What is HTML?

- HTML stands for Hyper Text Markup Language, which is the most widely used language on Web to develop web pages.

- HTML was created by Berners-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995.

- HTML 4.01 was a major version of HTML and it was published in late 1999.

- Though HTML 4.01 version is widely used but currently we are having HTML-5 version which is an extension to HTML 4.01, and this version was published in 2012.

- It is not case-sensitive language.

- The latest version of HTML is HTML 5

- HTML documents are also called as web pages. HTML pages can run on any browser.It is used to create Static Web pages.

- It is a global language i.e., it can be understood by all browsers.

Hypertext is text displayed on a computer or other electronic device with references to other text that the user can immediately access, usually by a mouse click or key press. Apart from text, hypertext may contain tables, images and other presentational elements. It is an easy-to-use and flexible format to share information over the Internet.
Markup Languages use sets of markup tags to characterize text elements within a document, which gives instructions to the Web browser on how the document should appear.

HTML Gives You Ability To:

- Publish documents online with text, headings, images, lists, tables, etc.
- Accessing online information or other web resources such as images, videos or other HTML document via hyperlinks.
- Create forms to collect user input like login information, feedback, comments or conducting transactions with remote servers, etc.
- Include videos, sound clips, flash movies, applications and other HTML document directly inside an HTML document.

HTML Document Overview

All HTML document seem to be plain text files. They contain no images,sounds, videos etc. but just plain text. However, they may contain links to images, sounds and videos. Note:The web browsers does not display the HTML tags, but uses the tags to interpret the content of the web pages.

HTML Example

```
<!DOCTYPE html>

<html>

<head></head>

<body>

<h1>My First Heading</h1>

<p>My first paragraph.</p>

</body>

</html>
```

Explainaton:

The DOCTYPE declaration defines the document type

The text between <html> and </html> describes the web page The text between <body> and </body> is the visible page content The text between <h1> and </h1> is

displayed as a heading

The text between and is displayed as a paragraph

The <!DOCTYPE html> declaration is the doctype for HTML5.

HTML Tags And Elements:

HTML is written in the form of HTML elements consisting of markup tags. These markup tags are the fundamental characteristic of HTML. Every markup tag is composed of a keyword, surrounded by angle brackets, like <html>, <head>, <body>, etc.

HTML tags normally come in pairs like <html> and </html>. The first tag in a pair is the opening tags the second tag is the closing tags. An opening tag and a closing tag are identical, except a "slash" (/) after the opening angle bracket of a closing tag, to tell the browser that the command has been completed. In between these tags you can add headings, paragraphs of text, tables, forms, images, videos etc. For example, a paragraph, which is represented by the pelement, would be written as:
This is a paragraph.

HTML Tags

HTML markup tags are usually called HTML tags

- HTML tags are keywords (tag names) surrounded by angle brackets like <html>
- HTML tags normally come in pairs like and
- The first tag in a pair is the start tag, the second tag is the end tag The end tag is written like the start tag, with a forward slash before the tag name

Start and end tags are also called opening tags and closing tags <tagname>content</ tagname>

HTML Elements

"HTML tags" and "HTML elements" are often used to describe the same thing.

But strictly speaking, an HTML element is everything between the start tag and the end tag, including the tags:

HTML Element:
This is a paragraph.

HTML Editors

Writing HTML

HTML code can be written in:

Notepad,Notepad++,Microsoft Visual Studio,Adobe Dreamweaver ,Microsoft Expression .Web Edit Plus,Text Pad,Notepad++ etc.

However, for learning HTML we recommend a text editor like Notepad (PC) or TextEdit (Mac). We believe using a simple text editor is a good way to learn HTML.

Follow the 4 steps below to create your first web page with Notepad.

Step 1: Start Notepad

To start Notepad go to: Notepad Application

Step 2: Edit Your HTML with Notepad

Type your HTML code into your Notepad:

<!DOCTYPE html>

<html>

<head></head>

<body>

<h1>My First Heading</h1>

My first paragraph.

</body>

</html>

Step 3: Save Your HTML

Select Save as.. in Notepad's file menu.

When you save an HTML file, you can use either the .htm or the .html file extension.

Step 4: Run the HTML in Your Browser

Start your web browser and open your html file double-click your HTML file.

Heading Tags

Any document starts with a heading. You can use different sizes for your headings. HTML also has six levels of headings, which use the elements<h1>, <h2>, <h3>, <h4>, <h5>, and <h6>. While displaying any heading, browser adds one line before and one line after that heading.

```
<!DOCTYPE html>
<html>
<head>
<title>Heading Example</title>
</head>
<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>
</body>
</html>
```

Paragraph Tag:

The tag offers a way to structure your text into different paragraphs. Each paragraph of text should go in between an opening and a closing tag as shown below in the example:

```
<!DOCTYPE html>
<html>
<head>
<title>Paragraph Example</title>
</head>
<body>
<p>Here is a first paragraph of text.</p>
<p>Here is a second paragraph of text.</p>
<p>Here is a third paragraph of text.</p>
</body>
</html>
```

LineBreak Tag:

Whenever you use the
 element, anything following it starts from the next line. This tag is an example of an empty element, where you do not need opening and closing tags, as there is nothing to go in between them.

The
 tag has a space between the characters br and the forward slash. If you omit this space, older browsers will have trouble rendering the line break, while if you miss the forward slash character and just use
 it is not valid in XHTML

```
<!DOCTYPE html>
<html>
<head>
<title>Line Break  Example</title>
</head>
<body>
<p>Hello<br />
```

You delivered your assignment ontime.
Thanks
Mahnaz</p>
</body>
</html>

Centering Content :

You can use tag to put any content in the center of the page or any table cell.

```
<!DOCTYPE html>
<html>
<head>
<title>Centring Content Example</title>
</head>
<body>
<p>This text is not in the center.</p>
<center>
<p>This text is in the center.</p>
</center>
</body>
</html>
```

Horizontal Lines:

Horizontal lines are used to visually break up sections of a document. The<hr> tag creates a line from the current position in the document to the right margin and breaks the line accordingly.

For example you may want to give a line between two paragraphs as in the given example below:

```
<!DOCTYPE html>
<html>
<head>
<title>Horizontal Line Example</title>
</head>
<body>
<p>This is paragraph one and should be on top</p>
<hr />
<p>This is paragraph two and should be at bottom</p>
```

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

Preserve Formatting:

Sometimes you want your text to follow the exact format of how it is written in the HTML document. In those cases, you can use the preformatted tag <pre>.

Any text between the opening <pre> tag and the closing </pre> tag will preserve the formatting of the source document.

```
<!DOCTYPE html>
<html>
<head>
<title>Preserve Formatting Example</title>
</head>
<body>
<pre>
function testFunction( strText ){
  alert (strText)
}
</pre>
</body>
</html>
```

NonBreaking Spaces:

In cases where you do not want the client browser to break text, you should use a nonbreaking space entity   instead of a normal space.

```
<!DOCTYPE html>
<html>
<head>
<title>Nonbreaking Spaces Example</title>
</head>
<body>
<p>Hello Every one  "Welcome WebDevelopment Course."</p>
</body>
</html>
```

Nested HTML Elements :

It is very much allowed to keep one HTML element inside another HTML element:

```
<!DOCTYPE html>
<html>
<head>
<title>Nested Elements Example</title>
</head>
<body>
<h1>This is <i>italic</i> heading</h1>
<p>This is <u>underlined</u> paragraph</p>
</body>
</html>
```

HTML Attributes :

An attribute is used to define the characteristics of an HTML element and is placed inside the element's opening tag. All attributes are made up of two parts: a name and a value:

- The name is the property you want to set. For example, the paragraph <p> element in the example carries an attribute whose name is align, which you can use to indicate the alignment of paragraph on the page.

- The value is what you want the value of the property to be set and always put within quotations. The below example shows three possible values of align attribute: left, center and right.

Attribute names and attribute values are case-insensitive. However, the World Wide Web Consortium (W3C) recommends lowercase attributes/attribute values in their HTML 4 recommendation.

```
<!DOCTYPE html>
<html>
<head>
<title>Align Attribute  Example</title>
</head>
<body>
<p align="left">This is left aligned</p>
<p align="center">This is center aligned</p>
```

```
<p align="right">This is right aligned</p>
</body>
</html>
```

Core Attributes:

The four core attributes that can be used on the majority of HTML elements (although not all) are:

- id
- title
- class
- style

Id Attribute :

The id attribute of an HTML tag can be used to uniquely identify any element within an HTML page. There are two primary reasons that you might want to use an id attribute on an element:

- If an element carries an id attribute as a unique identifier it is possible to identify just that element and its content.

- If you have two elements of the same name within a Web page (or style sheet), you can use the id attribute to distinguish between elements that have the same name.

```
<!DOCTYPE html>
<html>
<head>
<title>The id Attribute Example</title>
</head>
<body>
<h3 id="myheading">Heading Tag Example</h3>
</body>
</html>
```

Title Attribute:

The title attribute gives a suggested title for the element. They syntax for thetitle attribute is similar as explained for id attribute:

The behavior of this attribute will depend upon the element that carries it, although it is often displayed as a tooltip when cursor comes over the element or while the element is loading.

```
<!DOCTYPE html>
<html>
<head>
<title>The title Attribute Example</title>
</head>
<body>
<h3 title="Hello HTML!">Titled Heading Tag Example</h3>
</body>
</html>
```

Class Attribute:

The class attribute is used to associate an element with a style sheet, and specifies the class of element. You will learn more about the use of the class attribute when you will learn Cascading Style Sheet (CSS). So for now you can avoid it.

The value of the attribute may also be a space-separated list of class names. For example:

class="className1 className2 className3"

Style Attribute:

The style attribute allows you to specify Casecading Style Sheet (CSS) rules within the element.

```
<!DOCTYPE html>
<html>
<head>
<title>The style Attribute</title>
</head>
<body>
```

```
<p style="font-family:arial; color:#FF0000;">Some text...</p>
</body>
</html>
```

HTML Formatting:

If you use a word processor, you must be familiar with the ability to make text bold, italicized, or underlined; these are just three of the ten options available to indicate how text can appear in HTML and XHTML.

Bold Text

Anything that appears within ... element,

Italic Text

Anything that appears within ...

Underlined Text

Anything that appears within <u>...</u> element

Strike Text

Anything that appears within <strike>...</strike> element is displayed with strikethrough

Monospaced Font

The content of a <tt>...</tt> element is written in monospaced font. Most of the fonts are known as variable-width fonts because different letters are of different widths (for example, the letter 'm' is wider than the letter 'i'). In a monospaced font, however, each letter has the same width.

Superscript Text

The content of a ^{...} element is written in superscript; the font size used is the same size as the characters surrounding it but is displayed half a character's height above the other characters.

Subscript Text

The content of a _{...} element is written in subscript; the font size used is the same as the characters surrounding it, but is displayed half a character's height beneath the other characters.

Inserted Text

Anything that appears within <ins>...</ins> element

Deleted Text

Anything that appears within ~~...~~ element

Larger Text

The content of the <big>...</big> element is displayed one font size larger than the rest of the text surrounding

Smaller Text

The content of the <small>...</small> element is displayed one font size smaller than the rest of the text surrounding

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Formatting Example</title>
</head>
<body>
<p> Hello <b>Web Development</b> This is Srinivas .</p>
<p> Hello <i> Web Development</i> This is Srinivas .</p>
```

```
<p> Hello <u> Web Development</u> This is Srinivas .</p>
<p> Hello <strike> Web Development</strike> This is Srinivas .</p>
<p> Hello <tt> Web Development</tt> This is Srinivas .</p>
<p> Hello <sup> Web Development</sup> This is Srinivas .</p>
<p> Hello <sub> Web Development</sub> This is Srinivas .</p>
<p> Hello <ins> Web Development</ins> This is Srinivas .</p>
<p> Hello <del> Web Development</del> This is Srinivas .</p>
<p> Hello <big> Web Development</big> This is Srinivas .</p>
<p> Hello <small> Web Development</small> This is Srinivas .</p>
<p> Hello <mark> Web Development</mark> This is Srinivas .</p>
<p> Hello <abbr> Web Development</abbr> This is Srinivas .</p>
<p> Hello <bdo dir="rtl"> Web Development</bdo> This is Srinivas .</p>
<p> Hello <dfn> Web Development</dfn> This is Srinivas .</p>
<p> Hello <code> Web Development</code> This is Srinivas .</p>
<p> Hello <cite> Web Development</cite> This is Srinivas .</p>

</body>
</html>
```

Quoting Text

When you want to quote a passage from another source, you should put it in between <blockquote>...</blockquote> tags.

Text inside a <blockquote> element is usually indented from the left and right edges of the surrounding text, and sometimes uses an italicized font.

```
<!DOCTYPE html>
<html>
<head>
<title>Blockquote Example</title>
</head>
<body>
<p>The following description of XHTML is taken from the W3C Web site:</p>

<blockquote>XHTML 1.0 is the W3C's first Recommendation for XHTML, following on from earlier work on HTML 4.01, HTML 4.0, HTML 3.2 and HTML 2.0.</blockquote>
</body>
</html>
```

Short Quotations

The <q>...</q> element is used when you want to add a double quote within a

sentence.

```
<!DOCTYPE html>
<html>
<head>
<title>Double Quote Example</title>
</head>
<body>
<p>Amit is in Spain, <q>I think I am wrong</q>.</p>
</body>
</html>
```

Grouping Content

The and elements allow you to group together several elements to create sections or subsections of a page.

For example, you might want to put all of the footnotes on a page within a element to indicate that all of the elements within that element relate to the footnotes. You might then attach a style to this element so that they appear using a special set of style rules.

```
<!DOCTYPE html>
<html>
<head>
<title>Div Tag Example</title>
</head>
<body>
<div id="menu" align="middle" >
<a href="/index.htm">HOME</a> |
<a href="/about/contact_us.htm">CONTACT</a> |
<a href="/about/index.htm">ABOUT</a>
</div>
<div id="content" align="left" bgcolor="white">
<h5>Content Articles</h5>
<p>Actual content goes here.....</p>
<p>This is the example of <span style="color:green">span tag</span> and the
<span style="color:red">div tag</span> alongwith CSS</p>
</div>
</body>
```

</html>

HTML Meta Tags

Specifying Keywords

You can use <meta> tag to specify important keywords related to the document and later these keywords are used by the search engines while indexing your webpage for searching purpose.

Example

Following is an example where we are adding HTML, Meta Tags, Metadata as important keywords about the document.

```
<!DOCTYPE html>
<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
<meta name="revised" content="peers tech, 3/4/2016" />
<meta http-equiv="refresh" content="5; url=http://www.apple.com" />
<meta http-equiv="cookie" content="userid=xyz; expires=Wednesday, 08-Aug-16
23:59:59 GMT;" />
<meta name="author" content="Srinivas Gorantla" />
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>
```

HTML Comments

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

Comments are written like this:

Example

<!-- This is a comment -->

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

Conditional Comments

Conditional comments only work in Internet Explorer (IE) on Windows but they are ignored by other browsers. They are supported from Explorer 5 onwards, and you can use them to give conditional instructions to different versions of IE.

```
<!DOCTYPE html><html>
<head>
<title>Conditional Comments</title>
<!--[if IE 6]>
   Special instructions for IE 6 here
<![endif]-->
</head>
<body>
<p>Document content goes here.....</p>
</body>
</html>
```

Comment Tag example

There are few browsers that support <comment> tag to comment a part of HTML code.

```
<!DOCTYPE html><html>
<head>
<title>Using Comment Tag</title>
</head>
<body>
<p>This is <comment>not</comment> Internet Explorer.</p>
</body>
</html>
```

HTML Images

Images are very important to beautify as well as to depict many complex concepts in simple way on your web page. This example will take you through simple steps to use images in your web pages.

Insert Image

You can insert any image in your web page by using <img> tag. Following is the simple syntax to use this tag.


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

Set Image Width/Height

You can set image width and height based on your requirement using widthand height attributes. You can specify width and height of the image in terms of either pixels or percentage of its actual size.

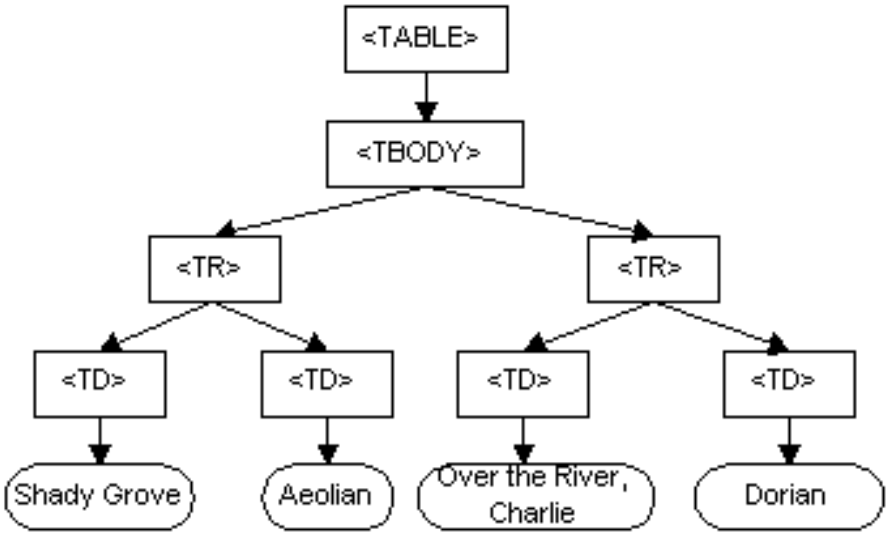
Set Image Border

By default image will have a border around it, you can specify border thickness in terms of pixels using border attribute. A thickness of 0 means, no border around the picture.

Set Image Alignement

By default image will align at the left side of the page, but you can use alignattribute to set it in the center or right.

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). td stands for "table data," and holds the con- tent of a data cell. A <td> tag can contain text, links, images, lists, forms, other tables, etc.

Table Example

<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>

HTML Tables and the Border Attribute

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

To display a table with borders, specify the border attribute:

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

</table>

HTML Table Headers

Header information in a table are defined with the <th> tag.

All major browsers display the text in the <th> element as bold and cen- tered.

<table border="1"> <tr>
<th>Header 1</th> <th>Header 2</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>

HTML Lists

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

- - An unordered list. This will list items using plain bullets.
- - An ordered list. This will use different schemes of numbers to list your

items.

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

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

The list items are marked with bullets (typically small black circles).

```
<ul>
<li>iPhone</li>
 <li>iPad</li>
 <li>iPod</li>
 </ul>
```

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

The list items are marked with numbers.

```
<ol>
<li>iPhone</li>
 <li>iPad</li>
 <li>iPod</li>
</ol>
```

An definition list starts with the <dl> tag. Each list item starts with the <dl> tag.

The list items are marked with numbers.

```
<dl>
<li>iPhone</li>
```

```
 <li>iPad</li>

 <li>iPod</li>

</dl>
```

Type Attribute

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

<ol type="1"> - Default-Case Numerals.
<ol type="I"> - Upper-Case Numerals.
<ol type="i"> - Lower-Case Numerals.
<ol type="a"> - Lower-Case Letters.
<ol type="A"> - Upper-Case Letters.

Start Attribute

You can use start attribute for <ol> tag to specify the starting point of numbering you need. Following are the possible options:

<ol type="1" start="4"> - Numerals starts with 4.
<ol type="I" start="4"> - Numerals starts with IV.
<ol type="i" start="4"> - Numerals starts with iv.
<ol type="a" start="4"> - Letters starts with d.
<ol type="A" start="4"> - Letters starts with D.

Text Links

A webpage can contain various links that take you directly to other pages and even specific parts of a given page. These links are known as hyperlinks.

Hyperlinks allow visitors to navigate between Web sites by clicking on words, phrases, and images. Thus you can create hyperlinks using text or images available on a webpage.

Linking Documents

A link is specified using HTML tag <a>. This tag is called anchor tag and anything between the opening <a> tag and the closing tag becomes part of the link and a user can click that part to reach to the linked document. Following is the simple syntax to use <a> tag.

Opens in New | Opens the linked document in a new window or tab.
Opens in Self | Opens the linked document in the same frame.
Opens in Parent | Opens the linked document in the parent frame.
Opens in Body Opens the linked document in the full body of the window.

Image Links

We have seen how to create hypertext link using text and we also learnt how to use images in our webpages. Now we will learn how to use images to create hyperlinks.

Example

It's simple to use an image as hyperlink. We just need to use an image inside hyperlink at the place of text as shown below:

```
<!DOCTYPE html>
<html>
<head>
<title>Image Hyperlink Example</title>
</head>
<body>
<p>Click following link</p>
<a href="http://www.apple.com" target="_self">
   <img src="/images/logo.png" alt="Apple" border="0"/>
</a>
</body>
</html>
```

Email Tag

HTML <a> tag provides you option to specifiy an email address to send an email. While using <a> tag as an email tag, you will use mailto:email address along with href attribute. Following is the syntax of using mailtoinstead of using http.

Send Email

You can specify a default email subject and email body alongwith your email address. Following is the example to use default subject and body.

Send Feedback

Iframes

inline frame with HTML tag <iframe>. The <iframe> tag is not somehow related to <frameset> tag, instead, it can appear anywhere in your document. The <iframe> tag defines a rectangular region within the document in which the browser can display a separate document, including scrollbars and borders.

The src attribute is used to specify the URL of the document that occupies the inline frame.

Example

Following is the example to show how to use the <iframe>:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Iframes</title>
</head>
<body>
<p>Document content goes here...</p>
<iframe src="/html/menu.htm" width="555" height="200">
  Sorry your browser does not support inline frames.
```

```
</iframe>
<p>Document content also go here...</p>
</body>
</html>
```

Background With Colors

The bgcolor attribute is used to control the background of an HTML element, specifically page body and table backgrounds. Following is the syntax to use bgcolor attribute with any HTML tag.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Background Colors</title>
</head>
<body>
<!-- Format 1 - Use color name -->
<table bgcolor="yellow" width="100%">
<tr><td>
This background is yellow
</td></tr>
</table>
<!-- Format 2 - Use hex value -->
<table bgcolor="#6666FF" width="100%">
<tr><td>
This background is sky blue
</td></tr>
</table>
<!-- Format 3 - Use color value in RGB terms -->
<table bgcolor="rgb(255,0,255)"  width="100%">
<tr><td>
This background is green
</td></tr>
</table>
</body>
</html>
```

Background With Images

The background attribute can also be used to control the background of an HTML

elmement, specifically page body and table backgrounds. You can specify an image to set background of your HTML page or table. Following is the syntax to use background attribute with any HTML tag

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Background Images</title>
</head>
<body>
<!-- Set table background -->
<table background="/images/html.gif" width="100%" height="100">
<tr><td>
This background is filled up with HTML image.
</td></tr>
</table>
</body>
</html>
```

Colors

Colours are very important to give a good look and feel to your website. You can specify colors on page level using <body> tag or you can set colors for individual tags using bgcolor attribute.

The <body> tag has following attributes which can be used to set different colors:

- bgcolor - sets a color for the background of the page.
- text - sets a color for the body text.
- alink - sets a color for active links or selected links.
- link - sets a color for linked text.
- vlink - sets a color for visited links - that is, for linked text that you have already clicked on.

Color Coding Methods:

There are following three different methods to set colors in your web page:

- Color names - You can specify color names directly like green, blue or red.

- Hex codes - A six-digit code representing the amount of red, green, and blue that makes up the color.

- Color decimal or percentage values - This value is specified using the rgb( ) property.

Example

Here are the examples to set background of an HTML tag by color name

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Colors by Name</title>
</head>
<body text="blue" bgcolor="green">
<p>Use different color names for for body and table and see the result.</p>
<table bgcolor="black">
<tr>
<td>
<font color="white">This text will appear white on black background.</font>
</td>
</tr>
</table>
</body>
</html>
```

Colors-HexCodes

A hexadecimal is a 6 digit representation of a color. The first two digits(RR) represent a red value, the next two are a green value(GG), and the last are the blue value(BB).

A hexadecimal value can be taken from any graphics software like Adobe Photoshop,

Paintshop Pro or MS Paint.

Each hexadecimal code will be preceded by a pound or hash sign #. Following is a list of few colors using hexadecimal notation.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Colors by Hex</title>
</head>
<body text="#0000FF" bgcolor="#00FF00">
<p>Use different color hexa for for body and table and see the result.</p>
<table bgcolor="#000000">
<tr>
<td>
<font color="#FFFFFF">This text will appear white on black background.</font>
</td>
</tr>
</table>
</body>
</html>
```

Colors-RGB Values

This color value is specified using the rgb( ) property. This property takes three values, one each for red, green, and blue. The value can be an integer between 0 and 255 or a percentage.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Colors by RGB code</title>
</head>
<body text="rgb(0,0,255)" bgcolor="rgb(0,255,0)">
<p>Use different color code for for body and table and see the result.</p>
<table bgcolor="rgb(0,0,0)">
<tr>
<td>
<font color="rgb(255,255,255)">This text will appear white on black background.</
font>
</td>
```

```
</tr>
</table>
</body>
</html>
```

Safe -Colors

Here is the list of 216 colors which are supposed to be safest and computer independent colors. These colors very from hexa code 000000 to FFFFFF and they will be supported by all the computers having 256 color palette.

Fonts

Fonts play very important role in making a website more user friendly and increasing content readability. Font face and color depends entirely on the computer and browser that is being used to view your page but you can use HTML tag to add style, size, and color to the text on your website. You can use a <basefont> tag to set all of your text to the same size, face, and color.

The font tag is having three attributes called size, color, and face to customize your fonts. To change any of the font attributes at any time within your webpage, simply use the tag. The text that follows will remain changed until you close with the tag. You can change one or all of the font attributes within one <font> tag.

Set Font Size

You can set content font size using size attribute. The range of accepted values is from 1(smallest) to 7(largest). The default size of a font is 3.

```
<!DOCTYPE html>
<html>
<head>
<title>Setting Font Size</title>
</head>
<body>
<font size="1">Font size="1"</font><br />
```

```
<font size="2">Font size="2"</font><br />
<font size="3">Font size="3"</font><br />
<font size="4">Font size="4"</font><br />
<font size="5">Font size="5"</font><br />
<font size="6">Font size="6"</font><br />
<font size="7">Font size="7"</font>
</body>
</html>
```

Relative Font Size

You can specify how many sizes larger or how many sizes smaller than the preset font size should be. You can specify it like <font size="+n"> or <font size="-n">

```
<!DOCTYPE html>
<html>
<head>
<title>Relative Font Size</title>
</head>
<body>
<font size="-1">Font size="-1"</font><br />
<font size="+1">Font size="+1"</font><br />
<font size="+2">Font size="+2"</font><br />
<font size="+3">Font size="+3"</font><br />
<font size="+4">Font size="+4"</font>
</body>
</html>
```

Setting Font Face

You can set font face using face attribute but be aware that if the user viewing the page doesn't have the font installed, they will not be able to see it. Instead user will see the default font face applicable to the user's computer.

```
<!DOCTYPE html>
<html>
<head>
<title>Font Face</title>
</head>
<body>
<font face="Times New Roman" size="5">Times New Roman</font><br />
```

```
<font face="Verdana" size="5">Verdana</font><br />
<font face="Comic sans MS" size="5">Comic Sans MS</font><br />
<font face="WildWest" size="5">WildWest</font><br />
<font face="Bedrock" size="5">Bedrock</font><br />
</body>
</html>
```

Setting Font Color

You can set any font color you like using color attribute. You can specify the color that you want by either the color name or hexadecimal code for that color.

```
<!DOCTYPE html>
<html>
<head>
<title>Setting Font Color</title>
</head>
<body>
<font color="#FF00FF">This text is in pink</font><br />
<font color="red">This text is red</font>
</body>
</html>
```

Forms

HTML Forms are required when you want to collect some data from the site visitor. For example during user registration you would like to collect information such as name, email address, credit card, etc.

A form will take input from the site visitor and then will post it to a back-end application such as CGI, ASP Script or PHP script etc. The back-end application will perform required processing on the passed data based on defined business logic inside the application.

There are various form elements available like text fields, textarea fields, drop-down menus, radio buttons, checkboxes, etc.

The HTML <form> tag is used to create an HTML form and it has following syntax:

```
<form action="Script URL" method="GET|POST">
   form elements like input, textarea etc.
</form>
```

Form Controls

There are different types of form controls that you can use to collect data using HTML form:

- Text Input Controls
- Checkboxes Controls
- Radio Box Controls
- Select Box Controls
- File Select boxes
- Hidden Controls
- Clickable Buttons
- Submit and Reset Button

Form Input Controls

There are three types of text input used on forms:

- Single-line text input controls - This control is used for items that require only one line of user input, such as search boxes or names. They are created using HTML <input> tag.
- Password input controls - This is also a single-line text input but it masks the

character as soon as a user enters it. They are also created using HTMl <input> tag.

- Multi-line text input controls - This is used when the user is required to give details that may be longer than a single sentence. Multi-line input controls are created using HTML <textarea> tag.

Form Single Line Text Input Controls

This control is used for items that require only one line of user input, such as search boxes or names. They are created using HTML <input> tag.

Example

Here is a basic example of a single-line text input used to take first name and last name:

```
<!DOCTYPE html>
<html>
<head>
<title>Text Input Control</title>
</head>
<body>
<form >
First name:  <input type="text" name="first_name" />
<br>
Last name:  <input type="text" name="last_name" />
</form>
</body>
</html>
```

Form Password Input Controls

This is also a single-line text input but it masks the character as soon as a user enters it. They are also created using HTML <input> tag but type attribute is set to password.

Example

Here is a basic example of a single-line password input used to take user password:

```
<!DOCTYPE html>
<html>
<head>
<title>Password Input Control</title>
</head>
<body>
<form >
User ID :  <input type="text" name="user_id" />
<br>
Password:  <input type="password" name="password" />
</form>
</body>
</html>
```

Form Multiple Line Text Input Controls

This is used when the user is required to give details that may be longer than a single sentence. Multi-line input controls are created using HTML <textarea> tag.

Example

Here is a basic example of a multi-line text input used to take item description:

```
<!DOCTYPE html>
<html>
<head>
<title>Multiple-Line Input Control</title>
</head>
<body>
<form>
Description : <br />
<textarea rows="5" cols="50" name="description">
Enter description here...
</textarea>
</form>
</body>
</html>
```

CheckBox Control

Checkboxes are used when more than one option is required to be selected. They are also created using HTML <input> tag but type attribute is set tocheckbox.

Example

Here is an example HTML code for a form with two checkboxes:

```
<!DOCTYPE html>
<html>
<head>
<title>Checkbox Control</title>
</head>
<body>
<form>
<input type="checkbox" name="maths" value="on"> Maths
<input type="checkbox" name="physics" value="on"> Physics
</form>
</body>
</html>
```

Radio Button Control

Radio buttons are used when out of many options, just one option is required to be selected. They are also created using HTML <input> tag but type attribute is set to radio.

Example

Here is example HTML code for a form with two radio buttons:

```
<!DOCTYPE html>
<html>
<head>
<title>Radio Box Control</title>
</head>
<body>
<form>
<input type="radio" name="subject" value="maths"> Maths
<input type="radio" name="subject" value="physics"> Physics
</form>
</body>
```

</html>

Select Box Control

A select box, also called drop down box which provides option to list down various options in the form of drop down list, from where a user can select one or more options.

Example

Here is example HTML code for a form with one drop down box

```
<!DOCTYPE html>
<html>
<head>
<title>Select Box Control</title>
</head>
<body>
<form>
<select name="dropdown">
<option value="Maths" selected>Maths</option>
<option value="Physics">Physics</option>
</select>
</form>
</body>
</html>
```

File Upload Control

If you want to allow a user to upload a file to your web site, you will need to use a file upload box, also known as a file select box. This is also created using the <input> element but type attribute is set to file.

Example

Here is example HTML code for a form with one file upload box:

```
<!DOCTYPE html>
<html>
<head>
<title>File Upload Box</title>
```

```
</head>
<body>
<form>
<input type="file" name="fileupload" accept="image/*" />
</form>
</body>
</html>
```

Button Control

There are various ways in HTML to create clickable buttons. You can also create a clickable button using <input> tag by setting its type attribute tobutton. The type attribute can take the following values:

```
<!DOCTYPE html>
<html>
<head>
<title>File Upload Box</title>
</head>
<body>
<form>
<input type="submit" name="submit" value="Submit" />
<input type="reset" name="reset"  value="Reset" />
<input type="button" name="ok" value="OK"  />
<input type="image" name="imagebutton" src="/html/images/logo.png" />
</form>
</body>
</html>
```

Marquee

An HTML marquee is a scrolling piece of text displayed either horizontally across or vertically down your webpage depending on the settings. This is created by using HTML <marquees> tag.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML marquee Tag</title>
</head>
<body>
<marquee>This is basic example of marquee</marquee>
```

```
<marquee width="50%">This example will take only 50% width</marquee>
<marquee direction="right">This text will scroll from left to right</marquee>
<marquee direction="up">This text will scroll from bottom to up</marquee>
</body>
</html>
```

<link> tag

The HTML <link> tag is used to specify relationships between the current document and external resource. Following is an example to link an external style sheet file available in css sub-directory within web root:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML link Tag Example</title>
<base href="http://www.google.com/" />
<link rel="stylesheet" type="text/css" href="/css/style.css">
</head>
<body>
<p>Hello, World!</p>
</body>
</html>
```

<style> tag

The HTML <style> tag is used to specify style sheet for the current HTML document. Following is an example to define few style sheet rules inside <style> tag:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML style Tag Example</title>
<base href="http://www.google.com/" />
<style type="text/css">
.myclass{
  background-color: #aaa;
  padding: 10px;
}
</style>
</head>
<body>
```

```
<p class="myclass">Hello, World!</p>
</body>
</html>
```

<script> tag

The HTML <script> tag is used to include either external script file or to define internal script for the HTML document. Following is an example where we are using Javascript to define a simple Javascript function:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML script Tag Example</title>
<base href="http://www.google.com/" />
<script type="text/javascript">
function Hello(){
   alert("Hello, World");
}
</script>
</head>
<body>
<input type="button" onclick="Hello();" name="ok" value="OK"  />
</body>
</html>
```

Style Sheet With HTML

Cascading Style Sheets (CSS) describe how documents are presented on screens, in print, or perhaps how they are pronounced. W3C has actively promoted the use of style sheets on the Web since the Consortium was founded in 1994.

Cascading Style Sheets (CSS) provide easy and effective alternatives to specify various attributes for the HTML tags. Using CSS, you can specify a number of style properties for a given HTML element. Each property has a name and a value, separated by a colon (:). Each property declaration is separated by a semi-colon (;).

Example

First let's consider an example of HTML document which makes use of <font> tag and associated attributes to specify text color and font size:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML CSS</title>
</head>
<body>
<p style="color:green;font-size:24px;">Hello, World!</p>
</body>
</html>
```

You can use CSS in three ways in your HTML document:

- External Style Sheet - Define style sheet rules in a separate .css file and then include that file in your HTML document using HTML <link> tag.

- Internal Style Sheet - Define style sheet rules in header section of the HTML document using <style> tag.

- Inline Style Sheet - Define style sheet rules directly along-with the HTML elements using style attribute.

Let's see all the three cases one by one with the help of suitable examples.

External Style Sheet

If you need to use your style sheet to various pages, then its always recommended to define a common style sheet in a separate file. A cascading style sheet file will have extension as .css and it will be included in HTML files using <link> tag.

Example

Consider we define a style sheet file style.css which has following rules:

```
.red{
   color: red;
```

```
}
.thick{
  font-size:20px;
}
.green{
  color:green;
}
```

Here we defined three CSS rules which will be applicable to three different classes defined for the HTML tags. I suggest you should not bother about how these rules are being defined because you will learn them while studying CSS. Now let's make use of the above external CSS file in our following HTML document:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML External CSS</title>
<link rel="stylesheet" type="text/css" href="/html/style.css">
</head>
<body>
<p class="red">This is red</p>

<p class="thick">This is thick</p>

<p class="green">This is green</p>

<p class="thick green">This is thick and green</p>
</body>
</html>
```

Internal Style Sheet

If you want to apply Style Sheet rules to a single document only then you can include those rules in header section of the HTML document using <style> tag.

Rules defined in internal style sheet overrides the rules defined in an external CSS file.

Example

Let's re-write above example once again, but here we will write style sheet rules in the

same HTML document using <style> tag:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Internal CSS</title>
<style type="text/css">
.red{
   color: red;
}
.thick{
   font-size:20px;
}
.green{
   color:green;
}
</style>
</head>
<body>
<p class="red">This is red</p>
<p class="thick">This is thick</p>
<p class="green">This is green</p>
<p class="thick green">This is thick and green</p>
</body>
</html>
```

Inline Style Sheet

You can apply style sheet rules directly to any HTML element using styleattribute of the relevant tag. This should be done only when you are interested to make a particular change in any HTML element only.

Rules defined inline with the element overrides the rules defined in an external CSS file as well as the rules defined in <style> element.

Example

Let's re-write above example once again, but here we will write style sheet rules along with the HTML elements using style attribute of those elements.

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Inline CSS</title>
</head>
<body>
<p style="color:red;">This is red</p>
<p style="font-size:20px;">This is thick</p>
<p style="color:green;">This is green</p>
<p style="color:green;font-size:20px;">This is thick and green</p>
</body>
</html>
```

Javascript

A script is a small piece of program that can add interactivity to your website. For example, a script could generate a pop-up alert box message, or provide a dropdown menu. This script could be written using Javascript or VBScript.

You can write various small functions, called event handlers using any of the scripting language and then you can trigger those functions using HTML attributes.

Now a days only Javascript and associated frameworks are being used by most of the web developers, VBScript is not even supported by various major browsers.

You can keep Javascript code in a separate file and then include it whereever it's needed, or you can define functionality inside HTML document itself. Let's see both the cases one by one with suitable examples.

External Javascript

If you are going to define a functionality which will be used in various HTML documents then it's better to keep that functionality in a separate Javascript file and then include that file in your HTML documents. A Javascript file will have extension as .js and it will be included in HTML files using <script> tag.

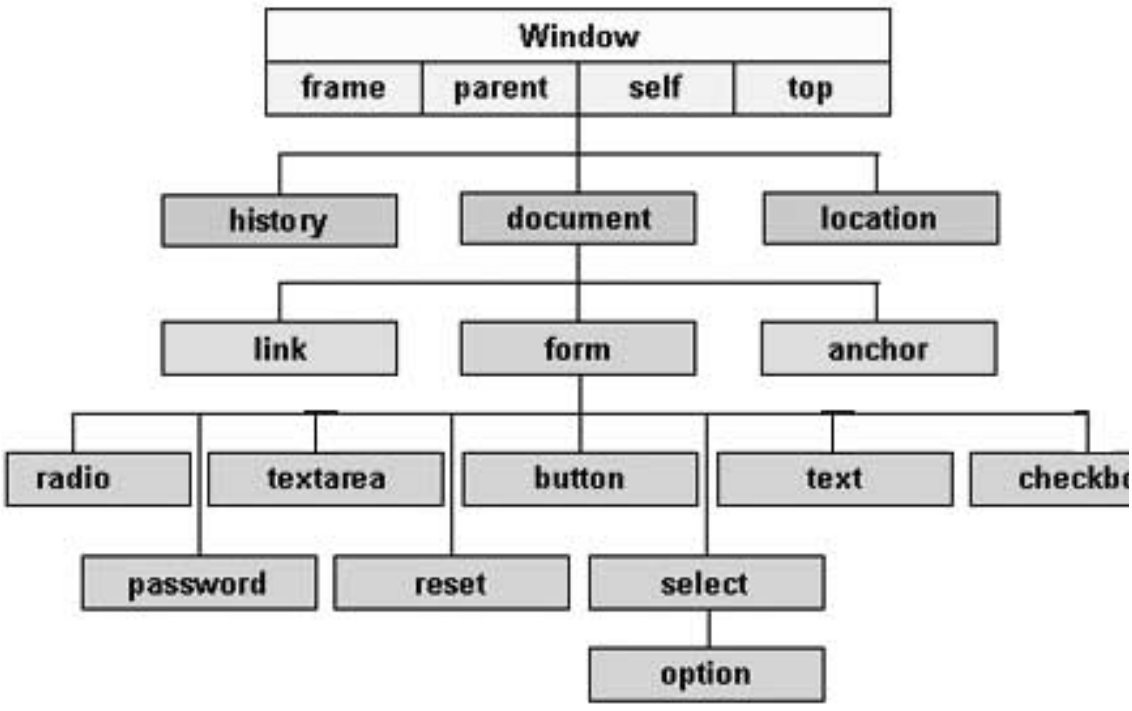
Example

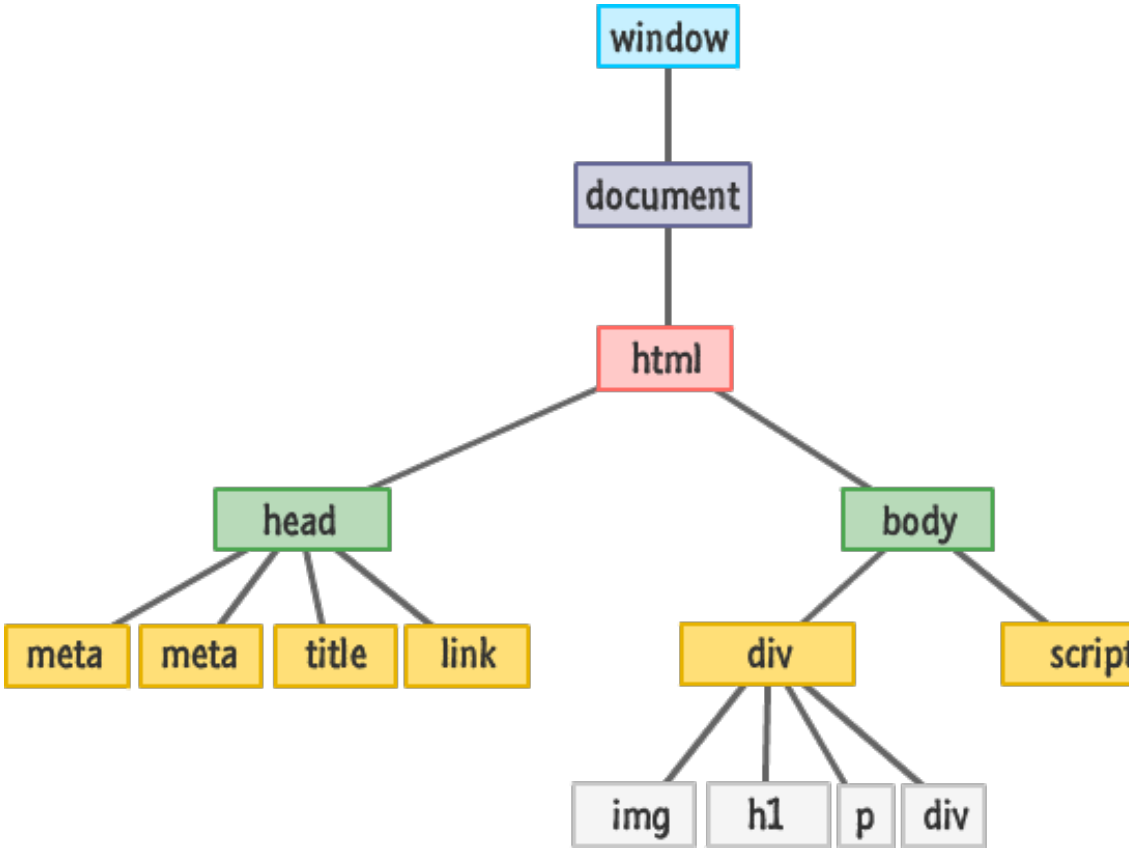
Consider we define a small function using Javascript in script.js which has following code:

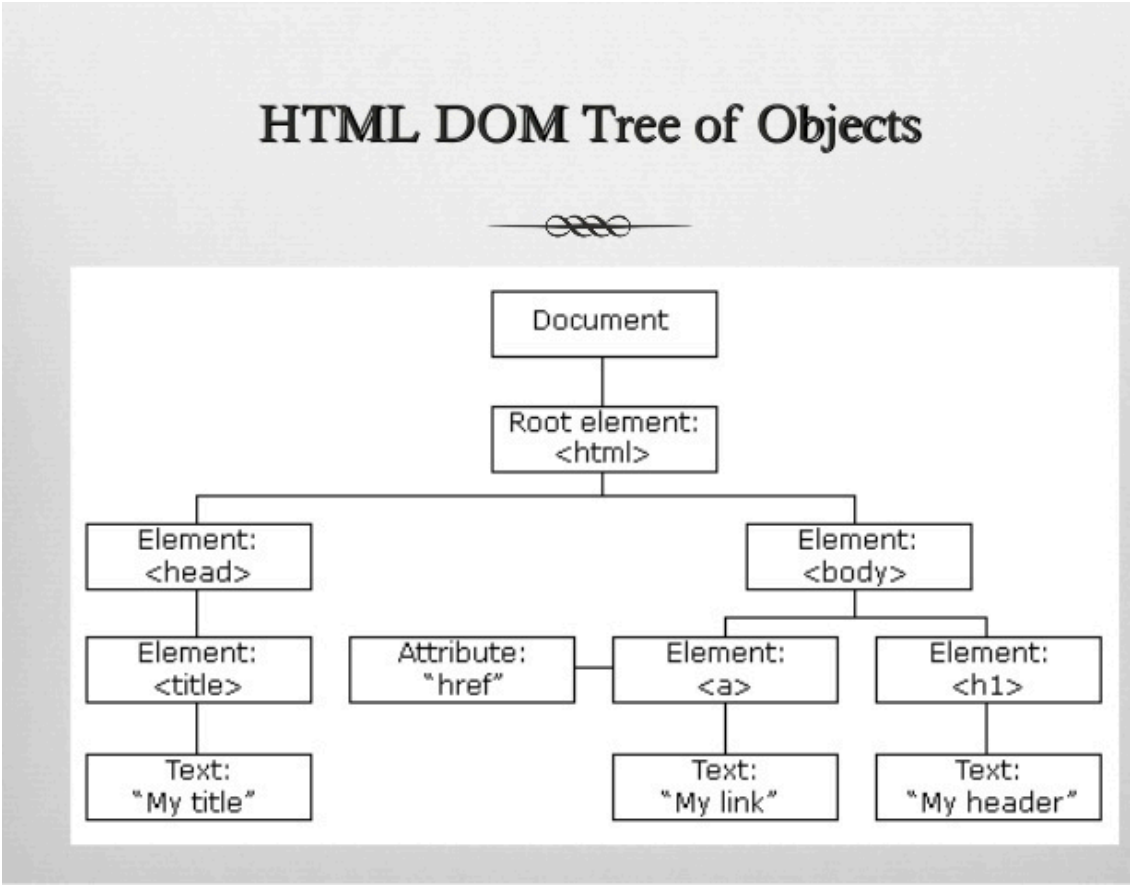
```
function Hello()

{
   alert("Hello, World");
}
```

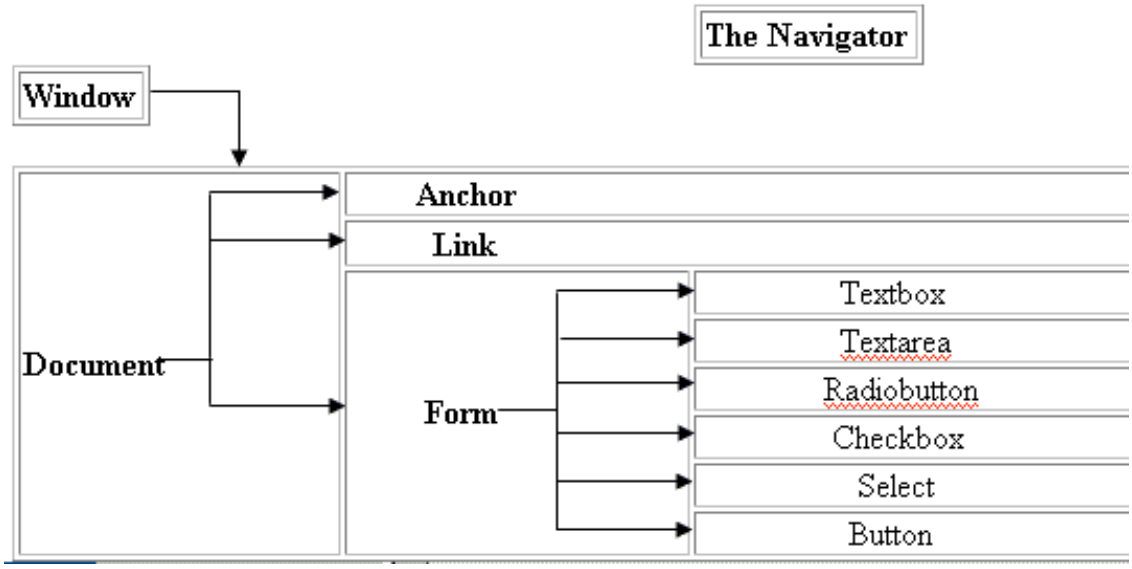
```
<!DOCTYPE html>
<html>
<head>
<title>Javascript External Script</title>
<script src="/html/script.js" type="text/javascript"/></script>
</head>
<body>
<input type="button" onclick="Hello();" name="ok" value="Click Me"  />
</body>
</html>
```

HTML DOM (Document Object Model)




window
document
html
head
body
meta
meta
title
link
div
script
img
h1
p
div


HTML DOM Tree of Objects
Document
Root element:
<html>
Element:
<head>
Element:
<body>
Element:
<title>
Attribute:
"href"
Element:
<a>
Element:
<h1>
Text:
"My title"
Text:
"My link"
Text:
"My header"


The Navigator
Window
Anchor
Link
Document
Form
Textbox
Textarea
Radiobutton
Checkbox
Select
Button