UNIT - I

INTRODUCTION:

HTML stands for Hyper Text Markup Language.HTML is a language for describing web pages.HTML is a language for describing web pages.HTML instructions divide the text of a document into blocks called elements.

An element in HTML represents some kind of structure or semantics and generally consists of a start tag, content, and an end tag.
The following is a paragraph element:

<p> This is the content of the paragraph element. </p>

Tags are used to mark up the start and end of an HTML element. A start tag consists of an opening angle bracket (<) followed by the element name, zero or more space separated attribute/value pairs, and a closing angle bracket (>).

Example:- A start tag with no attributes: <p>

End tags consist of an opening angle bracket followed by a forward slash, the element name, and a closing angle bracket </p>

An attribute defines a property for an element, consists of an attribute/value pair, and appears within the element's start tag. An element's start tag may contain any number of space separated attribute/ value pairs.
A markup language is a set of markup tags. HTML uses markup tags to describe web pages

Structure of an HTML page

A HTML page has two sections (Elements) HEAD , BODY. The head is used for text and tags that do not show directly on the page. The body is used for text and tags that are shown directly on the page .
TITLE, META are the Elements that appear within HEAD element.
The title of a webpage appears in your browser´s title bar when you view the page.
The META element used in the HEAD section is an empty element, i.e. the META element consists only of an opening tag. The most important use for this element is to give the search engines some basic information about your HTML document such as the most important keywords that appear inside the document and a short description of the document´s content.

<meta name="keywords" content="keyword1, keyword2, ...">

MIME types:
The World Wide Web actually uses MIME types (Multipurpose Internet Mail Extension) to define the type of a particular piece of information being sent from a Web server to a browser. A browser in turn determines, from the MIME type, how the data should be treated.

HTML Headings are defined with the <H1> to <H6> tags.<H1> defines the largest heading. <H6> defines the smallest heading.

```
<H1>This is h1</H1>
<H2>This is h2</H2>
<H3>This is h3</H3>
<H4>This is h4</H4>
<H5>This is h5</H5>
<H6>this is h6</H6>
```

Lists: HTML supports ordered, unordered and definition Lists

Unordered Lists

To create bulleted lists (Unordered Lists) we use a <ul> and a </ul> tag at the beginning and the end of the list. List items are added using <li> and </li> tags

Example 1: (Type the following code in Note pad editor , save as ex1.html and open in a browser)

```
<html>
<body>
<h4>Branches
Available:</h4> <ul
type="disc"> <li>CSEe</li>
  <li>ECE</li>
<li>EEE  </li>
</ul>
</body>
</html>
```

Note : <ul type="circle"> , <ul type="square"> are used to get Circle and Square symbols.

Ordered Lists (or) **Numbered Lists**

To create we use a Ordered Lists (or) Numbered Lists <ol> and a </ol> tag Example 2:

```
<html>
<body>
<h4>Branches
Available:</h4> <ol>
<li>CSE</li>
<li>ECE</li>
<li>EEE</li>
</ol>
</body>
</html>
```

Note: <ol type="A"> ,<ol type="a"> , <ol type="I"> , <ol type="i"> are used to create List with Capital Letters , Small Letters, Capital Roman Numbers, Small Roman Numbers as Numbers for the Lists.

Definition Lists is not a list of single items. It is a list of items (terms), with a description of each item (term). A definition list starts with a <dl> tag (definition list). Each term starts with a <dt> tag (definition term). Each description starts with a <dd> tag (definition description)

Example:

```
<html>
<body>
<dl>
<dt>Cascading Style Sheets</dt>
<dd><p>Style sheets are used to provide presentational suggestions.</p>
<p>Documents structured using XML or HTML are able to make use of
them.</p></dd> <dt>Content Management</dt>
<dd>The process of collecting, managing and publishing content to various
media.</dd> </dl>
</body>
</html>
```

Tables

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.

Example

```
<html>
<body>
<table
border="1"> <tr>
<td>Table cell 1</td><td>Table cell
2</td> </tr>
<tr>
<td>Table cell 3</td><td>Table cell
4</td> </tr>
</table>
</body>
</html>
```

Example

```
<html>
<body>
<table border="1"
> <tr>
<th colspan="2">Table
header</th> </tr>
<tr>
<td width="20%">Table cell 1</td><td>Table cell
2</td> </tr>
</table>
```

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

HTML Images

To embed an image into a web page, the image first needs to exist in either .jpg, .gif, or .png format To embed the image into your web page, use the <img /> tag,

```
<img src="image1.gif" width="100" height="100" alt="Image 1 " />
```

src -- Required attribute. This is the path to the image. It can be either an absolute path, or a relative path **width** --- This specifies the width to display the image. **height** -- This specifies the height to display the image. **alt** Alternate text. This specifies text to be used in case the browser/user agent can't render the image.

Frames: With frames, you can display more than one HTML document in the same browser window. Each HTML document is called a frame, and each frame is independent of the others.

The frameset (frameset1.html):

```
<html>
<head>
<title>Frameset
page<title> </head>
<frameset cols = "25%, *">
<frame src ="left.html" />
<frame src ="right.html" />
</frameset>
</html>
```

The left frame (left.html):

```
<html>
<body style="background-color:green">
<p>This is the left frame </p>
</body>
</html>
```

The right frame (right.html):

```
<html>
<body style="background- color:yellow">
<p>This is the right frame </p>
</body>
</html>
```

Forms

A form is an area that can contain form elements. Form elements are elements that allow the user to enter information . A form is defined using the <form></form> tags.

The Input Tag

This is the most commonly used tag within HTML forms . It allows you to specify various types of user input fields such as text, radio buttons, checkboxes etc.

Text

Text fields are used for when you want the user to type text or numbers into the form. <input type="text" >

Example :

```
<html>
<body>
<form action="MAILTO:g.sreenivasulu2@gmail.com" method="post"
enctype="text/plain"> Name: <input type="text" name="name" size="20"> <br>
e-mail <input type="text" name="mail" size="20"> <br>
Comments: <input type="text" name="comment" size="40"> <br>
<br> <input type="submit" value="Send">
<input type="reset"
value="Reset"> </form>
</body>
</html>
```

Radio Button

Radio Buttons are used when you want the user to select one of a limited number of choices. <form>

```
<input type="radio" name="sex" value="male" />
Male <br />
<input type="radio" name="sex" value="female" />
Female </form>
```

Checkboxes

Checkboxes are used when you want the user to select one or more options of a limited number of choices.

```
<form action="">
I have a bike:
<input type="checkbox" name="vehicle"
value="Bike"> <br />
I have a car:
<input type="checkbox" name="vehicle"
value="Car"> <br />
I have an airplane:
<input type="checkbox" name="vehicle"
value="Airplane"> </form>
```

```
<textarea rows="10" cols="30"> The cat was playing in the garden. </textarea>
```

```
<form>
Username: <input type="text" name="user"> <br>
Password: <input type="password"
name="password"> </form>
```

```
<select>
<option>California -- CA</option>
<option>Colorado -- CO</option>
<option selected="yes">Conneticut -- CN</option>
</select>
<select size="3">
<option>California -- CA</option>
<option>Colorado -- CO</option>
<option>Connecticut -- CN</option>
</select>
<select multiple="yes" size="3">
<option>California -- CA</option>
<option>Colorado -- CO</option>
<option>Connecticut -- CN</option>
</select>
```

CSS(Cascading Style Sheets)

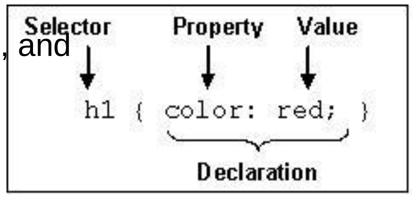
CSS is the acronym for: ‘Cascading Style Sheets’. CSS is an extension to basic HTML that allows you to style your web pages.

A typical CSS statement will SELECTOR { PROPERTY: VALUE } Property" is the CSS element you wish to manipulate and "VALUE" represents the value of the specified property.

Example

```
p {color: white; }
body {background-color: black; }
```

Cascading Style Sheets are three Types : external, internal, and inline.



External Style Sheet An external style sheet is ideal when the style is applied to many pages. With an external style sheet, you can change the look of an entire Web site by changing one file. Each page must link to the style sheet using the <link> tag. The <link> tag goes inside the head section.

```
-------- style.css -----------
h1 {
  font-size: 40px;
  height: 200px;
}
.warning {
  color: Red;
  font-weight: bold;
}
#footer { background-
  color: Gray;
```

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

Internal Style Sheet

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

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

Inline Styles

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

```
<p style="color: red; margin-left:
20px"> This is a paragraph
</p>
```

Class

Class is specified by including a period (.) before the selector name. The syntax for declaring a Class selector is as follows:

```
.navbar {
color:#0000FF;
}
<p class="navbar">This is a sample using a Class selector.</p>
```

ID

ID is specified by including a number sign (#) before the selector name. The syntax for declaring an ID selector is as follows:

```
#footer {
color:#FF00FF
;
}
```

To apply this style to the HTML, using the following code:

```
<p id="footer">This is a sample using an ID selector.</p>
```

The difference between ID and class is that an ID selector can be called only once in a document, while a class selector can be called multiple times in a document. The second difference is that ID can be called by Javascript's getElementByID function.

There is no hard rule on when to use ID and when to use Class. My suggestion is to use class as much as possible for maximum flexibility, with the only exception being when you want to use Javascript's getElementByID function, in which case you need use ID.

Class and ID names are both case sensitive. For example, .classone and .ClassOne are two different

classes.

DIV The div element is the generic block element. You can use div to apply styles to large sections of HTML, such as multiple paragraphs.

```
<div style="color:#00FF00">
<h3>This is a header</h3>
<p>This is a paragraph.</p>
</div>
```

SPAN The span element is the generic inline element. Used over a span.

```
<p> My friend has <span Style = "color:lightskyblue;font-weight:bol" >light blue</span> eyes
```

CSS Properties are divided into the below given logical groups

CSS Font Properties

The CSS font properties control all aspects of your text graphical representation.

- font
- font-family
- font-size
- font-style
- font-weight
- font-variant

CSS Text Properties

The CSS text properties control the spacing, alignment, decoration, and other miscellaneous aspects of the text.

- letter-spacing
- word-spacing
- text-decoration
- vertical-align
- text-transform
- text-align
- text-indent
- line-height

CSS Box Properties

The CSS box properties are used to define the spacing in and around HTML elements, their borders, and other positioning aspects. Border, Margin, and Padding all have four properties each: top, right, bottom, and left.

- Margin
- Padding
- Border
- Border-width
- Border-color
- Border-style
- Width

- Height
- Float
- Clear

CSS Color Properties

The CSS color property defines what color the text inside the specified HTML element will have h4 { color: red; } , h5 { color: #9000A1; } , h6 { color: rgb(0, 220, 98); }

CSS Background Properties

The CSS background properties control things like if the background is a single color or maybe an image

- Background
- Background Color
- Background Image
- Background Repeat
- Background Attachment
- Background Position

CSS Classification Properties

We think of the classification properties as having the list element and all the leftover elements that would not fit into any other category.

- Display
- Whitespace
- List Style
- List Style Type
- List Style Image
- List Style Position