. It is a good practice to include the "alt" attribute for each image on a page, to improve the display and usefulness of your document for people who have text-only browsers.

**Notes: If an HTML file contains ten images - eleven files are required to display the page right. Loading images take time, so my best advice is: Use images carefully**

### **Example**

```
<html>
```

```
<body>
```

```
<p>
```

An image:

```

```

```
</p>
```

```
<p>
```

A moving image:

```

```

```
</p>
```

```
<p>
```

Note that the syntax of inserting a moving image is no different from that of a non-moving image.

```
</p>
```

```
</body>
```

```
</html>
```

### Output

An image: 

A moving image: 

Note that the syntax of inserting a moving image is no different from that of a non-moving image.

### Example for ALT

```
<html>
```

```
<body>
```

```

```

```
<p>
```

Text-only browsers cannot display images and will only display the text that is specified in the "alt" attribute for the image. Here, the "alt"-text is "Go Left".</p>

```
<p>
```

Note that if you hold the mouse pointer over the image, most browsers will display the "alt"-text.

```
</p>
```

```
</body>
```

```
</html>
```

### Output



Text-only browsers cannot display images and will only display the text that is specified in the "alt" attribute for the image. Here, the "alt"-text is "Go Left".

**Note that if you hold the mouse pointer over the image, most browsers will display the "alt"-text**

### Image Tags

Tag	Description
<img>	Defines an image
<map>	Defines an image map
<area>	Defines a clickable area inside an image map

#### SAQ

1. Write a short note on body and image tags?

### 3.3. Creating Hyperlinks in a Web Pages

HTML make it possible to define hyperlinks to other information items located all over the world, thus allowing documents to join the global information space known as the World Wide Web. Linking is possible because every document on the Web has a unique address, known as a Uniform Resource Locator (URL).

## **Links and addressing**

HTML uses a hyperlink to link to another document on the Web

### **The Anchor Tag and the Href Attribute**

HTML uses the <a> (anchor) tag to create a link to another document.

An anchor can point to any resource on the Web: an HTML page, an image, a sound file, a movie, etc.

The syntax of creating an anchor:

```
<a href="url">Text to be displayed</a>
```

The <a> tag is used to create an anchor to link from, the href attribute is used to address the document to link to, and the words between the open and close of the anchor tag will be displayed as a hyperlink.

### **The Target Attribute**

With the target attribute, you can define where the linked document will be opened.

The line below will open the document in a new browser window:

```
<a href="http://www.rastiya sanskrit vidyapeetha.ac.in/"  
target="_blank">Visit RSVP!</a>
```

### **The Anchor Tag and the Name Attribute**

The name attribute is used to create a named anchor. When using named anchors we can create links that can jump directly into a specific section on a page, instead of letting the user scroll around to find what he/she is looking for.

Below is the syntax of a named anchor

```
<a name="label">Text to be displayed</a>
```

The name attribute is used to create a named anchor. The name of the anchor can be any text you care to use.



The line below defines a named anchor:

```
<a name="tips">Useful Tips Section</a>
```

You should notice that a named anchor is not displayed in a special way.

To link directly to the "tips" section, add a # sign and the name of the anchor to the end of a URL, like this:

```
<a href="http:// www.rastiya sanskrit vidyapeetha.ac.in /html_links.asp#tips">
Jump to the Useful Tips Section</a>
```

A hyperlink to the Useful Tips Section from WITHIN the file "html\_links.asp" will look like this:

```
<a href="#tips">Jump to the Useful Tips Section</a>
```

### Example

```
<html>
<body>
<a href="lastpage.htm" target="_blank">Last Page</a>
<p>
```

If you set the target attribute of a link to "\_blank", the link will open in a new window.

```
</p>
</body>
</html>
```

### Output

Last Page

#### SAQ

1. Write a short note on image tag in detail?
2. Create a web page to display alphabets and if you click on alphabet it should navigate to other pages details ?

If you set the target attribute of a link to "\_blank", the link will open in a new window

### **3.4. Summary**

1. In HTML, images are defined with the <img> tag.
2. The <img> tag is empty, which means that it contains attributes only and it has no closing tag.
3. Tables are defined with the <table> tag. A table is divided into rows (with the <tr> tag), and each row is divided into data cells (with the <td> tag).
4. HTML links are hyperlinks. You can click on a link and jump to another document. When you move the mouse over a link, the mouse arrow will turn into a little hand. Note: A link does not have to be text
5. When you click on a link, the link will take you to the target of the link, which may be a webpage, document or other online content. Websites use hyperlinks as a way to navigate online content.

### **3.5. SAQ**

1. Write a short note on Image tag?
2. Write a short note on Hyper link tag?
3. List different types of Hyperlinks?
4. Create a web page to display alphabets and if you click on alphabet it should navigate to other pages details ?

#### **4.0. Objective**

#### **4.1. Introduction**

#### **4.2. Creating Table in a Web Page**

#### **4.3. Summary**

#### **4.4. SAQ**

---

#### **4. 0. Objective**

This unit helps in inserting tables in web pages.

#### **4.1. Introduction**

The HTML table model allows authors to arrange data -- text, preformatted text, images, links, forms, form fields, other tables, etc. -- into rows and columns of cells

#### **4.2. Creating Table in a Web Page**

Tables are defined with the <table> tag. A table is divided into rows (with the <tr> tag), and each row is divided into data cells (with the <td> tag). The letters td stands for "table data," which is the content of a data cell. A data cell can contain text, images, lists, paragraphs, forms, horizontal rules, tables, etc.

#### **Syntax**

```
<table border="1">
<tr>
<td>row 1, cell 1</td>
<td>row 1, cell 2</td>
</tr>
<tr>
<td>row 2, cell 1</td>
<td>row 2, cell 2</td>
</tr>
</table>
```

How it looks in a browser:

row 1, cell 1	row 1, cell 2
row 2, cell 1	row 2, cell 2

### **Tables and the Border Attribute**

If you do not specify a border attribute the table will be displayed without any borders. Sometimes this can be useful, but most of the time, you want the borders to show.

To display a table with borders, you will have to use the border attribute:

```
<table border="1">  
<tr>  
<td>Row 1, cell 1</td>  
<td>Row 1, cell 2</td>  
</tr>  
</table>
```

### **Headings in a Table**

Headings in a table are defined as follows:

```
<table border="1">  
<tr>  
<th>Heading</th>  
<th>Another Heading</th>  
</tr>  
<tr>  
<td>row 1, cell 1</td>  
<td>row 1, cell 2</td>  
</tr>  
<tr>  
<td>row 2, cell 1</td>  
<td>row 2, cell 2</td>  
</tr>
```

</table>ined with the <th> tag.

### How it looks in a browser:

Heading	Another Heading
row 1, cell 1	row 1, cell 2
row 2, cell 1	row 2, cell 2

### Empty Cells in a Table

Table cells with no content are not displayed very well in most browsers.

```
<table border="1">
<tr>
<td>row 1, cell 1</td>
<td>row 1, cell 2</td>
</tr>
<tr>
<td>row 2, cell 1</td>
<td></td>
</tr>
</table>
```

### How it looks in a browser:

row 1, cell 1	row 1, cell 2
row 2, cell 1	

Note that the borders around the empty table cell are missing (NB! Mozilla Firefox displays the border).

To avoid this, add a non-breaking space (&nbsp;) to empty data cells, to make the borders visible:

```

<table border="1">
<tr>
<td>row 1, cell 1</td>
<td>row 1, cell 2</td>
</tr>
<tr>
<td>row 2, cell 1</td>
<td>&nbsp;</td>
</tr>
</table>

```

**How it looks in a browser:**

row 1, cell 1	Row 1, cell 2
row 2, cell 1	

SAQ

1. Write a short note on table tag in detail?
2. Create a web page to display all student details ?

**Table Tags**

Tag	Description
<table>	Defines a table
<th>	Defines a table header
<tr>	Defines a table row
<td>	Defines a table cell
<caption>	Defines a table caption
<colgroup>	Defines groups of table columns

<col>	Defines the attribute values for one or more columns in a table
<thead>	Defines a table head
<tbody>	Defines a table body
<tfoot>	Defines a table footer

### 4.3. Summary

1. Tables are defined with the <table> tag. A table is divided into rows (with the <tr> tag), and each row is divided into data cells (with the <td> tag).
2. A wide variety of tables can be created with only five tags: the <table> tag, which encapsulates a table and its elements in the document's body content.
3. The <tr> tag, which defines a table row. the <th> and <td> tags, which define the table's headers and data cells and the <caption> tag.

# BASICS OF INTERNET TECHNOLOGY

## Unit - I

### 1.0. Objective

#### 1.1. Introduction

#### 1.2. Working with Multimedia in a Web Page

#### 1.3. Embedding Multimedia Objects in a Web Page

#### 1.4. Summary

#### 1.5. SAQ

---

### 1.0. Objective

#### Able to design web page using multimedia

#### 1.1. Introduction

Multimedia comes in many different formats. It can be almost anything you can hear or see, like images, music, sound, videos, records, films, animations, and more. Web pages often contain multimedia elements of different types and formats

#### 1.2. Working with Multimedia in a Web Page

HTML helps you to add multimedia files on your website **by providing various multimedia tags**. These tags include AUDIO, VIDEO, EMBED, and OBJECT. The AUDIO tag is used to display the audio file on the Web page, whereas the VIDEO tag is used to display the video files on the Web page.

Sometimes you need to add music or video into your web page. The easiest way to add video or sound to your web site is to include the special HTML tag called **<embed>**. This tag causes the browser itself to include controls for the multimedia automatically provided browser supports **<embed>** tag and given media type.

You can also include a **<noembed>** tag for the browsers which don't recognize the **<embed>** tag. You could, for example, use **<embed>** to display a movie of your choice, and **<noembed>** to display a single JPG image if browser does not support **<embed>** tag.

#### The **<embed>** Tag Attributes

Following is the list of important attributes which can be used with **<embed>** tag.



Sr.No	Attribute & Description
1	<p><b>align</b></p> <p>Determines how to align the object. It can be set to either center, <i>left</i> or <i>right</i>.</p>
2	<p><b>autostart</b></p> <p>This boolean attribute indicates if the media should start automatically. You can set it either true or false.</p>
3	<p><b>loop</b></p> <p>Specifies if the sound should be played continuously (set loop to true), a certain number of times (a positive value) or not at all (false)</p>
4	<p><b>playcount</b></p> <p>Specifies the number of times to play the sound. This is alternate option for <i>loop</i> if you are using IE.</p>
5	<p><b>hidden</b></p> <p>Specifies if the multimedia object should be shown on the page. A false value means no and true values means yes.</p>
6	<p><b>width</b></p> <p>Width of the object in pixels</p>
7	<p><b>height</b></p> <p>Height of the object in pixels</p>
8	<p><b>name</b></p> <p>A name used to reference the object.</p>
9	<p><b>src</b></p> <p>URL of the object to be embedded.</p>
10	<p><b>volume</b></p>

Controls volume of the sound. Can be from 0 (off) to 100 (full volume).
---

### Supported Video Types

You can use various media types like Flash movies (.swf), AVI's (.avi), and MOV's (.mov) file types inside embed tag.

- **.swf files** – are the file types created by Macromedia's Flash program.
- **.wmv files** – are Microsoft's Window's Media Video file types.
- **.mov files** – are Apple's Quick Time Movie format.
- **.mpeg files** – are movie files created by the Moving Pictures Expert Group

### Example

```
<!DOCTYPE html>
<html>

  <head>
    <title>HTML embed Tag</title>
  </head>

  <body>
    <embed src = "/html/yourfile.swf" width = "200" height = "200" >
      <noembed><img src = "yourimage.gif" alt = "Alternative Media" ></noembed>
    </embed>
  </body>

</html>
```

### 1.3. Embedding Multimedia Objects in a Web Page

To embed a video in an HTML page, use the <iframe> element. The source attribute included the video URL. For the dimensions of the video player, set the width and height of the video appropriately.

### Example with syntax

```
#!DOCTYPE html
<html>
<head> </head>
<body>
<iframe width = "width" src="video_URL">
</iframe>
</body>
</html>
```

The Video URL is the video embed link.

### Example

```
<!DOCTYPE html>
```

```
<html>
  <head>
    <title>HTML Video embed</title>
  </head>
  <body>
    <p>Learn Eclipse</p>
    <br />
    <iframe width="560" height="315"
src="https://www.youtube.com/embed/y881t8ilMyc"
allowfullscreen></iframe>
  </body>
</html>
```

#### 1.4. Summary

- <object> tag can be used to embed multimedia also in to the HTML document.
- Embedded media refers to multimedia items that are placed inside a web page from third-party or off-site resources.
- Multimedia is gaining popularity on the Web with several technologies to support use of animation, video, and audio to supplement the traditional media of text and images.
- Multimedia content visually draw immense traffic, it makes your website visually stunning.

#### 1.5. SAQ

1. Define Multimedia?
2. Discuss how to embed multimedia in web pages?
3. list different types of multimedia used in web pages?

## **2.0. Objective**

### **2.1. Introduction**

### **2.2. Working with Frames**

### **2.3. Summary**

### **2.4. SAQ**

---

## **2.0. Objective**

Learn how to divide the web browser window into multiple sections where each section can be loaded separately

### **2.1. Introduction**

HTML Frames are used to divide the web browser window into multiple sections where each section can be loaded separately. A frameset tag is the collection of frames in the browser window. Instead of using body tag, use frameset tag in HTML to use frames in web browser.

### **2.2. Working with Frames**

A frame is an independent scrolling region, or window, of a Web page. Every Web page may be divided into many individual frames, which can even be nested within other frames. Fixed screen sizes limit how many frames can really be used simultaneously. Each frame in a window may be separated from the others with a border, in this way; a framed document may resemble a table. However, frames aren't a fancy form of tables. Each separate frame may contain a different document, indicated by a unique URL. Because the documents included in a framed region may be much larger than the room available onscreen, each frame may provide a scroll bar or other controls to manipulate the size of the frame. Individual frames usually are named, so that they may be referenced through links or scripting, allowing the contents of one frame to affect the contents of another. This referencing capability is a major difference between tables and frames. Frames provide layout facilities and, potentially, navigation.

Frames are included in a HTML document through the <FRAMESET> and <FRAME> elements. The <FRAMESET> and <FRAME> elements constitute the HTML needed for frames.

The disadvantages of using frames are:

The web developer must keep track of more HTML documents

It is difficult to print the entire page

### **The Frameset Tag**

The <frameset> tag defines how to divide the window into frames

Each frameset defines a set of rows **or** columns

The values of the rows/columns indicate the amount of screen area each row/column will occupy

#### **syntax:**

<FRAMESET>

<FRAMESET CLASS="class name(s)" COLS="list of columns" ID="unique alphanumeric identifier" ROWS="list of rows" STYLE="style information" TITLE="advisory text" onload="script" onunload="script">

<FRAME> elements and <NOFRAMES>

</FRAMESET>

#### **syntax:**

<FRAME>

<FRAME CLASS="class name(s)" FRAMEBORDER="0 | 1" ID="unique alphanumeric identifier" LONGDESC="URL of description" MARGINHEIGHT="pixels" MARGINWIDTH="pixels" NAME="string" NORESIZE SCROLLING="AUTO | NO | YES" SRC="URL" of frame contents" STYLE="style information" TITLE="advisory text">

In the example below we have a frameset with two columns. The first column is set to 25% of the width of the browser window. The second column is set to 75% of the width of the browser window. The HTML document "frame\_a.htm" is put into the first column, and the HTML document "frame\_b.htm" is put into the second column:

```
<frameset cols="25%,75%">
  <frame src="frame_a.htm">
  <frame src="frame_b.htm">
</frameset>
```

**Note:** The frameset column size value can also be set in pixels (cols="200,500"), and one of the columns can be set to use the remaining space (cols="25%,\*").

### **Basic Notes - Useful Tips**

If a frame has visible borders, the user can resize it by dragging the border. To prevent a user from doing this, you can add noresize="noresize" to the <frame> tag.

Add the <noframes> tag for browsers that do not support frames.

**Important:** You cannot use the <body></body> tags together with the <frameset></frameset> tags! However, if you add a <noframes> tag containing some text for browsers that do not support frames, you will have to enclose the text in <body></body> tags! See how it is done in the first example below.

### **Frame Targeting**

When using frames, you often may find that making the links in one frame target another frame is beneficial. This way, when a user clicks a button or activates a link in one framed document, the requested page loads in another frame. Link targeting has two steps:

- 1.Ensure frame naming by setting the NAME attribute in the <FRAME> element to a unique name.
- 2.Use the TARGET attribute in the <A> element to set the target for the anchor. For example, a link sych as <A HREF="http://www.yahoo.com"

TARGET="Display"> loads the site specified by the HREF into the window called Display, if such a frame exists. If the target specified by the name doesn't exist, the link loads over the window it is in.

Some particular values for the TARGET attribute have special meaning, which are summarized in the following table.

**Reserved TARGET values:**

Value	Meaning
_blank	Load the page into a new, generally unnamed, window.
_self	Load the page over the current frame.
_parent	Load the link over the parent frame.
_top	Load the link over the frames in the window.

**No frames**

The noframes element displays text for browsers that do not handle frames. The noframes element goes inside the frameset element.

**Note:** If a browser handles frames, it will not display the text in the noframes element.

**Important:** If you add a <noframes> tag to a frameset, you will have to enclose the text in <body></body> tags!

**Example**

```
<html>  
  
<frameset cols="25%,50%,25%">  
  
  <frame src="frame_a.htm">  
  
  <frame src="frame_b.htm">  
  
  <frame src="frame_c.htm">
```

<noframes>

<body>Your browser does not handle frames!</body>

</noframes>

</frameset>

</html>

### **The Use of <NOFRAMES>**

The <NOFRAMES> element is used to enclose the HTML and text that should be displayed when a browser that does not support frames accesses the Web page. The <NOFRAMES> element should be found only within the <FRAMESET> element. The contents of the <NOFRAMES> element should be correct HTML, potentially including the <BODY> element, which can be used as an alternative form for browsers that don't support frames.

### **Floating Frames**

A floating frame introduced by Microsoft, has been incorporated in the HTML 4 standard. The idea of the floating frame is to create an inline framed region, or window, that acts similarly to any other embedded object, insofar as text can be flowed around it. An inline frame is defined by the <IFRAME> element and may occur anywhere within the <BODY> element of an HTML document. Compare this to the <FRAME> element, which should occur only within the <FRAMESET> element and the <FRAMESET> element should preclude the <BODY> element. The major attributes to set for the <IFRAME> element include SRC, HEIGHT, and WIDTH. The SRC is set to the URL of the file to load, while the HEIGHT and WIDTH are set to either the pixel or percentage value of the screen that the floating frame region should consume. Like an <IMG> element, floating frames should support ALIGN, HSPACE, and VSPACE attributes for basic positioning within the flow of text. Note that, unlike the <FRAME> element, the <IFRAME> element comes with a close tag. <IFRAME> and </IFRAME> should contain any HTML markup code and text that is supposed to be displayed in browsers that don't support floating frames.



## Using Frames

One of the biggest problems with frames is that they initially were used simply because they existed/. Framed documents can provide considerable benefit, but at a price. A potential benefit of frames is that they allow content to be fixed onscreen. One frame may contain a table of contents, while the other frame contains the actual information. Keeping the table of contents onscreen provides a convenient way to navigate the body of information. Furthermore, if one frame has fixed navigation, the user may perceive the Web interface to be more responsive, because only part of the screen needs to update between selections. The primary benefit of frames is to present two or more things simultaneously, but this extra window of information has its costs.

## Frame Problems

The problems with frames are numerous, including design issues, navigation confusion, bookmarking problems, loss of URL context, and printing issues. The only potential design issue is the possibility that a framed document may sacrifice valuable screen real estate because of scroll bars, which could pose trouble for people with lower-resolution monitors. The only way to get around this problem is to limit the number of frames used on a page. Additional navigational problems include loss of context, because the URL of the document tends not to change, which accounts for why bookmarking does not work as expected. Many people use URLs as a way to orient themselves at a site. Frames give up this clue to location.

## Frame Tags

Tag	Description
<frameset>	Defines a set of frames
<frame>	Defines a sub window (a frame)
<noframes>	Defines a noframe section for browsers that do not handle frames
<iframe>	Defines an inline sub window (frame)

### **2.3. Summary**

- A frame is an independent scrolling region, or window, of a Web page. Every Web page may be divided into many individual frames, which can even be nested within other frames.
- Fixed screen sizes limit how many frames can really be used.
- The problems with frames are numerous, including design issues, navigation confusion, bookmarking problems, loss of URL context, and printing issues.
- The noframes element displays text for browsers that do not handle frames. The noframes element goes inside the frameset element.

### **2.4. SAQ**

1. How to create frames in web page.
2. Design a home page of the college using frames.
3. Discuss noframes in web design?

## Unit - III

- 3.0. Objective
  - 3.1. Introduction
  - 3.2. Dynamic HTML
  - 3.3. Cascading Style Sheets(CSS)
  - 3.4. Summary
  - 3.5. SAQ
- 

### **3.0. Objective**

The aim of this chapter is to understand about DHTML

### **3.1. Introduction**

Cascading Style Sheet, it helps in the styling of the web pages and helps in designing of the pages. The CSS rules for DHTML will be modified at different levels using JS with event handlers which adds a significant amount of dynamism with very little code.

### **3.2. DHTML (Dynamic Hyper Text Markup Language )**

Dynamic Hyper Text Markup Language (DHTML) is a combination of Web development technologies used to create dynamically changing websites. Web pages may include animation, dynamic menus and text effects. The technologies used include a combination of HTML, Java Script or VB Script, CSS and the Document Object Model(DOM).

Designed to enhance a Web user's experience, DHTML includes the following features:

- Dynamic content, which allows the user to dynamically change Web page content
- Dynamic positioning of Web page elements
- Dynamic style, which allows the user to change the Web page's color, font, size or content.

### **Uses of DHTML**

Following are the uses of DHTML (Dynamic HTML):

- It is used for designing the animated and interactive web pages that are developed in real-time.
- DHTML helps users by animating the text and images in their documents.
- It allows the authors for adding the effects on their pages.
- It also allows the page authors for including the drop-down menus or rollover buttons.
- This term is also used to create various browser-based action games.
- It is also used to add the ticker on various websites, which needs to refresh their content automatically.

### Features of DHTML

Following are the various characteristics or features of DHTML (Dynamic HTML):

- Its simplest and main feature is that we can create the web page dynamically.
- **Dynamic Style** is a feature, that allows the users to alter the font, size, color, and content of a web page.
- It provides the facility for using the events, methods, and properties. And, also provides the feature of code reusability.
- It also provides the feature in browsers for data binding.
- Using DHTML, users can easily create dynamic fonts for their web sites or web pages.
- With the help of DHTML, users can easily change the tags and their properties.
- The web page functionality is enhanced because the DHTML uses low-bandwidth effect.

### Difference between HTML and DHTML

Following table describes the differences between HTML and DHTML:

HTML (Hypertext Markup language)	DHTML (Dynamic Hypertext Markup language)
1. HTML is simply a markup language.	1. DHTML is not a language, but it is a set of technologies of web development.
2. It is used for developing and creating web	2. It is used for creating and designing the

pages.	animated and interactive web sites or pages.
3. This markup language creates static web pages.	3. This concept creates dynamic web pages.
4. It does not contain any server-side scripting code.	4. It may contain the code of server-side scripting.
5. The files of HTML are stored with the .html or .htm extension in a system.	5. The files of DHTML are stored with the .dhtm extension in a system.
6. A simple page which is created by a user without using the scripts or styles called as an HTML page.	6. A page which is created by a user using the HTML, CSS, DOM, and JavaScript technologies called a DHTML page.
7. This markup language does not need database connectivity.	7. This concept needs database connectivity because it interacts with users.

Let us see how to create a web page using DHTML, simply uses the **document.write()** method of JavaScript in the DHTML. In this example, we type the JavaScript code in the **<body>** tag.

```

<head>
<title>
Method of a JavaScript
</title>
</head>
<body>
<script type="text/javascript">
document.write("NATIONA SANSKRIT UNIVERSITY");
</script>
</body>
</html>

```

#### **OUT PUT**

NATIONAL SANSKRIT UNIVERSITY

### **3.3. Cascading Style Sheets**

#### **What is CSS?**

- 1) CSS stands for Cascading Style Sheets
- 2) Styles define how to display HTML elements
- 3) Styles are normally stored in Style Sheets
- 4) Styles were added to HTML 4.0 to solve a problem
- 5) External Style Sheets can save you a lot of work
- 6) External Style Sheets are stored in CSS files
- 7) Multiple style definitions will cascade into one

#### **Styles Solve a Common Problem**

HTML tags were originally designed to define the content of a document. They were supposed to say "This is a header", "This is a paragraph", "This is a table", by using tags like `<h1>`, `<p>`, `<table>`, and so on. The layout of the document was supposed to be taken care of by the browser, without using any formatting tags.

As the two major browsers - Netscape and Internet Explorer - continued to add new HTML tags and attributes (like the `<font>` tag and the color attribute) to the original HTML specification, it became more and more difficult to create Web sites where the content of HTML documents was clearly separated from the document's presentation layout.

#### **The Rise of Style Sheets**

Basically, style sheets separate the structure of a document from its presentation. Dividing layout and presentation has many theoretical benefits, most importantly, it can provide for flexible documents that display equally well across many types of output devices. As early as 1993, people have been interested in adding more layout control to HTML. Many approaches have been discussed and many continue to be used. Because of the theoretical benefits of style sheets, they have been the favorite solution of the standards bodies. More than one type of style sheet exists. Many industry pundits support a type of sheet known as Document Style Semantics and Specification Language (DSSSL), developed by the SGML community. The most

recent addition is Extensible Style Language (XSL), an industry proposal based on DSSSL that uses Extensible Markup Language (XML) syntax.

## **Style Sheet Basics**

CSS1 style sheets rely on an underlying markup structure, such as HTML. They are not a replacement for HTML. Without a binding to an element, a style really doesn't mean anything. The purpose of a style sheet is to create a presentation for a particular element or set of elements. Binding an element to a style specification consists of an element, followed by its associated style information within curly braces:

```
Element      {style specification}
```

Suppose that you want to bind a style rule to the <H1> element so that a 28-point Impact font is always used. The following rule would result in the desired display:

```
H1      {font-family: Impact;
         font-size: 28pt}
```

In general, a style specification or style sheet is simply a collection of rules. These rules include a selector, an HTML element, a CLASS name, or an ID value, which is bound to a style property such as font-family, followed by a colon and the values for that style property. Multiple style rules may be included in a style specification by separating the rules with semicolons. You can also use many shorthand notations and grouping rules that are available. Style sheets alone do nothing. First, you must bind the rule to a tag or class of HTML objects. Currently, more than 50 properties are specified under CSS1 that affect the presentation of an HTML document, and more than 50 more properties are defined under CSS2. Unfortunately, not all of them are supported consistently across the major browsers. Even worse, most of the newer style properties defined by CSS2 are not supported by any browser.

## **Adding Style to a Document**

Style information may be included in an HTML document in any one of three basic ways:

1. Use an outside style sheet, either by importing it or by linking to it.
2. Embed a document-wide style in the <HEAD> element of the document.

3. Provide an inline style exactly where the style needs to be applied.

Each of these style sheet approaches has its own pros and cons, as listed in the following table:

**Comparison of Style Sheet Approaches:**

Pros	External Style Sheets	Document-Wide Style	Inline Style
	Can set style for many documents with one style sheet	Can control style for a document in one place.  No additional download time for style information.	Can control style to a single character instance.  Overrides any external or document styles.
Cons	Require extra download time for the style sheet, which may delay page rendering	Need to reapply style information for other documents	Need to reapply style information throughout the document and outside documents.  Bound too closely to HTML, difficult to update.

In brief, Cascading style sheets provide better control over the look and feel of the Web pages. Style sheets aren't just useful for making attractive pages. By dividing structure and style, they make documents simpler, and easier to manipulate. While style sheets provide a great deal of flexibility in creating pages, they are not fully



implemented yet in today's browsers. Some inconsistencies exist between implementations. When used in a non-obtrusive manner, style sheets are a great way to improve the layout of pages, without locking into a proprietary solution.

The following is an example that implements CSS.

```
<!DOCTYPE html PUBLIC "working with CSS">
<html>
<head>
  <title>My first styled page</title>
  <style type="text/css">
< body> {
  padding-left: 11em;
  font-family: Georgia, "Times New Roman",
    Times, serif;
  color: purple;
  background-color: #d8da3d }
ul.navbar {
  list-style-type: none;
  padding: 0;
  margin: 0;
  position: absolute;
  top: 2em;
  left: 1em;
  width: 9em }
h1 {
  font-family: Helvetica, Geneva, Arial,
    SunSans-Regular, sans-serif }
ul.navbar li {
  background: white;
  margin: 0.5em 0;
  padding: 0.3em;
  border-right: 1em solid black }
ul.navbar a {
  text-decoration: none }
a:link {
  color: blue }
a:visited {
  color: purple }
address {
  margin-top: 1em;
  padding-top: 1em;
  border-top: thin dotted }
</style>
</head>
</body>
</html>
```

### SAQ

1. What is DHTML? How to implement?
2. Create a webpage to implement DHTML?

### 3.4. Summary

- Dynamic HyerText Markup Language (**DHTML**) is a combination of Web development technologies used to create dynamically changing websites.
- Web pages may include animation, dynamic menus and text effects. The technologies used include a combination of HTML, JavaScript or VB Script, CSS and the document object model (DOM).
- **CSS** is a language that describes the style of an HTML document. **CSS** describes how HTML elements should be displayed
- Javascript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages.

### 3.5. SAQ

1. What is DHTML? Explain various features of DHTML.
2. What is CSS? How to integrate CSS to DHTML?
3. Discuss about Style sheets.
4. Design a Web page to insert a table dynamically.

#### 4.0. Objective

#### 4.1. Introduction

#### 4.2. Attaching CSS to HTML Documents

##### 4.2.1. Using an InLine Style Sheet

##### 4.2.2. Embedding a Style Sheet

#### 4.3. Importing Style Sheet

#### 4.4. Linking an External Style Sheet

#### 4.5. Setting Dynamic Styles in a Web page

##### 4.5.1. Dynamic Alignment of the text in the Web Page

#### 4.6. Creating Dynamic Tables in a Web Page

#### 4.7. Creating animation using DHTML

#### 4.8. Summary

#### 4.9. SAQ

---

#### 4.0. Objective

This CSS will help both students as well as professionals who want to make their websites or personal blogs more attractive.

#### 4.2. Introduction

In this unit it discuss about few more topics related to CSS. Before going into depth let us check some applications of CSS

#### Applications of CSS

As mentioned before, CSS is one of the most widely used style language over the web. I'm going to list few of them here:

- **CSS saves time** - You can write CSS once and then reuse same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.
- **Pages load faster** - If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code means faster download times.

- **Easy maintenance** - To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.
- **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.
- **Multiple Device 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 cell phones or for printing.
- **Global web standards** - Now HTML attributes are being deprecated and it is being recommended to use CSS. So it's a good idea to start using CSS in all the HTML pages to make them compatible to future browsers.

#### 4.2. Attaching CSS to HTML Documents

CSS can either be attached as a separate document or embedded in the HTML document itself. There are three methods of including CSS in an HTML document:

- **Inline styles** — Using the style attribute in the HTML start tag.
- **Embedded styles** — Using the <style> element in the head section of a document.
- **External style sheets** — Using the <link> element, pointing to an external CSS file.

##### 4.2.1. InLine style sheets

Inline styles are styles that are applied to a specific element within the body section of the webpage. The style will be applied to that individual element only rather than to the entire page (internal style) or across all linked pages (external style sheet).

Inline style sheets is a term that refers to style sheet information being applied to the *current* element. By this, I mean that instead of defining the style once, then applying the style against all instances of an element (say the <p> tag), you only apply the style to the instance you want the style to apply to. Actually, it's not really a style sheet as such, so a more accurate term would be inline styles.

Consider the following example

```
<p style = "color:#ff9900"> this text has been styled using inline style sheets.</p>
```

Example:

```

<!DOCTYPE html>
  <html>
  <body>
    <h1 style="color:blue;">This is a Blue Heading</h1>

  </body>
</html>

```

#### 4.2.2. Embedding a style sheet

**Embedded CSS is also known as Internal style sheets.** This can be used when a single HTML document must be styled uniquely. The CSS rule set should be within the HTML file in the head section i.e the CSS is embedded within the HTML file.

Example:

```

<!DOCTYPE html>
<html>
  <head>
    <title>Internal CSS</title>
    <style>
      .main {
        text-align:center;
      }
      .GFG {
        color:#009900;
        font-size:50px;
        font-weight:bold;
      }
      .geeks {
        font-style:bold;
        font-size:20px;
      }
    </style>
  </head>
  <body>
    <div class = "main">
      <div class ="GFG">RASTRIYA SANSKRIT KIDYAPEETA</div>

      <div class ="sanskrit">
        A computer science portal for VIDYAPEETA students
      </div>
    </div>
  </body>
</html>

```

#### 4.3. Importing a Style Sheet

We can import additional CSS files inside another CSS declaration. **The @import rule is used for this purpose as it links a stylesheet in a document.** This is generally

used when one stylesheet is dependent upon another. It is specified at the top of the document after @charset declaration inside <head>.

## Usage

The @import rule allows you to import a style sheet into another style sheet.

The @import rule must be at the top of the document (but after any @charset declaration).

The @import rule also supports media queries, so you can allow the import to be media-dependent.

## 4.4. Linking an External style sheet

**External CSS:** External CSS contains separate CSS file which contains only style property with the help of tag attributes (For example class, id, heading, ... etc). CSS property written in a separate file with .css extension and should be linked to the HTML document using **link** tag. This means that for each element, style can be set only once and that will be applied across web pages. **Example:** The file given below contains CSS property. This file save with .css extension.

Let us consider an example

```
Body{
    Background-color: darkslategrey;
    Color: azure;
    font-size: 1.1em;
}
H1 {
    Color:coral;
}
#intro{
    Font-size:1.3em;
}
.colorful{
    Color:orange;
}
```

## Link to external style sheet

Below is the HTML file that is making use of the created external style sheet

- **link** tag is used to link the external style sheet with the html webpage.
- **href** attribute is used to specify the location of the external style sheet file.

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="nsktu.css"/>
  </head>

  <body>
    <div class = "main">
      <div class ="GFG">SanskritUniversity</div>
      <div id ="sanskrit">
        A computer science portal for National Sanskrit University
      </div>
    </div>
  </body>
</html>
```

### SAQ

1. Create a web page to display text with different colours ?
2. Discuss InLine Style Sheet?
3. Discuss how to import a style sheet?

## 4.5. Setting Dynamic Styles in a Web page

It is worth noting that while pre/postprocessor variables are only used at compilation-time, CSS variables can be used and updated dynamically. What does this mean? It means that **they are preserved in the actual CSS stylesheet**. So the notion that they are variables will remain even after the stylesheets are compiled.

#### 4.5.1. Dynamic Alignment of the text in the Web Page

The **text-align** property in CSS is used to specify the horizontal alignment of text in an element i.e., it is used to set the alignment of the content horizontally, inside a block element or table-cell box.

##### Syntax:

text-align: left|right|center|justify|initial|inherit;

**Default Value** : left if direction is ltr, and right if direction is rtl

##### Property Value:

- **left**: It is used to set the text-alignment into left. This is the default property.
- **right**: It is used to set the text-alignment to right.
- **center**: It is used to set the text-alignment into the center.
- **justify**: It is used to spreads the words into the complete line i.e., by stretching the content of an element.
- **initial**: It is used to set an element's CSS property to its default value.
- **inherit**: It is used to inherit a property to an element from its parent element property value.

Consider the following examples

```
<!DOCTYPE html>
<html>
<head>
  <title>text-align property</title>
  <style>
    h1 {
      color: green;
    }

    .main {
      border: 1px solid black;
    }

    .gfg1 {
      text-align: left;
    }

    .gfg2 {
      text-align: right;
    }

    .gfg3 {
      text-align: center;
    }
  </style>
</head>
</html>
```



```

        .fg4 {
            text-align: justify;
        }
    </style>
</head>

<body>
    <h1>GeeksforGeeks</h1>
    <h2>text-align property</h2>
    <div class="main">
        <h3>text-align: left;</h3>
        <div class="fg1">
            The course is designed for students
            as well as working professionals to
            prepare for coding interviews. This
            course is going to have coding questions
            from school level to the level needed
            for product based companies like Amazon,
            Microsoft, Adobe, etc.
        </div>
    </div>
    <br>
    <div class="main">
        <h3 style="text-align: right;">text-align: right;</h3>
        <div class="fg2">
            The course is designed for students
            as well as working professionals to
            prepare for coding interviews. This
            course is going to have coding questions
            from school level to the level needed
            for product based companies like Amazon,
            Microsoft, Adobe, etc.
        </div>
    </div>
    <br>
    <div class="main">
        <h3 style="text-align: center;">text-align: center;</h3>
        <div class="fg3">
            The course is designed for students
            as well as working professionals to
            prepare for coding interviews. This
            course is going to have coding questions
            from school level to the level needed for
            product based companies like Amazon,
            Microsoft, Adobe, etc.
        </div>
    </div>
    <br>
    <div class="main">

```

```

    <h3 style="text-align: justify;">text-align: justify;</h3>
    <div class="gfg4">
    The course is designed for students
    as well as working professionals to
    prepare for coding interviews. This
    course is going to have coding questions
    from school level to the level needed
    for product based companies like Amazon,
    Microsoft, Adobe, etc.
    </div>
</div>
</body>
</html>

```

#### 4.6.Creating Dynamic Tables in a Web Page

In invoice templates, a Dynamic Table is an essential element. A Dynamic Table is different from a standard table in that it has a variable number of rows. In a Print context it will automatically overflow into as many pages as necessary to output all rows and it can display a transport line.

following is demonstration of dynamic table in Web Page using javascript

**Note:** with the help of javascript it is easy to work with dynamic web page, so below example is demonstration of dynamic table in Web Page using javascript

```

<!DOCTYPE html>
<html>
<head>
  <title>HTML dynamic table using JavaScript</title>
  <script type="text/javascript" src="app.js"></script>
</head>
<body onload="load()">
<div id="myform">
<b>Simple form with name and age ...</b>
<table>
  <tr>
    <td>Name:</td>
    <td><input type="text" id="name"></td>
  </tr>
  <tr>
    <td>Age:</td>
    <td><input type="text" id="age">
      <input type="button" id="add" value="Add"
onclick="Javascript:addRow()"></td>
  </tr>
  <tr>
    <td>&nbsp;</td>

```



```

    background-color: red;
  }
</style>
<body>

<p><button onclick="myMove()">Click Me</button></p>

<div id ="container">
  <div id ="animate"></div>
</div>

<script>
function myMove() {
  let id = null;
  const elem = document.getElementById("animate");
  let pos = 0;
  clearInterval(id);
  id = setInterval(frame, 5);
  function frame() {
    if (pos == 350) {
      clearInterval(id);
    } else {
      pos++;
      elem.style.top = pos + "px";
      elem.style.left = pos + "px";
    }
  }
}
</script>

</body>
</html>

```

#### 4.8. Summary

- **CSS** is a language that describes the style of an HTML document. **CSS** describes how HTML elements should be displayed
- Javascript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages.
- Java Script can be implemented using Java Script statements that are placed within the `<script>...</script>` HTML tags in a web page.
- A Dynamic Table is different from a standard table in that it has a variable number of rows. In a Print context it will automatically overflow into as many pages as necessary to output all rows and it can display a transport line.

#### **4.9. SAQ**

1. Design a Web page to insert a table dynamically.
2. Design a web page to implement animation?