ABESIT
CRC-Training

UI Design using HTML, CSS, BootStrap

ABESIT
CRC-Training

Introduction to **Web page**

- A **web page** or **webpage** is a document, commonly written in HTML (hypertext markup language) viewed in an Internet browser.
- A web page can be accessed by entering a URL (uniform resource locator) address into a browser's address bar.
- A web page may contain text, graphics, and hyperlinks to other web pages and files.
- A web page provides information to viewers, including pictures or videos to help illustrate important topics.
- A web page may also be used as a method to sell products or services to viewers.
- Multiple web pages make up a website, like "**https://www.abesit.in/**" website.

ABESIT
CRC-Training

What is **Browser?**

- A browser is an **application program** that provides a way to look at and interact with all the information on the World Wide Web.
- This includes Web pages, videos and images.
- The word "**browser**" originated prior to the Web as a generic term for user interfaces that let you browse (navigate through and read) text files online.
- Many people will use web browsers today for access to the internet and is seen almost as a necessity in how many navigate their daily life.
- A Web browser is a client program that uses HTTP (Hypertext Transfer Protocol) to make requests of Web servers throughout the Internet on behalf of the browser user.
- Most browsers support e-mail and the File Transfer Protocol (FTP), but a Web browser is not required for those Internet protocols and more specialized client programs are more popular.

ABESIT
CRC-Training

Why we need a **WebServer**?

- Web servers are the focal point of **web hosting**. In fact, web hosting is simply a service in which you remotely lease a server for your website's use.
- Every web server and computer that is connected to the Internet is assigned an IP address that identifies that device on the network.
- When you visit a website, a request is sent out from your IP address to the IP address of the web server. The web server then responds by sending data to the requesting IP address, and this is the process through which all data is transferred to your computer from a website.
- Without web servers, the Internet we know and love would simply vanish. Web servers play a crucial role in how the Internet operates.
- Web servers are the **gateway** between the average individual and the world wide web, they are the backbone of the Internet, providing access to the vast amounts of information and resources available online.

Role of **HTML** in a **Browser**

- HTML (Hypertext Markup Language) is the standard markup language used to create web pages. The role of HTML in a website is to provide the **structure** and **content** of a webpage, including text, images, videos, and other multimedia elements. It defines the various elements that make up a webpage, such as headings, paragraphs, lists, tables, and links.
- HTML code is used to create the visual structure of a webpage, which is then styled and formatted using Cascading Style Sheets (CSS) and JavaScript. HTML tags are used to indicate the different elements of a webpage, such as a title, headings, paragraphs, images, and links.
- In summary, the role of HTML in a website is to define the structure and content of a webpage, which is then styled and formatted using CSS, and made interactive using JavaScript. HTML is the **backbone** of the web and essential for creating a functional and interactive website.

Static and **Dynamic** web page

- Webpages differ in the way they display content and facilitate user interaction. There are two main types of web pages:
  - Static
  - Dynamic
- **Static webpages** remain unchanged for every user.
- On the other hand, **dynamic webpages** support user interaction and may display different content for every user.

Static and **Dynamic** web page

Static webpages

- A static web page consists of fixed content and structure. It does not change according to a user's action, such as clicking a button. Hence, the webpage will be the same for every user.
- Content included in a static web page includes the following:
  - Images
  - Audio and video
  - Static forms
- Static webpages are typically created using HTML, CSS, and JavaScript. They're stored on the web server. Any alteration to the page needs to be done manually by editing the HTML source code. The client first sends an HTTP request to the server. The server then replies with an HTTP response and the requested web page without any additional processing.

Static and **Dynamic** web page

Dynamic webpages

- A dynamic webpage is less rigid in its structure and content. It supports user interaction, and the display may change accordingly.
- Content included in a dynamic web page includes the following:
  - Database
  - Real-time data, such as weather forecast
  - Chat applications
- Dynamic webpages depend on both client-side and server-side programming, such as using HTML and CSS alongside JavaScript or PHP. While the final display page is still an HTML document, the server may introduce additional processing in response to the sent HTTP request. It does this by forwarding the HTTP request to an application program, which executes and produces an HTML output.

Editor

- An HTML editor is a piece of software for creating and editing HTML code. It can be a stand-alone software dedicated to code writing and editing or a part of an **IDE** (Integrated Development Environment).
- An HTML editor provides more advanced features and is specifically designed for developers to create web pages more efficiently. It ensures every string of code is clean and works properly.
- The most common features of a good HTML editor are:
  - Syntax highlighting.
  - Auto-completion.
  - Error detection.
  - Search and replace.
  - FTP integration.
  - Code folding.
- **Ex:** Notepad, Notepad ++, Sublime Text, Visual Studio Code (VS Code) etc.

What is HTML5?

- HTML5 is the 5th version or update of HTML. This is the most advanced HTML programming.
- In this version, many tags were added to make the designer's life easier. A basic website contains 3 types of codes.
  - HTML
  - CSS
  - Javascript
- HTML is developed so that no other software or browser plugin is required to design a complete website from scratch.
- With HTML5, we can develop huge applications, animations, and other complicated applications that run on the browser. It finds applications in developing complex web applications designed for various purposes, including entertainment, online streaming, video players, audio players, and more.

History of HTML5?

- **1991–1999:** HTML was created by web legend Tim Berners-Lee in 1991, and HTML versions 1–4 were developed throughout the 1990s by W3C.
- **2000:** W3C recommends XHTML 1.0 – an XML-based markup language that mirrors/extends HTML.
- **2004:** Development of HTML is closed by W3C, who instead decide to focus on XHTML. WHATWG is formed to develop HTML further, with the aim of reflecting the modern dynamic web, while keeping backwards compatibility with existing HTML code.
- **2004–2006:** WHATWG gains support from major web browser developers. In 2006, W3C also announced its support for the project.
- **2008:** The first public draft of HTML5 is released by WHATWG.
- **2012:** W3C and WHATWG decide to separate development of HTML5. W3C would work on a definitive standard of HTML5, while WHATWG would pursue development of a 'living standard' – a continual evolution with ongoing improvements.
- **2014:** The official HTML5 release date, according to W3C recommendations.

Philosophy of HTML5?

- **Three Key Principles of HTML5**

1. **Don't Break the Web**
   - "Don't break the Web" means that a standard shouldn't introduce changes that make other people's web pages stop working. This rarely happens.
2. **Pave the Cowpaths**
   - A cowpath is the rough, heavily trodden track that gets people from one point to another. A cowpath exists because it's being used. It might not be the best possible way to move around, but at some point it was the most practical working solution.
3. **Be Practical**
   - This principle is simple: Changes should have a practical purpose. And the more demanding the change, the bigger the payoff needs to be. A good clue is to look at what web pages are already doing—or trying to do.

Future of HTML5?

- HTML5 is the future of web development. It provides a more powerful and flexible way to develop web applications and has empowered business owners and entrepreneurs to provide better customer experiences.
- In the years to come, you can expect more advances in HTML5 that will continue to make the web a more powerful and user-friendly place.
- Five big reasons why HTML5 represents the future:

1. Improved semantics
2. Improved performance
3. Offline support
4. Audio and video support
5. Storage and databases

Introduction to HTML

- Hypertext Markup Language
- Language of web page
- Books ← written ← English
- Web page ← written ← **HTML**
- Web pages are text files.
- The key to **hypertext** is the use of **hyperlinks**, which allow you to jump from one topic to another.

Definitions

- W W W – World Wide Web.
- HTML – **HyperText Markup Language** – The Language of Web Pages on the World Wide Web.

  HTML is a text formatting language.
- URL – Uniform Resource Locator.
- Browser – A software program which is used to show web pages.
- **"Normal text"** surrounded by bracketed *tags* that tell browsers how to display web pages.
- **Pages** end with ".htm" or ".html".

What is HTML?

- HTML describes the **content** and **format** of web pages using **tags**.

  Ex. Title Tag: *<title>A title </title>*
- It's the job of the web browser to interpret tags and display the content accordingly.

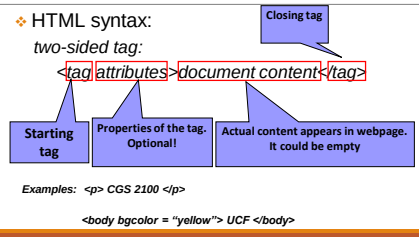
HTML Syntax

- An HTML file contains both **formatting tags** and **content**
- **Document content** is what we see on the webpage.
- **Tags** are the HTML codes that control the appearance of the document content.

HTML Syntax

- HTML syntax:

two-sided tag:



Examples: * CGS 2100 *

<body bgcolor = "yellow"> UCF </body>

HTML Syntax

- HTML syntax:

one-sided tag:

<tag />

e.g. **Breaking line tag:** **
**

Horizontal line tag: **<hr/>**

Structure of the web page

- Starting with the tag <html>...</html>

Everything about the web page should be enclosed here

<html>
.....
</html>

Structure of the web page

- Inside the <html></html> tag
  - Each web page has a **head** part described in <head></head> tag:

The title of the web page should be put here

<html>
<head>
<title> MY FIRST WEB PAGE </title>
</head>
</html>

ABESIT
CRC-Training

Structure of the web page

- Inside the <html></html> tag
  - Each web page has a **body** part described in <body></body> tag:

```
<html>
   <head>
      <title> MY FIRST WEB PAGE </title>
   </head>
    <body>
        This is a sample HTML file.
   </body>
</html>
```

The content of the whole web page should be put here

ABESIT
CRC-Training

ABESIT
CRC-Training

Setting Document Properties

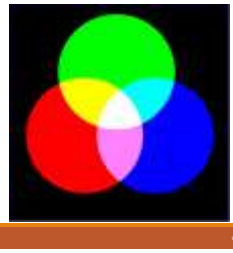
- Document properties are controlled by attributes of the BODY element. For example, there are color settings for the background color of the page, the document's text and different states of links.

Color Codes

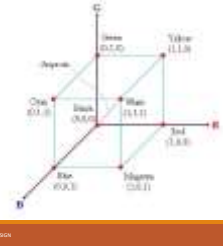
- Colors are set using "RGB" color codes, which are, represented as hexadecimal values. Each 2-digit section of the code represents the amount, in sequence, of red, green or blue that forms the color. For example, a RGB value with 00 as the first two digits has no red in the color.

ABESIT
CRC-Training

Main Colours



RGB Colour Model



Color Codes

1. WHITE	1. #FFFFFF
2. BLACK	2. #000000
3. RED	3. #FF0000
4. GREEN	4. #00FF00
5. BLUE	5. #0000FF
6. MAGENTA	6. #FF00FF
7. CYAN	7. #00FFFF
8. YELLOW	8. #FFFF00
9. AQUAMARINE	9. #70DB93
10. BAKER'S CHOCOLATE	10. #5C3317
11. VIOLET	11. #9F5F9F
12. BRASS	12. #B5A642
13. COPPER	13. #B87333
14. PINK	14. #FF6EC7
15. ORANGE	15. #FF7F00

The Body Element

- The BODY element of a web page is an important element in regards to the page's appearance. Here are the attributes of the **BODY** tag to control all the levels:

  TEXT="#RRGGBB" to change the color of **all the text** on the page (**full page text color.**)

- This element contains information about the page's background color, the background image, as well as the text and link colors.

Background Color

- It is very common to see web pages with their background color set to white or some other colors.
- To set your document's background color, you need to edit the <BODY> element by adding the BGCOLOR attribute. The following example will display a document with a white background color:

```
<BODY BGCOLOR="#FFFFFF"></BODY>
```


TEXT Color

- The TEXT attribute is used to control the color of all the normal text in the document. The default color for text is black. The TEXT attribute would be added as follows:

```
<BODY BGCOLOR="#FFFFFF" TEXT="#FF0000"></BODY>
```

- In this example the document's page color is white and the text would be red.

Using Image Background

- The BODY element also gives you ability of setting an image as the document's background.
- An example of a background image's HTML code is as follows:

```
<BODY BACKGROUND="hi.gif" BGCOLOR="#FFFFFF"></BODY>
```

ABESIT
CRC-Training

Headings, Paragraphs, Breaks & Horizontal Rules

- In this you will add headings to your page, insert paragraphs, add some breaks, and add horizontal rules.
- **Objectives**
  - List and describe the different Heading elements.
  - Use Paragraphs to add text to a document.
  - Insert breaks where necessary.
  - Add a Horizontal Rule.

ABESIT
CRC-Training

Headings, <Hx> </Hx>

- Inside the **BODY** element, heading elements **H1** through **H6** are generally used for major divisions of the document. Headings are permitted to appear in any order, but you will obtain the best results when your documents are displayed in a browser if you follow these guidelines:
  - **H1**: should be used as the highest level of heading, **H2** as the next highest, and so forth.
  - You should not skip heading levels: e.g., an **H3** should not appear after an **H1**, unless there is an **H2** between them.

ABESIT
CRC-Training

Headings, <Hx> </Hx>

```
<HTML>
<HEAD>
<TITLE> Example Page</TITLE>
</HEAD>
<BODY>
<H1> Heading 1 </H1>
<H2> Heading 2 </H2>
<H3> Heading 3 </H3>
<H4> Heading 4 </H4>
<H5> Heading 5 </H5>
<H6> Heading 6 </H6>
</BODY>
</HTML>
```

Heading 1
Heading 2
Heading 3
Heading 4
Heading 5
Heading 6

ABESIT
CRC-Training

Paragraphs, <P> </P>

- Paragraphs allow you to add text to a document in such a way that it will automatically adjust the end of line to suite the window size of the browser in which it is being displayed. Each line of text will stretch the entire length of the window.

```
<HTML><HEAD>
<TITLE> Example Page</TITLE>
</HEAD>
<BODY></H1> Heading 1 </H1>
<P> Paragraph 1,....</P>
<H2> Heading 2 </H2>
<P> Paragraph 2, ....</P>
<H3> Heading 3 </H3>
<P> Paragraph 3, ....</P>
<H4> Heading 4 </H4>
<P> Paragraph 4, ....</P>
<H5> Heading 5 </H5>
<P> Paragraph 5, ....</P>
<H6> Heading 6</H6>
<P> Paragraph 6, ....</P>
</BODY></HTML>
```

Heading 1
Paragraph 1,....
Heading 2
Paragraph 2,....
Heading 3
Paragraph 3,....
Heading 4
Paragraph 4,....
Heading 5
Paragraph 5,....
Heading 6
Paragraph 6,....

Break,

- Line breaks allow you to decide where the text will break on a line or continue to the end of the window.
- A
 is an empty Element, meaning that it may contain attributes but it does not contain content.
- The
 element does not have a closing tag.

Break,

```
<HTML>
<HEAD>
<TITLE> Example Page</TITLE>
</HEAD>
<BODY>
<H1> Heading 1 </H1>
<P>Paragraph 1, <BR>
Line 2 <BR> Line 3 <BR>....
</P>
</BODY>
</HTML>
```

Heading 1

Paragraph 1,....
Line 2
Line 3
....

Horizontal Rule, <HR>

- The <HR> element causes the browser to display a horizontal line (rule) in your document.
- <HR> does not use a closing tag, </HR>.

Attribute	Description	Default Value
SIZE	Height of the rule in pixels	2 pixels
WIDTH	Width of the rule in pixels or percentage of screen width	100%
NOSHADE	Draw the rule with a flat look instead of a 3D look	Not set (3D look)
ALIGN	Aligns the line (Left, Center, Right)	Center
COLOR	Sets a color for the rule (IE 3.0 or later)	Not set

Horizontal Rule, <HR>

```
<HTML>
<HEAD>
<TITLE> Example Page</TITLE>
</HEAD>
<BODY>
<H1> Heading 1 </H1>
<P>Paragraph 1, <BR>
Line 2 <BR>
<HR>Line 3 <BR>
</P>
</BODY>
</HTML>
```

Heading 1

Paragraph 1,....
Line 2

Line 3

ABESIT
CRC-Training

Character Formatting

- In this you will learn how to enhance your page with Bold, Italics, and other character formatting options.

Objectives

- Change the color and size of your text.
- Use Common Character Formatting Elements.
- Align your text.
- Add special characters.
- Use other character formatting elements.

ABESIT
CRC-Training

Bold, Italic and other Character Formatting Elements

- **<FONT SIZE="+2">** Two sizes bigger**</FONT>**
- The size attribute can be set as an absolute value from 1 to 7 or as a relative value using the "+" or "-" sign. Normal text size is 3.
- **<B> Bold </B>**
- **<I>** ***Italic*** **</I>**
- **<U>** Underline **</U>**
- Color = "#RRGGBB" The COLOR attribute of the FONT element. E.g., **<FONT COLOR="#RRGGBB">this text has color</FONT>**
- **<PRE> Preformatted </PRE>** Text enclosed by PRE tags is displayed in a mono-spaced font. Spaces and line breaks are supported without additional elements or special characters.

ABESIT
CRC-Training

Bold, Italic and other Character Formatting Elements

- **<EM>** ***Emphasis*** **</EM>** Browsers usually display this as italics.
- **<STRONG> STRONG </STRONG>** Browsers display this as bold.
- **<TT> TELETYPE </TT>** Text is displayed in a mono-spaced font. A typewriter text, e.g. fixed-width font.
- **<CITE> Citation </CITE>** represents a document citation (**italics**). **For titles of books, films, etc. Typically displayed in italics. (*A Beginner's Guide to HTML*)**

ABESIT
CRC-Training

Bold, Italic and other Character Formatting Elements

```
<P> <FONT SIZE="+1"> One Size
    Larger </FONT><BR> - Normal
    –<BR>
<FONT SIZE="-1"> One Size
    Smaller </FONT> <BR>
<B> Bold</B> - <I> italics</I> - <U>
    Underlined </U> -
<FONT COLOR="#FF0000">
    Colored </FONT> <BR>
<EM> Emphasized</EM> -
    <STRONG> Strong
    </STRONG> - <TT> Tele Type
    </TT> <BR>
```

One Size Larger
- Normal –
One Size Smaller
Bold - *italics* - Underlined - Colored
Emphasized - **Strong** - Tele Type

Alignment

- Some elements have attributes for alignment (ALIGN) e.g. Headings, Paragraphs and Horizontal Rules.
- The Three alignment values are : LEFT, RIGHT, CENTER.
- <CENTER></CENTER> Will center elements.
- **<DIV ALIGN="value"></DIV>** Represents a division in the document and can contain most other element type. The alignment attribute of the DIV element is well supported.
- **<TABLE></TABLE>** Inside a TABLE, alignment can be set for each individual cell.

Special Characters & Symbols

- These Characters are recognized in HTML as they begin with an ampersand and end with with a semi-colon e.g. **&value;** The value will either be an entity name or a standard ASCII character number. They are called **escape sequences**.
- The next table represents some of the more commonly used special characters. For a comprehensive listing, visit the W3C's section on special characters at: http://www.w3.org/MarkUp/HTMLPlus/htmlplus_13.html

Special Characters & Symbols

Special Character	Entity Name	Special Character	Entity Name
Ampersand	& &	Greater-than sign	> >
Asterisk	∗ ∗	Less-than sign	< <
Cent sign	¢ ¢	Non-breaking space	
Copyright	© ©	Quotation mark	" "
Fraction one qtr	¼ ¼	Registration mark	® ®
Fraction one half	½ ½	Trademark sign	™ ™

Special Characters & Symbols

- Additional escape sequences support accented characters, such as:
  - **ö**
    - a lowercase o with an umlaut: ö
  - **ñ**
    - a lowercase n with a tilde: ñ
  - **È**
    - an uppercase E with a grave accent: È

NOTE: Unlike the rest of HTML, the escape sequences are **case sensitive**. You cannot, for instance, use < instead of <.

Additional Character Formatting Elements

- **<STRIKE>** strike-through text**</STRIKE>**
  DEL is used for STRIKE at the latest browsers
- **<BIG>** places text in a big font**</BIG>**
- **<SMALL>** places text in a small font**</SMALL>**
- **_{** places text in subscript position **}**
- **^{** places text in superscript style position **}**

Lists

- Lists provide methods to show item or element sequences in document content. There are three main types of lists:

  1. Unordered lists:- unordered lists are bulleted.

  2. Ordered lists:- Ordered lists are numbered.

  3. Definition lists:- Used to create a definition list

List Elements

- HTML supplies several list elements. Most list elements are composed of one or more <LI> (List Item) elements.
- **UL : Unordered List.** List items are not listed in a particular order. Items in this list start with a list mark such as a bullet. Browsers will usually change the list mark in nested lists.

```
<UL>
<LI> List item 1 ...</LI>
<LI> List item 2 ...</LI>
</UL>
```

- List item 1 ...
- List item 2 ...

List Elements

- ***e.g.***

```
<ul>
    <li> Name:  Your name </li>
    <li> Section: ###  </li>
    <li>  Instructor: Yuping </li>
</ul>
```

Result

- Name: Your name
- Section: ###
- Instructor: Yuping

List Elements

- You have the choice of three bullet types: **disc(default), circle, square.**
- These are controlled in Netscape Navigator by the "TYPE" attribute for the <UL> element.

```
<UL TYPE="square">
    <LI> List item 1 …</LI>
    <LI> List item 2 …</LI>
    <LI> List item 3 …</LI>
</UL>
```

- List item 1 …
- List item 2 …
- List item 3 …

List Elements

- **OL: Ordered List.** used to display information in a numeric order. Items in this list are numbered automatically by the browser.

```
<OL>
    <LI> List item 1 …</LI>
    <LI> List item 2 …</LI>
    <LI> List item 3 …</LI>
</OL>
```

1. **List item 1 …**
2. **List item 2 …**
3. **List item 3 …**

List Elements

- ***e.g.***

```
<ol>
    <li> Name: Your name </li>
    <li> Section: ### </li>
    <li> Instructor: Yuping </li>
</ol>
```

Result

1. Name: Your name
2. Section: ###
3. Instructor: Yuping

List Elements

- You have the choice of setting the TYPE Attribute to one of five numbering styles.

TYPE	Numbering Styles	
1	Arabic numbers	1,2,3, ……
a	Lower alpha	a, b, c, ……
A	Upper alpha	A, B, C, ……
i	Lower roman	i, ii, iii, ……
I	Upper roman	I, II, III, ……

List Elements

- You can specify a starting number for an ordered list.

```
<OL TYPE =“i”>
    <LI> List item 1 …</LI>
    <LI> List item 2 …</LI>
</OL>
<P> text ….</P>
<OL TYPE=“i” START=“3”>
    <LI> List item iii …</LI>
    <LI VALUE=“6”> List item vi …</LI>
</OL>
```

i. List item 1 …
ii. List item 2 …

text ….

iii. List item iii …
vi. List item vi …

List Elements

- **DL: Definition List**. This kind of list is different from the others. Each item in a DL consists of one or more **Definition Terms (DT elements),** followed by one or more **Definition Description (DD elements).**

```
<DL>
<DT> HTML </DT>
<DD> Hyper Text Markup Language </DD>
<DT> DOG </DT>
<DD> A human’s best friend!</DD>
</DL>
```

HTML
Hyper Text Markup Language
DOG
A human’s best friend!

Nesting Lists

- You can nest lists by inserting a UL, OL, etc., inside a list item (LI).

Example

```
<UL TYPE = “square”>
    <LI> List item …</LI>
    <LI> List item …
        <OL TYPE=“i” START=“3”>
            <LI> List item …</LI>
            <LI> List item …</LI>
            <LI> List item …</LI>
            <LI> List item …</LI>
            <LI> List item …</LI>
        </OL>
    </LI>
    <LI> List item …</LI>
</UL>
```

- List item …
- List item …
  iii. List item …
  iv. List item …
  v. List item …
  vi List item …
  vii. List item …
- List item …

Hyperlink

- A hyperlink is a reference (an address) to a resource on the web.
- Hyperlinks can point to any resource on the web: an HTML page, an image, a sound file, a movie, etc.
- The HTML anchor element <a>, is used to define both hyperlinks and anchors.

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

- The href attribute defines the link address.

```
<a href="https://www.abesit.in">Visit ABESIT!</a>
```

Hyperlink

- Link to another location or file
- Syntax:

Link to ABESIT

Content displayed on the page

Starting Tag

Attribute of the tag: the address of the hyperlink

Ending tag

- The tags used to produce links are the <A>and </A>.
- The <A> tells where the link should start and the </A> indicates where the link ends. Everything between these two will work as a link.

Result of Hyperlink tag

Link to ABESIT

Hyperlink

- <body LINK="#C0C0C0" VLINK="#808080" ALINK="#FF0000">
- LINK - standard link - to a page the visitor hasn't been to yet. (standard color is blue - #0000FF).
- VLINK - visited link - to a page the visitor has been to before. (standard color is purple - #800080).
- ALINK - active link - the color of the link when the mouse is on it. (standard color is red - #FF0000). If one need to change the color

Click here   to go to ABESIT.

Internal Links

- Internal Links : Links can also be created inside large documents to simplify navigation. Today's world wants to be able to get the information quickly. Internal links can help you meet these goals.
- Select some text at a place in the document that you would like to create a link to, then add an anchor to link to like this:

<A NAME="bookmark_name"></A>

- The Name attribute of an anchor element specifies a location in the document that we link to shortly. All NAME attributes in a document must be unique.
- Next select the text that you would like to create as a link to the location created above.

<A HREF="#bookmark_name">Go To Book Mark</A>

E-Mail (Electronic Mail)

- E.g. mailto:kmf@gmail.com
- The type of service is identified as the mail client program. This type of link will launch the users mail client.
- The recipient of the message is kmf@gmail.com

```
<A HREF="mailto:kmf@gmail.com">Send me More  Information </A>
```


Images

- To display an image on a page, you need to use the src attribute.
- src stand for "source". The value of the src attribute is the URL of the image you want to display on your page.
- It is a empty tag.

```
<img src="url">
```

- Ex

```
<img src="picture.jpg">
```


Images

- **<IMG>**This element defines a graphic image on the page.
- **Image File (SRC: source):** This value will be a URL (location of the image).
- **Alternate Text (ALT):** This is a text field that describes an image or acts as a label. It is displayed when they position the cursor over a graphic image.
- **Alignment (ALIGN):** This allows you to align the image on your page.
- **Width (WIDTH):** is the width of the image in pixels.
- **Height (HEIGHT):** is the height of the image in pixels.
- **Border (BORDER):** is for a border around the image, specified in pixels.
- **HSPACE:** is for Horizontal Space on both sides of the image specified in pixels. A setting of 5 will put 5 pixels of invisible space on both sides of the image.
- **VSPACE:** is for Vertical Space on top and bottom of the image specified in pixels. A setting of 5 will put 5 pixels of invisible space above and bellow the image.

Images

- **Examples of** image.

1) <IMG SRC="jordan.gif" border="4">

2) <IMG SRC="jordan.gif" width="60" height="60">

3) <IMG SRC="jordan.gif" ALT="This is a text that goes with the image">

4) <IMG SRC="jordan.gif" Hspace="30" Vspace="10" border="20">

5) <IMG SRC ="jordan.gif" align="left">

ABESIT
CRC-Training

Images

```
<html>
   <body>
       <p><img src="p1.jpg" width="20" height="20"></p>
       <p><img src="p1.jpg" width="40" height="40"></p>
       <p><img src="p1.jpg" width="60" height="60"></p>
       <p><img src="p1.jpg" width="80" height="80"></p>
       <p><img src="p1.jpg" width="100" height="100">
            </p>
   </body>
</html>
```

ABESIT
CRC-Training

Images

Align=*position*	Image/Text Placement
Left	Image on left edge; text flows to right of image
Right	Image on right edge; text flows to left
Top	Image is left; words align with top of image
Bottom	Image is left; words align with bottom of image
Middle	Words align with middle of image

ABESIT
CRC-Training

Images

```
<html>
   <body>
       <img src="p1.jpg" align="left" width="48" height="48">
       <img src="p1.jpg" align="right" width="48" height="48">
   </body>
</html>
```

ABESIT
CRC-Training

Images

```
<html>
   <body>
       <p>An image  <img src="g1.jpg" align="bottom" width="48"
            height="48"> in the text</p>
       <p>An image  <img src="g1.jpg" align="middle" width="48"
            height="48"> in the text</p>
       <p>An image  <img src="g1.jpg" align="top" width="48"
            height="48"> in the text</p>
       <p>Note that bottom alignment is the default alignment</p>
       <p><img src="g1.jpg" width="48" height="48">  An image
            before the text</p>
       <p>An image after the text <img src="g1.jpg" width="48"
            height="48"> </p>
   </body>
</html>
```

Images - gif

Tables

- The <TABLE></TABLE> element has four sub-elements:
  1. Table Row<TR></TR>.
  2. Table Header <TH></TH>.
  3. Table Data <TD></TD>.
  4. Caption <CAPTION></CAPTION>.
- The table row elements usually contain table header elements or table data elements.

Tables

```
<table border="1">
    <tr>
        <th> Column 1 header </th>
        <th> Column 2 header </th>
    </tr>
    <tr>
        <td> Row1, Col1 </td>
        <td> Row1, Col2 </td>
    </tr>
    <tr>
        <td> Row2, Col1 </td>
        <td> Row2, Col2 </td>
    </tr>
</table>
```


Tables

Column 1 Header	Column 2 Header
Row1, Col1	Row1, Col2
Row2, Col1	Row2, Col2

Tables Attributes

- **BGColor:** Some browsers support background colors in a table.
- **Width:** you can specify the table width as an absolute number of pixels or a percentage of the document width. You can set the width for the table cells as well.
- **Border:** You can choose a numerical value for the border width, which specifies the border in pixels.
- **CellSpacing:** Cell Spacing represents the space between cells and is specified in pixels.
- **CellPadding:** Cell Padding is the space between the cell border and the cell contents and is specified in pixels.
- **Align:** tables can have left, right, or center alignment.
- **Background:** Background Image, will be titled in IE3.0 and above.
- BorderColor, BorderColorDark.

Tables Caption

- A table caption allows you to specify a line of text that will appear centered above or bellow the table.

```
<TABLE BORDER=1 CELLPADDING=2>
    <CAPTION ALIGN="BOTTOM"> Label For My Table </CAPTION>
```

- The Caption element has one attribute ALIGN that can be either TOP (Above the table) or BOTTOM (below the table).

Tables Header

- Table Data cells are represented by the TD element. Cells can also be TH (Table Header) elements which results in the contents of the table header cells appearing centered and in bold text.

Table Data and Table Header Attributes

- **Colspan:** Specifies how many cell columns of the table this cell should span.
- **Rowspan:** Specifies how many cell rows of the table this cell should span.
- **Align:** cell data can have left, right, or center alignment.
- **Valign:** cell data can have top, middle, or bottom alignment.
- **Width:** you can specify the width as an absolute number of pixels or a percentage of the document width.
- **Height:** You can specify the height as an absolute number of pixels or a percentage of the document height.

Basic Table Code

```
<TABLE BORDER=1 width=50%>
    <CAPTION>  <h1>Spare Parts <h1> </Caption>
      <TR><TH>Stock Number</TH><TH>Description</TH><TH>List Price</TH></TR>
      <TR><TD bgcolor=red>3476-AB</TD><TD>76mm Socket</TD><TD>45.00</TD></TR>
      <TR><TD>3478-AB</TD><TD><font color=blue>78mm Socket</font>
                </TD><TD>47.50</TD></TR>
      <TR><TD>3480-AB</TD><TD>80mm Socket</TD><TD>50.00</TD></TR>
</TABLE>
```

Spare Parts

Stock Number	Description	List Price
[redacted]	76mm Socket	45.00
3478-AB	78mm Socket	47.50
3480-AB	80mm Socket	50.00

Table Data and Table Header Attributes

```
<Table border=1 cellpadding =2>
  <tr> <th> Column 1 Header</th> <th> Column 2 Header</th> </tr>
  <tr> <td colspan=2> Row 1 Col 1</td> </tr>
  <tr> <td rowspan=2> Row 2 Col 1</td> <td> Row 2 Col2</td> </tr>
  <tr> <td> Row 3 Col2</td> </tr>
</table>
```

Column 1 Header	Column 2 Header
Row 1 Col 1	
Row 2 Col 1	Row 2 Col 2
	Row 3 Col 2

Special Things to Note

- **TH, TD and TR should always have end tags.**
  Although the end tags are formally optional, many browsers will mess up the formatting of the table if you omit the end tags. In particular, you should ***always*** use end tags if you have a TABLE within a TABLE -- in this situation, the table parser gets hopelessly confused if you don't close your TH, TD and TR elements.
- **A default TABLE has no borders**
  By default, tables are drawn without border lines. You need the BORDER attribute to draw the lines.
- **By default, a table is flush with the left margin**
  TABLEs are plopped over on the left margin. If you want centered tables, You can either: place the table inside a DIV element with attribute ALIGN="center". Most current browsers also supports table alignment, using the ALIGN attribute. Allowed values are "left", "right", or "center", for example: <TABLE ALIGN="left">. The values "left" and "right" float the table to the left or right of the page, with text flow allowed around the table. This is entirely equivalent to IMG alignment

What will be the output?

```
<TABLE BORDER width="750">
    <TR> <TD colspan="4" align="center">Page Banner</TD></TR>
    <TR> <TD rowspan="2" width="25%">Nav Links</TD><TD
colspan="2">Feature Article</TD> <TD rowspan="2" width="25%">Linked
Ads</TD></TR>
    <TR><TD width="25%">News Column 1 </TD> <TD
width="25%"><News Column 2 </TD></TR>
</TABLE>
```

FORMS

- To insert a form we use the <FORM></FORM> tags. The rest of the form elements must be inserted in between the form tags.

```
<HTML>
  <HEAD>   <TITLE> Sample Form</TITLE>  </HEAD>
  <BODY BGCOLOR="FFFFFF">
    <FORM ACTION = "formtest.php">
       <P> First Name: <INPUT TYPE="TEXT" NAME="fname" MAXLENGTH="50"> </P>
       <P> <INPUT TYPE="SUBMIT" NAME="fsubmit1" VALUE="Send Info"> </P>
    </FORM>
  </BODY>
</HTML>
```


<FORM> element attributes

- **ACTION:** is the **URL** of the **CGI** (Common Gateway Interface) program that is going to accept the data from the form, process it, and send a response back to the browser.
- **METHOD: GET** (default) or **POST** specifies which **HTTP** method will be used to send the form's contents to the web server. The CGI application should be written to accept the data from either method.
- **NAME:** is a form name used by VBScript or JavaScripts.
- **TARGET:** is the target frame where the response page will show up.
- **Form elements have properties: Text boxes, Password boxes, Checkboxes, Option(Radio) buttons, Submit, Reset, File, Hidden and Image.**
- **The properties are specified in the TYPE Attribute of the HTML element <INPUT></INPUT>.**

Form Elements

<INPUT> Element's Properties
TYPE= Type of INPUT entry field.
NAME = Variable name passed to CGI application
VALUE= The data associated with the variable name to be passed to the CGI application
CHECKED= Button/box checked
SIZE= Number of visible characters in text field
MAXLENGHT= Maximum number of characters accepted.

Text Box

- **Text boxes:** Used to provide input fields for text, phone numbers, dates, etc.

  <INPUT TYPE= " TEXT ">

  Browser will display

Textboxes use the following attributes:

- **TYPE:** text.
- **SIZE:** determines the size of the textbox in characters. **Default=20** characters.
- **MAXLENGHT:** determines the maximum number of characters that the field will accept.
- **NAME:** is the name of the variable to be sent to the CGI application.
- **VALUE:** will display its contents as the default value.

Text Box : Example

```
<HTML> <HEAD> <TITLE>Form_Text_Type</TITLE> </HEAD>
<BODY>
 <h1> <font color=blue>Please enter the following bioData</font></h1>
  <FORM name="fome1"  Method=" get " Action=" URL " >
  First Name: <INPUT TYPE="TEXT" NAME="FName" SIZE="15" MAXLENGTH="25"><BR>
  Last Name: <INPUT TYPE="TEXT" NAME="LName" SIZE="15" MAXLENGTH="25"><BR>
  Nationality: <INPUT TYPE="TEXT" NAME="Country" SIZE="25" MAXLENGTH="25"><BR>
 The Phone Number: <INPUT TYPE="TEXT" NAME="Phone"
  SIZE="15" MAXLENGTH="12"> <BR>
  </FORM>
</BODY> </HTML>
```

Password

- **Password:** Used to allow entry of passwords.

  <INPUT TYPE= " PASSWORD " >

  Browser will display

  Text typed in a password box is starred out in the browser display.

  Password boxes use the following attributes:

- **TYPE:** password.
- **SIZE:** determines the size of the textbox in characters.
- **MAXLENGHT:** determines the maximum size of the password in characters.
- **NAME:** is the name of the variable to be sent to the CGI application.
- **VALUE:** is usually blank.

Password : Example

```
<HTML>
  <HEAD> <TITLE>Form_Password_Type</TITLE> </HEAD>
 <BODY>
  <h1> <font color=red>To Access, Please enter:</font></h1>
   <FORM name="fome2"  Action="url"  method="get">
    User Name:  <INPUT TYPE="TEXT" Name="FName" SIZE="15" MAXLENGTH="25"><BR>
    Password: <INPUT TYPE="PASSWORD" NAME="PWord"   value="" SIZE="15"
     MAXLENGTH="25"><BR>
   </FORM>
</BODY> </HTML>
```

Hidden

- **Hidden:** Used to send data to the CGI application that you don't want the web surfer to see, change or have to enter but is necessary for the application to process the form correctly.

  <INPUT TYPE="HIDDEN">

  Nothing is displayed in the browser.

  Hidden inputs have the following attributes:

- **TYPE:** hidden.
- **NAME:** is the name of the variable to be sent to the CGI application.
- **VALUE:** is usually set a value expected by the CGI application.

ABESIT
CRC-Training

Check Box

- **Check Box:** Check boxes allow the users to select more than one option.

  <INPUT TYPE="CHECKBOX">

  Browser will display 

  Checkboxes have the following attributes:

- **TYPE:** checkbox.
- **CHECKED:** is blank or CHECKED as the initial status.
- **NAME:** is the name of the variable to be sent to the CGI application.
- **VALUE:** is usually set to a value.

ABESIT
CRC-Training

Check Box : Example

```
<HTML> <HEAD><TITLE>CheckBoxType</TITLE> </HEAD>
<BODY>
  <h1> <font color=green>Please check one of the following</font></h1>
  <FORM name="fome3"  Action="url"  method="get">
   <font color=red> Select Country: </font><BR>
    jordan:<INPUT TYPE="CheckBox" Name="country"  CHECKED><BR>
    Yemen<INPUT TYPE="CheckBox"  Name="country"><BR>
    Qatar:<INPUT TYPE="CheckBox" Name="country"><BR> <BR>
   <font color=blue>Select Language:</font><BR>
      Arabic:<INPUT TYPE="CheckBox" Name="language"  CHECKED><BR>
      English:<INPUT TYPE="CheckBox" Name="language"><BR>
      French:<INPUT TYPE="CheckBox" Name="language"> <BR>
</FORM> </BODY></HTML>
```

ABESIT
CRC-Training

Radio Button

- **Radio Button:** Radio buttons allow the users to select only one option.

  <INPUT TYPE="RADIO">

  Browser will display 

  Radio buttons have the following attributes:

- **TYPE:** radio.
- **CHECKED:** is blank or CHECKED as the initial status. Only one radio button can be checked.
- **NAME:** is the name of the variable to be sent to the CGI application.
- **VALUE:** usually has a set value.

ABESIT
CRC-Training

Radio Button : Example

```
<HTML> <HEAD><TITLE>CheckBoxType</TITLE> </HEAD>
  <BODY>
   <h1> <font color=green>Please check one of the following</font></h1>
    <FORM name="fome3"  Action="url"  method="get">
     <font color=red> Select Country: </font><BR>
    jordan:<INPUT TYPE= "RADIO"  Name="country"  CHECKED><BR>
    Yemen<INPUT TYPE="RADIO "  Name="country"><BR>
    Qatar:<INPUT TYPE="RADIO"  Name="country"><BR> <BR>
    <font color=blue>Select Language:</font><BR>
    Arabic:<INPUT TYPE="RADIO"  Name="language"  CHECKED><BR>
    English:<INPUT TYPE=" RADIO " Name="language"><BR>
    French:<INPUT TYPE=" RADIO "  Name="language"> <BR>
  </FORM> </BODY></HTML>
```

Radio Button : Example

```
<HTML>
 <HEAD> <TITLE>RADIOBox</TITLE> </HEAD>
 <BODY>
  Form #1:  <FORM>
     <INPUT TYPE="radio" NAME="choice" VALUE="one"> Yes.
     <INPUT TYPE="radio" NAME="choice" VALUE="two"> No.
   </FORM>
  <HR color=red size="10" >
Form #2:  <FORM>
    <INPUT TYPE="radio" NAME="choice" VALUE="three" CHECKED> Yes.
  <INPUT TYPE="radio" NAME="choice" VALUE="four"> No.
</FORM>
</BODY></HTML>
```


Push Button

- **Push Button:** This element would be used with JavaScript to cause an action to take place.

  `<INPUT TYPE="BUTTON">`

  Browser will display 

  Push Button has the following attributes:

- **TYPE:** button.
- **NAME:** is the name of the button to be used in scripting.
- **VALUE:** determines the text label on the button.

Push Button : Example

```
<FORM>
 <FONT Color=red>
  <h1>Press Here to see a baby crying:<BR>
  <INPUT TYPE="button" VALUE="PressMe"><BR><BR>
  <FONT Color=blue>
   Click Here to see a baby shouting:<BR>
  <INPUT TYPE="button" VALUE="ClickMe" > <BR><BR>
  <FONT Color=green>
  Hit Here to see a baby eating:<BR>
  <INPUT TYPE="button" VALUE="HitME" > <BR><BR>
 <FONT Color=yellow>
</FORM>
```

Submit Button

- **Submit:** Every set of Form tags requires a Submit button. This is the element causes the browser to send the names and values of the other elements to the CGI Application specified by the ACTION attribute of the FORM element.

  `<INPUT TYPE="SUBMIT">`

  The browser will display 

  Submit has the following attributes:

- **TYPE:** submit.
- **NAME:** value used by the CGI script for processing.
- **VALUE:** determines the text label on the button, usually Submit Query.

Submit Button : Example

```
<FORM Action="URL" method="get">
  First Name: <INPUT TYPE="TEXT" Size=25 name="firstName"><BR>
  Family Name: <INPUT TYPE="TEXT" Size=25 name="LastName"><BR>
  <BR>
  <FONT Color=red>
  Press Here to submit the data:<BR>
  <INPUT TYPE="submit" VALUE="SubmitData ">
</FORM>
```

Reset Button

- **Reset:** It is a good idea to include one of these for each form where users are entering data. It allows the surfer to clear all the input in the form.

  `<INPUT TYPE="RESET">`

  Browser will display 

  Reset buttons have the following attributes:

- **TYPE:** reset.
- **VALUE:** determines the text label on the button, usually Reset.

Reset Button : Example

```
<FORM  Action="URL"  method="get">
  First Name: <INPUT TYPE="TEXT" Size=25 name="firstName"><BR>
  Family Name: <INPUT TYPE="TEXT" Size=25 name="LastName"><BR>
  <BR>
  <FONT Color = red>
  <STRONG><font size=5>Press Here to submit the data:</font></STRONG><BR>
  <INPUT TYPE="submit" VALUE="SubmitData">
  <INPUT TYPE="RESET" VALUE="Reset">
</FORM>
```

Image Submit Button

- **Image Submit Button:** Allows you to substitute an image for the standard submit button.

  `<INPUT TYPE="IMAGE" SRC="rajsthan.gif">`

  Image submit button has the following attributes:

- **TYPE:** Image.
- **NAME:** is the name of the button to be used in scripting.
- **SRC:** URL of the Image file.

Image Submit Button : Example

```
<form>
   <H1><font color=blue>
   Click to go rajasthan's Map:
    <INPUT  TYPE="IMAGE"  SRC="rajasthan.gif">
</form>
```

click to go rajasthan's map

File

- **File Upload:** You can use a file upload to allow surfers to upload files to your web server.

  <INPUT TYPE="FILE">

  Browser will display 

  File Upload has the following attributes:

- **TYPE:** file.
- **SIZE:** is the size of the text box in characters.
- **NAME:** is the name of the variable to be sent to the CGI application.
- **MAXLENGHT:** is the maximum size of the input in the textbox in characters.

File : Example

```
<BODY bgcolor="lightblue">
   <form>
      <H3><font color=forestgreen>
         Please attach your file here to for uploading to My
          <font color =red>SERVER...<BR>
        <INPUT  TYPE="File"  name="myFile"  size="30">
        <INPUT  TYPE="Submit"  value="SubmitFile">
   </form>
</BODY>
```


TEXTAREA

- **<TEXTAREA></TEXTAREA>:** is an element that allows for free form text entry.

  Browser will display 

  Textarea has the following attributes:

- **NAME:** is the name of the variable to be sent to the CGI application.
- **ROWS:** the number of rows to the textbox.
- **COLS:** the number of columns to the textbox.

TEXTAREA : Example

```
<BODY bgcolor=lightblue>
 <form>
  <TEXTAREA  COLS=40  ROWS=20  Name="comments" >
  From observing the apathy of those
  about me during flag  raising I
  concluded that patriotism if not
  actually  on the decline is at least
  in a state of dormancy.
     Written by Khaled Al-Fagih
  </TEXTAREA>:
</form>
</BODY>
```

SELECT

- **<SELECT></SELECT>** elements, where the attributes are set differently.

  The Select elements attributes are:

- **NAME:** is the name of the variable to be sent to the CGI application.
- **SIZE:** this sets the number of **visible** choices.
- **MULTIPLE:** the presence of this attribute signifies that the user can make multiple selections. By default only one selection is allowed.

SELECT : Example

```
<BODY bgcolor=lightblue>
  <form>
  Select the cities you have visited:
    <SELECT name="list" size=5>
      <option> London</option>
      <option> Tokyo</option>
      <option> Paris</option>
      <option> New York</option>
      <option> LA</option>
      <option> KL</option>
    </SELECT>
</form> </BODY>
```

Drop Down List

- **Drop Down List:**

- **Name:** is the name of the variable to be sent to the CGI application.
- **Size:** 1.

ABESIT
CRC-Training

List Box

- **List Box:**

- **Name:** is the name of the variable to be sent to the CGI application.
- **SIZE:** is greater than one.

ABESIT
CRC-Training

Option

- **Option :** The list items are added to the **<SELECT>** element by inserting **<OPTION></OPTION>** elements.

  The Option Element's attributes are:

- **SELECTED:** When this attribute is present, the option is selected when the document is initially loaded. **It is an error for more than one option to be selected.**
- **VALUE:** Specifies the value the variable named in the select element.

ABESIT
CRC-Training

Example

```
<HTML>
 <BODY>
  <h2><font color=blue>What type of Computer do you have?</font><h2>
  <FORM>
 <SELECT NAME="ComputerType" size=4>
 <OPTION  value="IBM" SELECTED> IBM</OPTION>
 <OPTION  value="INTEL"> INTEL</OPTION>
 <OPTION value=" Apple"> Apple</OPTION>
 <OPTION value="Compaq"> Compaq</OPTION>
</SELECT>
</FORM></BODY></HTML>
```

ABESIT
CRC-Training

SELECT : Example

```
<HEAD> <TITLE>SELECT with Mutiple </TITLE> </HEAD>
<BODY>
<h2><font color=blue>What type of Computer do you have?</font><h2>
<FORM>
<SELECT NAME="ComputerType" size=5  multiple>
 <OPTION  value="IBM" > IBM</OPTION>
 <OPTION  value="INTEL"> INTEL</OPTION>
 <OPTION value=" Apple"> Apple</OPTION>
 <OPTION value="Compaq" SELECTED> Compaq</OPTION>
 <OPTION value=" other"> Other</OPTION>
</SELECT>
</FORM></BODY></HTML>
```

ABESIT
CRC-Training