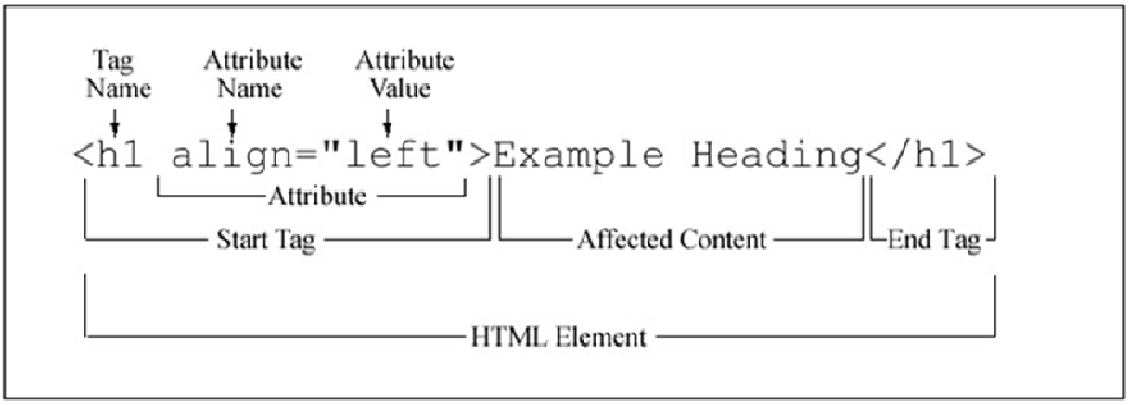
M.K.Institute of Computer Studies,Bharuch

What is HTML?

HTML is the standard markup language for creating Web pages.

- HTML stands for Hyper Text Markup Language
- HTML describes the structure of Web pages using markup
- HTML elements are the building blocks of HTML pages
- 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
- A complete overview of the structure of HTML elements is shown here:



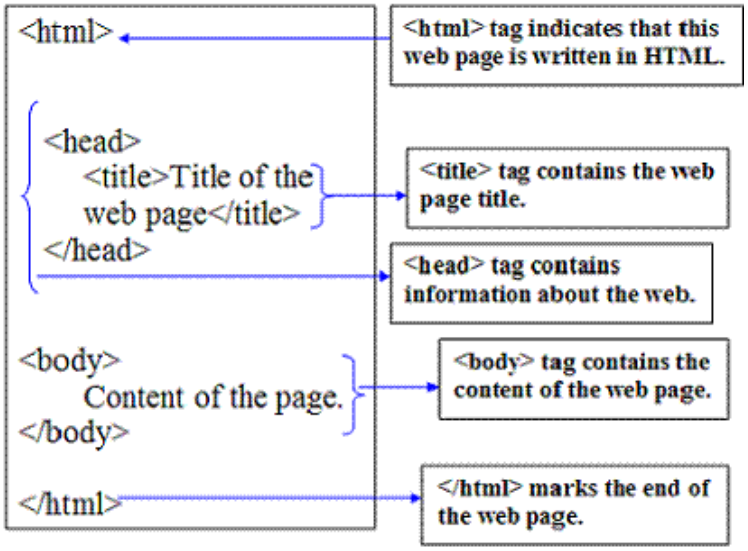
Structure of an HTML document

- In HTML document has two main parts:

    1. **head**: The head element contains title and meta data of a web document.
    2. **body**: The body element contains the information that you want to display on a web page.

- To make your web pages compatible with HTML 4, you need to add a document type declaration (DTD) before the HTML element. In a web page, the first tag (specifically, <html>) indicates the markup language that is being used for the document. The <head> tag contains information about the web page. Lastly, the content appears in the <body> tag.

M.K.Institute of Computer Studies,Bharuch



- The <head> element is a container for metadata (data about data) and is placed between the <html> tag and the <body> tag.
- HTML metadata is data about the HTML document. Metadata is not displayed.
- Metadata typically define the document title, character set, styles, links, scripts, and other meta information.
- The following tags describe metadata: <title>, <style>, <meta>, <link>,<script>, and <base>.

Tag	Description
<head>	Defines information about the document
<title>	Defines the title of a document
<base>	Defines a default address or a default target for all links on a page
<link>	Defines the relationship between a document and an external resource

<meta>	Defines metadata about an HTML document
<script>	Defines a client-side script
<style>	Defines style information for a document

M.K.Institute of Computer Studies,Bharuch

<Body>

- When you create a Web page, you can change several things in the body of your document by adding attributes to the <body> tag.
- <body bgcolor="black" text="red" link="yellow" alink="orange" vlink="white" background="image.gif">
- We can use as many or as few of these attributes as we wish. The options we don't use will be set to the Web browser's default values.
- Let us see the brief explanation of each attribute:

Attribute	**Value**	**Description**
background	Image.gif	Specifies a background image for a document. If you use this attribute, the background image will take the place of any background color you may have specified. If you don't use a background image, the browser will use your background color or its default background color.
bgcolor	color	Specifies the background color of a document
text	color	Specifies the color of the text in a document
alink	color	Specifies the color of an active link, which is a link that has just been clicked on by a user's mouse in a document
link	color	Specifies the color of all the unvisited links in a document
vlink	color	Specifies the color of visited links in a document
Bottommargin	Number /%	Specifies the bottom margin in a document
Topmargin	Number /%	Specifies the top margin in a document
Leftmargin	Number /%	Specifies the left margin in a document
Rightmargin	Number /%	Specifies the right margin in a document

The Rules of HTML

- **HTML Is Not Case Sensitive**

  These markup examples

  <B>Go boldly!</B>

  <B>Go boldly!

  Go boldly!</B>

  Go boldly!

- **HTML Attribute Values May Be Case Sensitive**

  One interesting aspect of HTML's case sensitivity is that although HTML element names and attribute names are not case sensitive, we can't assume everything is case insensitive. For example, consider <img SRC="test.gif"> and <img src="test.gif">.

- **HTML Is Sensitive to a Single White Space Character**

  Browsers collapse white space between characters down to a space. This includes all tabs, line breaks, and carriage returns. Note that in some situations, HTML does treat white space characters differently

- **HTML Follows a Content Model**

  Both HTML and XHTML support a strict content model that says that certain elements are supposed to occur only within other elements. For example, markup like this,

  ```
  <ul>
  <p>What a simple way to break the content model!</p>
  </ul>
  ```

  The <ul> tag is only supposed to contain <li> tags. The <p> tag is not really appropriate in this context.

- **Elements Should Have Close Tags Unless Empty**

  Under traditional HTML, some elements have optional close tags. For example, both of the paragraphs here are allowed, although the second one is better:

  ```
  <p> This isn't closed.
  <p>This is.</p>
  ```

A few tags, such as the horizontal rule <hr> or line break <br>, do not have close tags because they do not enclose any content. These are considered empty elements.

- **Unused Elements May Minimize**

  Sometimes tags may not appear to have effect in a document. Consider, for example, the tag, which specifies a paragraph. As a block tag it induces a return by default, but when used repeatedly, like so,

   it does not produce numerous blank lines because the browser minimizes the empty p elements.

- **Elements Should Nest**

  A simple rule states that tags should nest, not cross, thus is in error as tags cross whereas is not since tags nest.

- **Attributes Should Be Quoted**

  <img src=robot.gif height=10 width=10 alt=robot>

- **Browsers Ignore Unknown Attributes and Elements**

  For better or worse, browsers will ignore unknown elements and attributes, so
  <bogus> this text will display on screen</bogus> will be ignored.

M.K.Institute of Computer Studies,Bharuch

HTML Text Formatting Tags

HTML also defines special **elements** for defining text with a special **meaning**.

1. **HTML and Elements**
   The HTML element defines bold text, without any extra importance.

   Example: This text is bold
   Output: **This text is bold.**

   The HTML element defines strong text, with added semantic "strong" importance.

   Example: This text is strong
   Output: **This text is strong.**

2. **HTML and Elements**
   The HTML element defines italic text, without any extra importance.

   Example: This text is italic
   Output: *This text is italic.*

   The HTML element defines emphasized text, with added semantic importance.

   Example: This text is emphasized
   Output: *This text is emphasized*

   Note: Browsers display as , and as . However, there is a difference in the meaning of these tags: and defines bold and italic text, but and means that the text is "important".

3. **HTML <small> Element**
   The HTML <small> element defines smaller text:

   Example: <h2>HTML <small>Small</small> Formatting</h2>
   Output: **HTML Small Formatting**

4. **HTML <mark> Element**
   The HTML <mark> element defines marked or highlighted text:

Example: <h2>HTML <mark>Marked</mark> Formatting</h2>
Output: **HTML Marked Formatting**

5. **HTML <del> Element**
   The HTML <del> element defines ~~deleted~~ (removed) text.

   Example: My favorite color is ~~blue~~ red.
   Output: My favorite color is ~~blue~~ red.

6. **HTML <ins> Element**
   The HTML <ins> element defines inserted (added) text.

   Example: My favorite <ins>color</ins> is red.
   Output: My favorite color is red.

7. **HTML <sub> Element**
   The HTML <sub> element defines subscripted text.

   Example: This is _{subscripted} text.
   Output: This is $_{\text{subscripted}}$ text.

8. **HTML <sup> Element**
   The HTML <sup> element defines superscripted text.

   Example: This is ^{superscripted} text.
   Output: This is $^{\text{superscripted}}$ text.

Tag	Description
<b>	Defines bold text
	Defines emphasized text
<i>	Defines italic text
<small>	Defines smaller text
	Defines important text
<sub>	Defines subscripted text
<sup>	Defines superscripted text

<ins>	Defines inserted text
<del>	Defines deleted text
<mark>	Defines marked/highlighted text

HTML Headings

HTML heading tags <h1> - <h6>:
HTML supports six different levels of headings. The highest level header format is <h1> and the lowest is <h6>.
All the styles appear in Boldface and the size of the heading depends on the level chosen,i.e <h1> to <h6>
By default heading is left aligned, we can change it using align attribute of heading

Attribute	Value	Description
align	left center right	Not supported in HTML5. Specifies the alignment of a heading

Example:

```
<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>
```

Output:

This is heading 1
This is heading 2
This is heading 3
This is heading 4
This is heading 5
This is heading 6

HTML Comment

- The contents of HTML comments are not displayed within a browser window. Comments are denoted by a start value of <!-- and an end value of -->.
- Comments can be many lines long.
  Example:
  <!–
  This is a Comment. Comment will not be displayed
  -->

HTML Line Break

 tag is used to insert a line break in html content.

Images in HTML

- In HTML, images are defined with the **<img>** tag.
- The **<img>** tag is empty, it contains attributes only, and does not have a closing tag.
- The **src** attribute specifies the URL (web address) of the image.
- The **alt** attribute provides an alternate text for an image, if the user for some reason cannot view it.
- If a browser cannot find an image, it will display the value of the alt attribute.
- The **width** and **height** attributes always defines the width and height of the image in pixels.
- The **width**, **height**, and **style** attributes are valid in HTML5.

Attribute	Value	Description
align	top bottom middle left right	Not supported in HTML5. Specifies the alignment of an image according to surrounding elements
alt	text	Specifies an alternate text for an image
border	pixels	Not supported in HTML5. Specifies the width of the border around an image

height	pixels	Specifies the height of an image
hspace	pixels	Not supported in HTML5. Specifies the whitespace on left and right side of an image
src	URL	Specifies the URL of an image
vspace	pixels	Not supported in HTML5. Specifies the whitespace on top and bottom of an image
width	pixels	Specifies the width of an image

Example:

```
<mg src="mkics_logo.jpeg" alt="MKICS" height="42"width="42">
```

Output:

Links and bookmark

- 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.
- 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.
- The text enclosed by an a element specifies a "hot spot" to activate the hyperlink.
- Anchor content can include text, images, or a mixture of the two.
- For linking purposes, the <a> tag requires one attribute: href. The href attribute is set to the URL of the target resource, which basically is the address of the document to link to
- **Syntax**

```
<a href="url">link text</a>
```

Example:

```
<a href="https://www.mkics.in/">MKICS a BCA College</a>
```

HTML Link Colors

- By default, a link will appear like this (in all browsers):
  - An unvisited link is underlined and blue
  - A visited link is underlined and purple
  - An active link is underlined and red
- HTML authors can override these default link colors with changes to the link, alink, and vlink attributes of the body element.
- Otherwise You can change the default colors, by using CSS as shown below:

Example:

```
<style>
a:link {
  color: green;
  background-color: transparent;
  text-decoration: none;
}
a:visited {
                                                    color: pink;
```

```
                                                    background-color: transparent;
                                                        text-decoration: none;
        }
        a:hover {
                                                                color: red;
                                                    background-color: transparent;
                                                        text-decoration: underline;
        }
        a:active {
                                                                color: yellow;
                                                    background-color: transparent;
                                                        text-decoration: underline;
        }
        </style>
```

The target Attribute

The target attribute specifies where to open the linked document.
The target attribute can have one of the following values:

- _blank - Opens the linked document in a new window or tab
- _self - Opens the linked document in the same window/tab as it was clicked (this is default)
- _parent - Opens the linked document in the parent frame
- _top - Opens the linked document in the full body of the window
- *framename* - Opens the linked document in a named frame

Link Titles

The title attribute specifies extra information about an element. The information is most often shown as a tooltip text when the mouse moves over the element.

HTML Links to a Page Section- a Bookmark

- HTML bookmarks are used to allow readers to jump to specific parts of a Web page.
- Bookmarks can be useful if your webpage is very long.
- To make a bookmark, you must first create the bookmark, and then add a link to it.
- When the link is clicked, the page will scroll to the location with the bookmark.

Example:

- First, create a bookmark with the id attribute:

```
<h2 id="C4">Chapter 4</h2>
```

- Then, add a link to the bookmark ("Jump to Chapter 4"), from within the same page:

```
<a href="#C4">Jump to Chapter 4</a>
```

- Or, add a link to the bookmark ("Jump to Chapter 4"), from another page:

```
<a href="html_demo.html#C4">Jump to Chapter 4</a>
```

Attributes :

- Use the <a> element to define a link
- Use the href attribute to define the link address
- Use the target attribute to define where to open the linked document
- Use the <img> element (inside <a>) to use an image as a link
- Use the id attribute (id="*value*") to define bookmarks in a page

Image as a Link

- <a> tags can enclose text and other content, including images. When an anchor encloses an image, the image becomes clickable, thus providing the basic mechanism for a graphic button.

Example:

```
<a href="about.html">
<img src="about.gif" alt="About Button" height="55" width="55" />
</a>
```

Tables In HTML

- A table is an orderly arrangement of data distributed across a grid of rows and columns.
- Web page tables also are commonly used to structure a page for layout.
- In its simplest form, a table places information inside the cells formed by dividing a rectangle into rows and columns.
- Most cells contain data; some cells, usually on the table's top or side, contain headings.
- HTML represent a basic table using four elements.
  - table tag pair, <table> … </table>, contains an optional caption element
  - one or more rows, <tr> … </tr>.
  - Each row contains cells holding a heading, <th> … </th>, or data, <td>.. </td>.

- **Example:**

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

```
<head>
  <title>HTML Tables</title>
</head>

<body>
  <table border = "1">
    <tr>
       <th>Fruit</td>
      <th>Color</td>
    </tr>
    <tr>
      <td>Apple</td>
      <td>Red</td>
    </tr>
    <tr>
      <td>Orange</td>
      <td>Orange</td>
    </tr>
  </table>

  </body>
</html>
```

- **Output:**

Fruit	Color
Apple	Red
Orange	Orange

Cellpadding and Cellspacing Attributes

- There are two attributes called *cellpadding* and *cellspacing* which will be used to adjust the white space in your table cells.
- The cellspacing attribute defines space between table cells, while cellpadding represents the distance between cell borders and the content within a cell.

Colspan and Rowspan Attributes

- **Colspan** attribute is used to merge two or more columns into a single column.
- Similarly **rowspan** is used to merge two or more rows.

Tables Backgrounds

We can set table background using one of the following two ways −

- **bgcolor** attribute − You can set background color for whole table or just for one cell.
- **background** attribute − You can set background image for whole table or just for one cell.
- We can also set border color also using **bordercolor** attribute.

Note − The bgcolor, background, and bordercolor attributes deprecated in HTML5.

Table Height and Width

- We can set a table width and height using **width** and **height** attributes. You can specify table width or height in terms of pixels or in terms of percentage of available screen area.

Table Caption

- The **caption** tag will serve as a title or explanation for the table and it shows up at the top of the table.
- We can also add an attribute **align** in **<caption>** to set it to the bottom of table.
- This tag is deprecated in newer version of HTML/XHTML.

Cell Alignment

- Cells defined by <td> or <th> are generally aligned horizontally by setting the align attribute to left, right, or center with left being the default. It is also possible to justify their contents by setting the attribute to justify.
- The contents of cells can also be aligned vertically by setting valign on <th> or <td> tags to top, middle, bottom or baseline.

Table Header, Body, and Footer

- Tables can be divided into three portions − a header, a body, and a foot.
- The head and foot are rather similar to headers and footers in a word-processed document that remain the same for every page, while the body is the main content holder of the table.
- The three elements for separating the head, body, and foot of a table are −
    - <thead> − to create a separate table header.
    - <tbody> − to indicate the main body of the table.
    - <tfoot> − to create a separate table footer.

- A table may contain several <tbody> elements to indicate different pages or groups of data.
- But it is notable that <thead> and <tfoot> tags should appear before <tbody>.

- **Example:**

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML Table Colspan/Rowspan</title>
  </head>
  <body>
    <table border = "1" cellpadding=”5” cellspacing=”5” bordercolor=”red” bgcolor=”yellow” width=”500”
           height=”200”>
      <caption> Student Result</caption>
       <tr>
         <th rowspan=”2”>ROLLNO</th>
         <th rowspan=”2”>NAME</th>
         <th colspan=”3”>SUBJECTS</th>
       </tr>
       <tr>
         <th>Subject1</td>
         <th>Subject2</td>
         <th>Subject3</td>
       </tr>
       <tr>
         <td> 1</td>
         <td> Ajay</td>
         <td> 100</td>
         <td> 94</td>
         <td> 91</td>
      </tr>
       <tr>
         <td>2</td>
         <td> Vijay</td>
         <td> 80</td>
         <td> 88</td>
         <td> 75</td>
      </tr>
    </table>
  </body>
</html>
```

- **Output:**

Student Result

ROLLNO	NAME	SUBJECTS		
		Subject1	Subject2	Subject3
1	Ajay	100	94	91
2	Vijay	80	88	75

Forms In HTML

- 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 Attributes:**

Attribute	Description
action	Specifies an address (url) where to submit the form (default: the submitting page).
enctype	Specifies the encoding of the submitted data (default: is url-encoded). Possible values are − **application/x-www-form-urlencoded** − This is the standard method most forms use in simple scenarios. **mutlipart/form-data** − This is used when you want to upload binary data in the form of files like image, word file etc.
method	Specifies the HTTP method used when submitting the form GET/POST(default: GET).
name	Specifies a name used to identify the form (for DOM usage: document.forms.name).
target	Specifies the target of the address in the action attribute. It takes values like

_blank, _self, _parent etc. (default: _self)

M.K.Institute of Computer Studies,Bharuch

GET & POST METHOD OF FORM SUBMISSION:

The default method when submitting form data is GET.

When GET is used, the submitted form data will be **visible in the page address field as shown below**:
/action_page.php?firstname=Mickey&lastname=Mouse

- Appends form-data into the URL in name/value pairs
- The length of a URL is limited (about 3000 characters)
- Never use GET to send sensitive data! (will be visible in the URL)
- Useful for form submissions where a user wants to bookmark the result
- GET is better for non-secure data, like query strings in Google

Always use POST if the form data contains sensitive or personal information.
The POST method does not display the submitted form data in the page address field.

- POST has no size limitations, and can be used to send large amounts of data.
- Form submissions with POST cannot be bookmarked.

Differentiate GET & POST Method

	GET	**POST**
BACK button/Reload	Harmless	Data will be re-submitted (the browser should alert the user that the data are about to be re-submitted)
Bookmarked	Can be bookmarked	Cannot be bookmarked
Cached	Can be cached	Not cached
History	Parameters remain in browser history	Parameters are not saved in browser history
Restrictions on data length	Yes, when sending data, the GET method adds the data to the URL; and the length of a URL is limited (maximum URL length is 2048 characters)	No restrictions
Restrictions on data type	Only ASCII characters allowed	No restrictions. Binary data is also allowed

Security	GET is less secure compared to POST because data sent is part of the URL Never use GET when sending passwords or other sensitive information!	POST is a little safer than GET because the parameters are not stored in browser history or in web server logs
Visibility	Data is visible to everyone in the URL	Data is not displayed in the URL

HTML 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

Text 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.

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.
- Following is the list of attributes for <input> tag for creating text field.

Attribute	Description
type	Indicates the type of input control and for text input control it will be set to **text**.
name	Used to give a name to the control which is sent to the server to be recognized and get the value.
value	This can be used to provide an initial value inside the control.
size	Allows to specify the width of the text-input control in terms of characters.
maxlength	Allows to specify the maximum number of characters a user can enter into the text box.

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.
- Following is the list of attributes for <input> tag for creating password field.

Attribute	Description
type	Indicates the type of input control and for text input control it will be set to **password**.
name	Used to give a name to the control which is sent to the server to be recognized and get the value.
value	This can be used to provide an initial value inside the control.
size	Allows to specify the width of the text-input control in terms of characters.
maxlength	Allows to specify the maximum number of characters a user can enter into the text box.

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.
- Following is the list of attributes for <textarea> tag

Attribute	Description
name	Used to give a name to the control which is sent to the server to be recognized and get the value.
rows	Indicates the number of rows of text area box.
cols	Indicates the number of columns of text area box.

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 to **checkbox**.
- Following is the list of attributes for <checkbox> tag

Attribute	Description
type	Indicates the type of input control and for text input control it will be set to **checkbox**.
name	Used to give a name to the control which is sent to the server to be recognized and get the value.
value	The value that will be used if the checkbox is selected..
checked	Set to *checked* if you want to select it by default.

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**.
- Following is the list of attributes for radio buttons

Attribute	Description
type	Indicates the type of input control and for text input control it will be set to **radio**.

name	Used to give a name to the control which is sent to the server to be recognized and get the value.
value	The value that will be used if the radio box is selected.
checked	Set to *checked* if you want to select it by default.

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.
- The tag should contain one or more occurrences of the option element. Each <option> tag specifies a menu choice.
- Following is the list of attributes for <select> tag

Attribute	Description
name	Used to give a name to the control which is sent to the server to be recognized and get the value.
size	This can be used to present a scrolling list box.
multiple	If set to "multiple" then allows a user to select multiple items from the menu.

- Following is the list of important attributes of <option> tag –

Attribute	Description
value	The value that will be used if an option in the select box is selected.
selected	Specifies that this option should be the initially selected value when the page loads..
label	An alternative way of labeling options.

File Upload Box

- 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.
- Following is the list of important attributes of file upload box:

Attribute	Description
name	Used to give a name to the control which is sent to the server to be recognized and get the value.
accept	Specifies the types of files that the server accepts

Button Controls

- 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 to **button**.
- Following is the list of important attributes of button

Attribute	Description
type	Can be either submit, reset, button or clickable button used as image background of.
name	Used to give a name to the control which is sent to the server to be recognized and get the value.
value	Value transmitted to the server and the text wording on the button.

- The type attribute can take the following values:

Value of type attribute	Description
submit	This creates a button that automatically submits a form.
reset	This creates a button that automatically resets form controls to their initial values.
button	This creates a button that is used to trigger a client-side

	script when the user clicks that button.
image	This creates a clickable button but we can use an image as background of the button.

Hidden Form Controls

- Hidden form controls are used to hide data inside the page which later on can be pushed to the server. This control hides inside the code and does not appear on the actual page.

HTML5 Input Types

HTML5 added several new input types:

- color
- date
- email
- url
- number
- range

Input type	Description
color	Used for input fields that should contain a color. Depending on browser support, a color picker can show up in the input field.
date	Used for input fields that should contain a date. Depending on browser support, a date picker can show up in the input field. We can also use the min and max attributes to add restrictions to dates
email	Used for input fields that should contain an e-mail address. Depending on browser support, the e-mail address can be automatically validated when submitted.
number	Defines a **numeric** input field. You can also set restrictions on what numbers are accepted.
range	Defines a control for entering a number whose exact value is not important (like a slider control).

	Default range is 0 to 100. We can still set restrictions on what numbers are accepted with the min, max, and step attributes
url	Used for input fields that should contain a URL address. Depending on browser support, the url field can be automatically validated when submitted.

Input Restrictions

- Here is a list of some common input restrictions (some are new in HTML5):

Attribute	**Description**
disabled	Specifies that an input field should be disabled
max	Specifies the maximum value for an input field (new in html5)
maxlength	Specifies the maximum number of character for an input field
min	Specifies the minimum value for an input field (new in html5)
pattern	Specifies a regular expression to check the input value against (new in html5)
readonly	Specifies that an input field is read only (cannot be changed)
required	Specifies that an input field is required (must be filled out) (new in html5)
size	Specifies the width (in characters) of an input field
step	Specifies the legal number intervals for an input field (new in html5)
value	Specifies the default value for an input field

M.K.Institute of Computer Studies,Bharuch

Difference between HTML and HTML5

HTML stands for *Hyper Text Markup Language*. It is used to design web pages using a markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. A markup language is used to define the text document within tag which defines the structure of web pages. This language is used to annotate (at the note for computer) text so that a machine can understand it and manipulate text accordingly. Most of the markup (e.g. HTML) languages are human readable. The language uses tags to define what manipulation has to be done on the text. It is used for structuring and presenting the content on the web pages. HTML5 is the fifth version of HTML. Many elements are removed or modified from HTML5.

There are many differences between HTML and HTML5 which are discussed below:

HTML	**HTML5**
It didn't support audio and video without the use of flash player support.	It supports audio and video controls with the use of <audio> and <video> tags.
It uses cookies to store temporary data.	It uses SQL databases and application cache to store offline data.
Does not allow JavaScript to run in browser.	Allows JavaScript to run in background. This is possible due to JS Web worker API in HTML5.
Vector graphics is possible in HTML with the help of various technologies such as VML, Silver-light, Flash, etc.	Vector graphics is additionally an integral a part of HTML5 like SVG and canvas.
It does not allow drag and drop effects.	It allows drag and drop effects.
Not possible to draw shapes like circle, rectangle, triangle etc.	HTML5 allows to draw shapes like circle, rectangle, triangle etc.
It works with all old browsers.	It supported by all new

	browser like Firefox, Mozilla, Chrome, Safari, etc.
Older version of HTML are less mobile-friendly.	HTML5 language is more mobile-friendly.
Doctype declaration is too long and complicated.	Doctype declaration is quite simple and easy.

Elements like nav, header were not present.	New element for web structure like nav, header, footer etc.
Character encoding is long and complicated.	Character encoding is simple and easy.
It is almost impossible to get true GeoLocation of user with the help of browser.	One can track the GeoLocation of a user easily by using JS GeoLocation API.
It can not handle inaccurate syntax.	It is capable of handling inaccurate syntax.
Attributes like charset, async and ping are absent in HTML.	Attributes of charset, async and ping are a part of HTML 5.

There are many HTML elements which have been modified or removed from HTML5. Some of them are listed below:

ELEMENT	**IN HTML5**
<applet>	Changed to <object>
<acronym>	Changed to <abbr>
<dir>	Changed to
<frameset>	Removed
<frame>	Removed
<noframes>	Removed
<strike>	No new tag. CSS is used for this
<big>	No new tag. CSS is used for this
<basefont>	No new tag. CSS is used for this

<font>	No new tag. CSS is used for this
	No new tag. CSS is used for this
<tt>	No new tag. CSS is used for this

Many new elements are added in HTML5 like nav, audio, figcaption, progress, command, time, datalist, video, figure, meter, data, section, time, aside, canvas, summary, rp, rt, details, wbr, header, footer, keygen, embed, article, hgroup, bdi, mark, output, source, track, section, ruby and many more.

M.K.Institute of Computer Studies,Bharuch

QUESTIONS FROM UNIT I

	Short Questions:
1)	What is difference between cell spacing and cell padding with example
2)	Write merits and demerits of frame
3)	Name the tags written within <HEAD> tag.
4)	Name any two text formatting commands.
5)	Explain anchor tag
6)	Explain Structure of HTML document
7)	Explain frameset tag attributes
8)	Explain <IMG>
9)	What is significance of method attribute of form
10)	How are bookmarks created?
11)	What is image map?
	Long Questions:
1)	Differentiate HTML4.0 and HTML 5.0
2)	Various text formatting tags
3)	Various block formatting tags
4)	Lists in HTML
5)	Links and bookmarks in HTML
6)	Frames and frameset
7)	Tables in detail
8)	Audio and Video in HTML
9)	Form elements with example
10)	Input elements like color, range, date, number, url and email

Prepared By: Meghna Vithlani(Asst. Prof, MKICS, Bharuch)