

LESSON 1

Introduction

WHAT IS THE INTERNET?

Internet or internetworking refers to a wide network through which computers are interconnected globally with one another, and capable of sharing resources among themselves. This network is called 'internet' (with a lower case i). The 'Internet' refers to millions of computers, connected in a gigantic network, which communicate via TCP/IP protocols. This collection of computers store immense quantity of information. Computer networks are very common in businesses and government establishments. But the Internet is different from all others in two aspects. First, it is the biggest network in the world, and second it is freely accessible to the public.

HISTORY OF THE INTERNET

In 1957, when Soviet satellite Sputnik was launched, the United States believed it to be a great threat to its security. In response to this perceived threat, the United States started a major military and academic initiative to promote research in the areas of mathematics, pure sciences, and engineering. This initiative began under the auspices of the Department of Defense, with the formation of an organizational body called Defense Advanced Research Projects Agency (DARPA), later shortened to ARPA.

Marshall McLuhan proposed the idea of the global village, where electronic communications would come to function as the nerve system of the planet, replacing the town-square as the gathering place for information. Leonard Kleinrock (1961] of MIT proposed an alternative, called packet-switching. Instead of the traditional streaming of data in wave forms, packet-switching sends data in packets. The data packets in packet-switching systems are small and in addition to the data, they also carry information on where the data belongs in the original data stream and error checking codes to ensure that the data arrives uncorrupted. This

means that if a packet is lost or corrupted, the receiving computer can request that only the missing packet be resent, not the entire data stream. Although originally developed for radio transmission, packet-switching proved to be robust enough to merit its adoption in all forms of electronic data transfer. It would also become vital in the development of such technologies as Ethernet, which is now a standard for local area networks (LANs).

The year 1967, Wesley Clark coined the term Interface Message Processor or IMP which is a separate mini-computer that functions as an interface between a computer and the wide area network (WAN). In other words, IMPs allow different types of computers to talk to each other. All of this research would reach its first culmination in 1969 with the commissioning of ARPANET, the first wide area packet-switched network. ARPANET ran under a control program called the Network Control Program (NCP). NCP would be supplanted in the 70s by TCP/IP. ARPANET initially consisted of four computers at the

- University of California at Los Angeles (UCLA)
- Stanford Research Institute
- University of California at Santa Barbara (UCSB) and
- University of Utah

In two years, this network had expanded to fifteen nodes. In twenty years' time, the number of nodes on the burgeoning Internet topped 100,000. At the beginning of 2001, the number of nodes on the Internet was more than 100 million.

In 1972, Bob Kahn proposed the idea of an open architecture for the Internet. The idea was to allow any computer to plug into the Internet and be able to function with no modification to the computer itself. The original form of ARPANET had no error checking for packets lost between networks and could not address a packet beyond the IMP for the receiving network. This was because it was designed to be the only network. This worked fine as long as the people using it

were the military and affiliated academic researchers. But as the Internet grew, the need for open architectures as well as new protocols became inevitable.

In 1973, Bob Kahn and Vinton Cerf began working on a detailed protocol that addressed Kahn's Internetworking standards. What they produced was called TCP/IP which stands for Transmission Control Protocol/Internet Protocol. This protocol divides the process of transmitting data into two separate steps.

1. TCP handles the transmission of data and correction of in-stream, transmission errors.
2. IP handles the bundling of data into packets, addressing them and also functions as the interface between the LAN and the Internet, and handles errors where the data needs to be resent.

Along with TCP/IP came the IP address. Each computer on the Internet had a unique IP address, which has four parts, each part being between 0 and 255. The number 255 was convenient breakpoint in binary math because it equals 11111111, or the highest value presentable in one byte. Each number provided more detail for the address in a hierarchical order. So the first number referred to the geographic region, while the last number referred to a specific computer on a specific network. This system yielded over four billion possible combinations, although the number of usable addresses was significantly smaller than that.

With the increase in the number of computers connected to the Internet in the 70s, human-friendly addresses called domain names were implemented and maps to IP addresses, which the router used to send information to the correct recipient. IP addresses were all stored in the same large table, slowing down communications as every packet of data had to be checked against it. Paul Mockapetris solved this problem by developing the Domain Name System (DNS). The DNS is a distributed mechanism for domain name resolution, where different portions of the net handled only parts of the addresses. Since the address is hierarchical, main routers only checked the first level of the address, while local gateways only

needed to check the lowest level of the address to determine the specific computer to route the data packet to.

By 1986, there were 5000 Internet hosts. By 1989, this number reached 1004000. One year later in 1990, it was 300,000. The Internet explosion had begun. Now; the Internet offers a lot of services which even help the common man to explore.

INTERNET SERVICES AND ACCESSIBILITY

As already discussed, today the Internet offers an extensive range of services many of which were not speculated when the Internet was first launched services. Some of the services offered are as follows:

- **Electronic mail:** A service that helps to send and receive messages and to attach—files
- **Electronic mailing lists:** Everyone subscribed to the list gets a message sent to the list
- **USENET newsgroups:** Electronic bulletin board service
- **Real-time communication:** Chat, messengers, videoconferencing, whiteboards, etc.
- **File Transfer Protocol (FTP):** A service that helps to send and receive files to and from a file server
- **Telnet, ssh:** A remote login to other computers on the Internet, possibly anonymously
- **Gopher:** This is an older menu-driven document system, which is now mostly replaced by the Web
- **World Wide Web (WWW):** Documents and files of various types which are connected using hypertext links to create a Web-like structure and are accessed through the Internet by addresses called Uniform Resource Locators (URLs).

The Internet users gain access to these services through Internet Service Providers (ISPs). Usually, home PC is connected to the Internet through the landline telephone connection using a device called modem (short form of modulator—

demodulator), which enables a computer to transmit data over telephone lines. There are different types of Internet connections:

Dial-Up Connections: The dial-up access to Internet is a very popular method because it is—relatively—expensive. To use this access, the computer should have a modem to connect to the phone system and a software that uses the modem. This software instructs the modem to place a telephone call to the number provided by the ISP. At the ISP, another modem answers the call by giving the connection.

Most of the Internet Service Providers supply 56 Kbps connection. The 56 Kbps refer to the speed with which the information from the Internet can travel down the phone line into your computer. The connection is normally established by dialing up to a particular ISP.

High Speed Connections: High Speed Connections allow information to travel quickly. With this connection, a lot of Internet services work better at higher speeds. There are four basic types of high speed connections:

Digital Subscriber Lines (DSL): With DSL, the computer is always connected to the Internet. This technology sends digital data through the existing phone lines to carry Internet services.

Cable: Although the same wires are used, there is no interference to/from the telephone calls. The Internet signal is carried on the TV cable network.

Satellite : This connection needs a phone line to send information. It receives data at high speed from the satellite.

Integrated Services Digital Network (ISDN): ISDN is an international communications standard for sending voice, video and data over digital telephone lines or over normal telephone lines.

USES OF THE INTERNET

The easy connectivity and convenient accessibility of Internet services have led to its use in a number of different areas. Some of the services provided by the Internet are:

- telecommuting (working from home or anywhere) and online conferencing
- business, advertising and online shopping (e.g. Dell.com for computers, Amazon.com for books)

- news, jobs, soft wares
- online courses, virtual classrooms, coachings
- government services, politics and national defense
- electronic publishing (magazines, newspapers and news services)
- entertainment (television, radio, videos, audio MP3s, etc.)
- teaching and learning (course websites, conferencing, simulation, visualization, etc.)
- scholarly research (searchable databases of journal articles; individual Web publishing, etc.)
- General information about a subject, especially which is not easily available from other sources
- Correspondence (email, chatting, etc.)

PROTOCOLS

Protocol is a set of rules or an agreement that specifies a common language that computers on a network use for communication with other computers. It also - specifies the conditions under which a particular message should be sent or responded and the particular method of doing it. They give the specification on how the computers talk with each other. There are various protocol suites available.)

Ethernet: This is used to transfer information on a LAN. It specifies a number of wiring and signalling standards for the physical layer, two ways of network access (Media Access Control/Data Link Layer) and a common address format.

Internet Protocol (IP): This protocol provides communicable global addresses of/to the computers. The computers identify each other by the IP addresses.

Transport Control Protocol (TCP): This protocol guarantees reliable, proper delivery of data from the sender to the receiver. It breaks large messages, transports them reliably and reassembles them.

File Transfer Protocol (FTP): This is used to connect two computers over the Internet so that the user of one computer can transfer files and perform file commands on the other computer. It exchanges files over any network that supports TCP/IP protocol.

Hypertext Transport Protocol (HTTP): This protocol is used to retrieve Web pages from a Web server.

Simple Mail Transfer Protocol (SMTP): This protocol is used for email transmissions.

LESSON-2

WEB CONCEPTS

The World Wide Web is an international hypertext system that links together millions of documents. A hypertext link is a word or Picture, which requests a different file from the Internet when you click on it. Hypertext markup language (HTML) is a collection of tags, which is used to create formatted hypertext documents. A Web page is a document created using HTML. A website is a collection of related pages. Webpages and sites can be stored on the hard drive of a local computer or a Web server on the Internet. A Web browser is a program that displays the Web pages it retrieves. Popular browsers include Microsoft Internet Explorer, Netscape Navigator, Mozilla, and Konqueror.

The Client/Server Model of the Web

Most Internet services rely on the client/server model. The Internet user is the client and has the client software installed on his computer to access various Internet services. When a user wants to connect to a particular information tool, he uses his client software to connect to server programs, Which provide the service or the information needed. The client/server model has become one of the central ideas of network computing, and is the basis of the TCP/IP protocol.

Figure 1.1 shows the client software on a student's computer contacting a server, using a particular protocol (the http protocol will produce a Web page response). The server may return the data retrieved from a database.

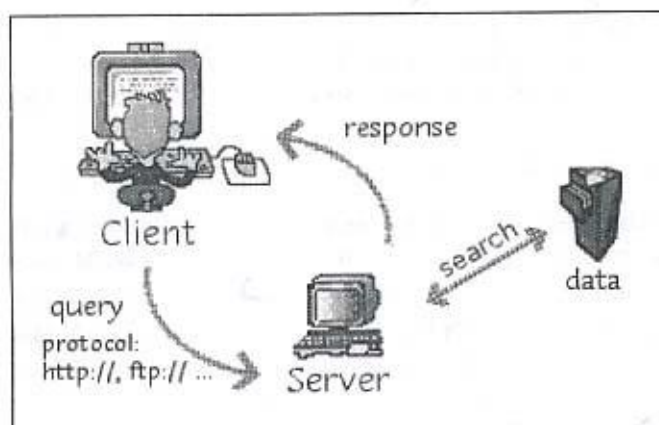


FIGURE 1.1 The client software on a student's computer connecting a server.

The Web browser is an example of the client software needed to access the World Wide Web servers. Most browsers function as client programs for World Wide Web and FTP access. For access to Telnet sites, a Telnet client is needed.

Retrieving Data from the Web

We can retrieve data by using the address of the file. This address is called a Uniform Resource Locator (URL). The Address box of the browser shows the URL of the currently displayed document. We can type a URL into the address box and can get the file we wanted.

Structure of URLs: In the same way as each host computer has a unique IP address, every data file or document on the Internet also has a unique address called a URL (Uniform Resource Locator). The URL consists of three parts: the protocol, the domain name and the path.

The protocol, as discussed earlier, is the set of rules which the computer follows in order to communicate with other computers. It lets the computer know how to process the information it receives. If the protocol is http: // for example, the computer knows it will be processing a World Wide Web document.

The domain name is the Internet address of the computer (server) that is hosting the site and storing the documents. This domain name may be, expressed as an IP address.

The **path** is the directory and file specification; it lets the computer know which directory and file to access after connecting to the server. The path is not a required element, but if you know the path, it will take you directly to the desired file or document. The path is also part of the URL which changes frequently. For example,

`http://www.linccweb.org/eresources.asp`

ht tp: // is the protocol. This lets you know that you are retrieving a World Wide Web document and lets the computer know how to process the hypertext file it is receiving.

`www.linocweb.org/` is the domain name, the address of the computer that is hosting the Web page.

resources. asp provides the path to the specific page you want; in this case, the name of the file (resources . asp) that provides links to electronic databases.

How the Web Works

The model of Web communication conforms to client server communication. The client is the Web browser like Netscape or Internet Explorer and the server is the Web server where the requested Web pages are residing. The user uses a Web browser and clicks on a document link. The browser reads the URL of the link and recognizes the different parts of the URL. The client (the user's computer) contacts the Web server (through the appropriate port number) and requests the file. The server retrieves the file from its storage device (e.g. a hard disk). If no filename is specified, default page (usually the home page) is retrieved. If the file is a program, the program is run on the server. If the file contains images, those are also retrieved, one by one. The server sends the results of the request to the Web browser, usually in the form of an HTML document. The Web browser reads the HTML codes, formats the document and images, and displays the results for the user to see.

Web Browsers: Navigation Features

The following points give the overview of the default features available in the browser.

The **back** button returns you to the page you last visited.

The **forward** button reverses the action of the back button.

The **home** button takes you back to a page that your browser has chosen as the home page.

- Your browser's home page opens when you open the browser. You can change the home page by opening the Preferences or Internet Options dialog box listed in the browser menu.
- Use Current makes the current page the default. Use Blank opens a browser window more quickly.

The **reload or refresh** button redisplay a page that has been changed or did not display properly. The **stop** button stops loading or downloading a page of a file that is taking too long. The **history** button lists all the pages you have recently visited, and makes it easy to return. **Bookmarks or favorites** store addresses so you can easily return to them later. You can organize them into folders.

Searching Information on the Web

If we know a website address you can type it in the address box and go there directly. Several tools quickly search multiple sites on the Internet:

- A subject guide or directory is a website that organizes its content hierarchically by subject (topics and subtopics).
- A search engine searches the Web for one or more keywords you type and displays a .list of pages found.
- A meta-search site submits the search to more than one search engine, thereby by reducing the distance between the client and the target.
- Using a human search service, we can hire a real person to perform our search.

REVIEW QUESTIONS

1. What is the Internet?
2. Distinguish the Internet and internet.
3. Explain the different types of Internet connections.
4. Write short notes on various services offered by the Internet.
5. The allows computer users to locate and view multimedia based documents over the Internet.
6. Define protocol.
7. Describe various protocols used for communication over the Internet.
8. Explain the client/server concepts of Web.
9. What is a Web browser? Name any four.
10. is a language to create Web pages.
11. Justify the use of the Internet.
12. How will you retrieve data from the Web?
13. Name the essential features of Web browser.
14. Expand the following acronyms
 - (a) HTTP
 - (b) FTP
 - (c) URL
 - (d) WWW
 - (e) SMTP
15. How will you locate information on the Web?
16. What are the components of the Web? What is the communication protocol between them?
17. How a Web browser communicates with a Web server?
18. Illustrate the architectural model of WWW and illustrate the steps involved in accessing a Web page.

LESSON-3

HTML

INTRODUCTION:-

HTML (the Hypertext Markup Language) is a predominant markup language. It describes the structure of text-based information in a document by identifying some texts as heading, paragraph, lists, etc. It can also describe the appearance and semantics of a text and takes it easy for any browser to render that page into the computer screen. In HTML, text is marked up with elements delineated by tags that are keywords contained in pairs of angled brackets. Furthermore, HTML tags are not case sensitive. The absence of a closing tag for elements that require them is considered as a syntax error and can grossly affect the formatting and look of a page.

HTML documents can be created with any text editor for Front Page Express. All HTML files have either the .htm or the.html file name extensions.

SGML

Standard Generalized Markup Language (SGML) is an international standard for the description of marked – up electronic text. Electronic text is textual and / or multimedia information which has to be marked up to enable it to send through an electronic channel. SGML was made international standard by ISO in October 1986.

SGML is a meta language which in itself does not define a markup language, but provides a framework to develop various kinds of markup language. A markup language is one that specifies what markup is allowed, how it is differentiated from text and what markup is required. In short, it is a language to represent document structure but is not a document structure itself.

Hyper Text Markup Language (HTML) developed by Web community and Wireless Markup Language (WML) developed by WEBFORUM are representative markup language based on SML.

Every markup language defined by SGML is called SGML application . An application is characterized by

1. Declaration: specifies character and delimiters that appear in the application.
2. Document Type Definition (DTD): defines the format of document constructs.
3. Specification that tells about the semantics of the defined markup.
4. Document instances that contain markup and data.

DTD

It is the most important contribution of SGML. Consider the example given below:

```
<!DOCTYPE document [
<!ELEMENT document - - (header, body) >
<!ELEMENT header - o (( title & author) & abstract? ) >
<!ELEMENT body - o ( block*; ack)>
<!ELEMENT title - o ( # PCDATA ) >
<!ELEMENT authors - o ( # PCDATA )>
<!ELEMENT abstract - o ( # PCDATA ) >
<!ELEMENT block - o ( ( # PCDATA link ) * ) >
<!ELEMENT link - o ( # PCDATA ) >
<!ELEMENT ack - o ( # PCDATA ) >
<!ATTLIST link url CDATA # REQTRED >
```

The below is an example document based on the above DTD.

```
< document >
< header>
<title> Introduction to SGML
<authors> Ramesh and Puri
<abstract> A short description on SGML and its constructs
<body>
<block> Standard Generalized Markup Language is popularly
        called as SGML
<block> The International standards organization has declared
        it as a standard in the year 1986
<block> The information of SGML can be found at <link
        url=http://www.foo.com></link>
<ack> The following is acknowledged by peer authors
< / document>
```

The DTD is defined with the special delimiters angled brackets and known keywords. The first statement in the DTD is the declaration. The second statement is the declaration of an element in the document.

DTD Elements

The ELEMENT keyword is used to specify the element of tag that should be present in the document structure. The ELEMENT keyword is followed by

1. name of the element
2. two characters that specify the minimization rules
3. a content model

The minimization rule determines the presence of start and end tag. They are specified as a pair of characters either a hyphen or a letter O (Omitted or Optional) separated by a blank space. The hyphen indicates that the tag must be present and the letter O indicates that it may be omitted.

```
<!ELEMENT document - - (header,body)>
```

In the above example, declaration element should contain both start and end tag.

The content model specifies the legitimate occurrence of element in the document structure.

```
<!ELEMENT authors -O (#PCDATA)>
```

The element authors is specified as #PCDATA, an abbreviation for parsed character data tells that the element may contain any valid character data.

The declaration also contains occurrence indicator plus(+), question mark(?), and asterisk(*). The plus sign indicates that the element may occur for one or more times. The question mark means that the elements may almost one and possibly no occurrence. The asterisk indicates that the element may either be absent or appear one or more times.

```
<!ELEMENT body -O (block*, ack)>
```

In the example given, the element block may appear for zero or more times in the document.

The declaration in DTD also contains group connectors, represented by comma, vertical bar and ampersand. The group connectors are used to specify the order of elements. If comma is specified, the components it connects must appear in the order. The ampersand is similar to comma specification but in any order. The vertical bar indicates that only one of the components it connects will appear.

```
<!ELEMENT document - - (header, body)>
```

In the above example, both header and body should appear in the order specified in the document.

Attributes

Attributes are used to describe information for a particular element. For example,

```
<link url=http://www.foo.com></link>
```

Like elements, attributes are defined and declared in DTD with similar syntax.

The declaration begins with ATTLIST keyword followed by name of the element to which the attributes should be added. The declaration then contains the following components:

1. name of the attribute
2. type of attribute value or set of possible values.
3. how a parser should interpret the absence of attribute concerned. This is emphasized by the following keywords.

REQUIRED

A Value must be specified

IMPLIED

Value need not be given.

FIXED

Fixed Value must be given.

Example:

```
<! ATTLIST link url CDATA # REQUIRED>
```

```
<! ATTLIST ack status (sent not sent) # REQUIRED)
```

In the first example, the value for url attribute of link element must be specified and the value should be of character data. The second example declares an attribute status for ack element the attribute can have the value either as sent or not sent.

The following are some of the declarations in HTML DTD.

```
<! ELEMENT UL - - ( LI ) + >
```

```
<! ELEMENT IMG - O EMPTY >
```

```
<! ATTLIST MAP
```

```
name CDATA # IMPLIED
```

```
>
```

OUTLINE OF AN HTML DOCUMENT

The HTML tag is a keyword used to specify the appearance of the Web page. Every tag should be enclosed with angled brackets and should have closing tag like `<html> . . . </html>`. The code below shows an outline of an HTML document.

```
<html>
<head >
<!-- header part -->
<title > An example </title >
</head >
<!-- body section -->
<body >
    <h1 > Hello World </h1 >
</body >
</html >
```

If this code is interpreted, the browser will display the following Web page (Figure 4. 1).

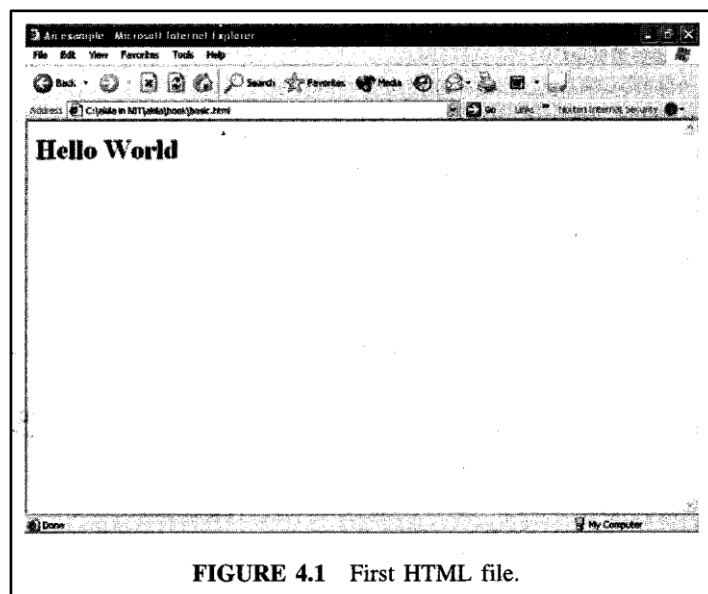


FIGURE 4.1 First HTML file.

Every html document has 2 sections: header and body. The header section begins with `<head >` and ends with `</head >`. It contains the information about the document, which is not visible to the user. It is mandatory to include a title in the header section, which gives a name to the Web page. The title appears on the bar at the top of the browser window. The search engines use the title for indexing

purposes. The other components in header section are prologue, link, meta, script, base, style and ink. The body part contains the actual page content. There are various tags under the body tag. The comments are enclosed in the special tags `<!-- comment -->`.

HEAD SECTION

Prologue

The prologue component is only a comment which tells about the HTML version.

```
<! Doctype HTML 4.0 >
```

Link

The link tag is used to inform the browser's previous and next document, to link the banner and to inform about the location of the base document.

```
<Link rel = Previous href = "prev.html">
```

```
<Link rel = "next href = " next.html">
```

```
<Link rel = banner href = "bann.gif">
```

Base

This tag declares global reference values for href and target attribute. The syntax is

```
<base href ="base url " > or <base target=" frame name" >
```

The attributes href specifies the reference URL, that is used to help compute relative URLs. The attribute target specifies the default frame name to which all links are targeted.

Meta

The meta tag defines the document's meta information such as keywords, expiry date, author, page generation software used, etc. It also supports client pull, a dynamic technique in which the browser loads a new document after a specified delay. For example.

```
<meta name = "author" content = "yyy">
```

```
<meta name = "keywords" content = "web technology, HTML, CGI">
```

Script

The script tag contains the code reference in the body of the document. The syntax is

```
<Script language = "javascript">  
----- Script code  
</Script>
```

BODY SECTION

As described earlier, the actual contents of the Web page are placed in the body section. The tag consists of various attributes. They are listed below

- bgcolor : specifies the background color of the Web page. It can be specified as bgcolor = "red"
- Link : specifies the color of the unvisited link.
- Alink: specifies the color of the active link.
- Vlink: specifies the visited link color.
- Text: specifies the text color.

For example,

```
<body bgcolor="green" alink="blue" vlink="yellow">
```

The document's contents can be placed between various tags for special appearance. Some of those tags are discussed in the following sections.

Headers

These are formatting tags to vary the text size. There are six header tags H1 through H6. Figure 4.2 shows the output screen of the following HTML code.

```
<html>  
<head>  
<title> header examples </title>  
<body>  
<center>  
  <h1> Hello World </h1>  
  <h2> Hellp world </h2>
```

```

<h3> Hello world </h3>
<h4> Hello world </h4>
<h5> Hello world </h5>
<h6> Hello world </h6>
</body>
</html>

```

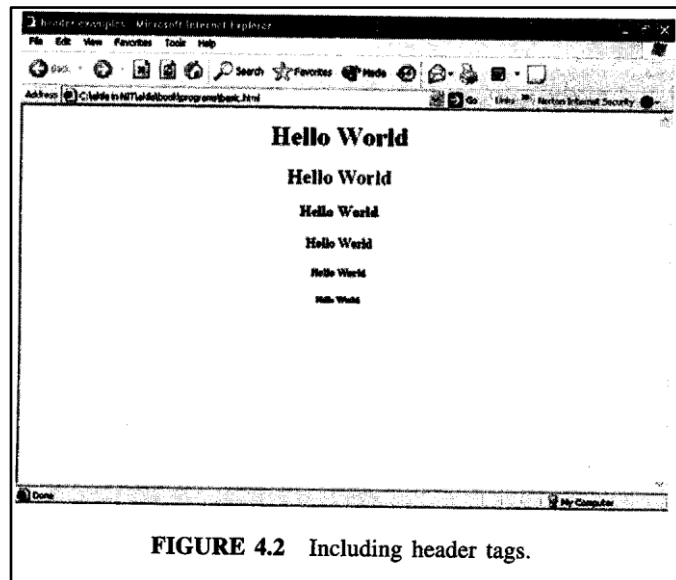


FIGURE 4.2 Including header tags.

The `<center>` tag is used to align the text following the tag, in the centre. The attribute for the header tag align [left, center, right] is used to align the header text as left justified or as centred or as right justified. For example, `<h1 align = left >`, `<h2 align=center>` or `<h6 align=right>`

Paragraphs

The tag which is used to start a paragraph is `<p>...</p>`. This tag also consists of align attributes with the same values. To insert a line in between, the `
` tag is used. An example is shown in Figure 4.3. These tags are called block level tags.

Following is an example of the use of these tags.

```

<html>
<head>
<title> header examples </title>
</head>
<body>
  <h1> Hello world </h1>

```

```

<p> The world is so beautiful. It contains a lot of beautiful and
wonderful creatures. The Most beautiful creation of the
world is the human kind. All the inner operations of the
human being are such wonderful things which cannot be
imitated by any scientific invention.
<br> Even the computer scientists tried to imitate the brain of the
human and later concluded that the full operations of the
neural activity cannot be simulated.
</p >
</body>
</html>

```

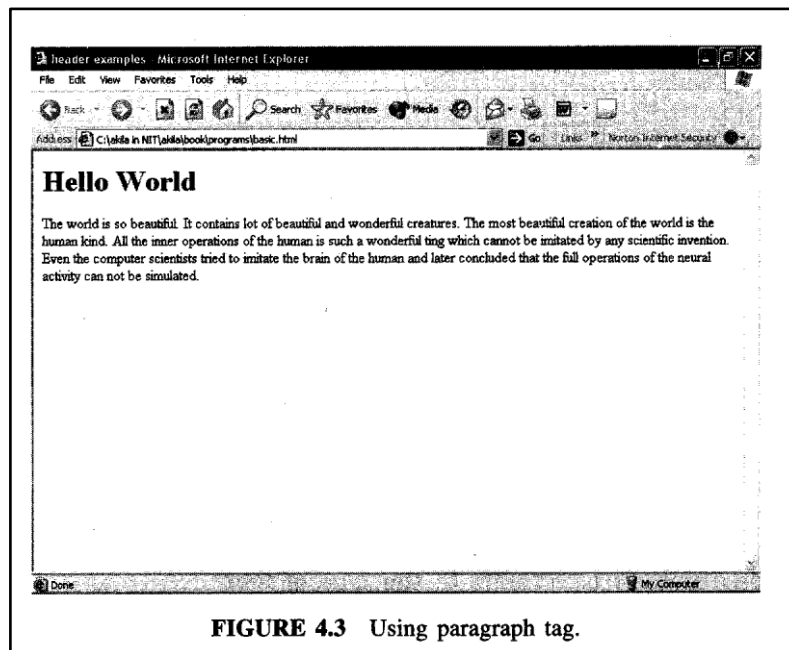


FIGURE 4.3 Using paragraph tag.

Text Formatting

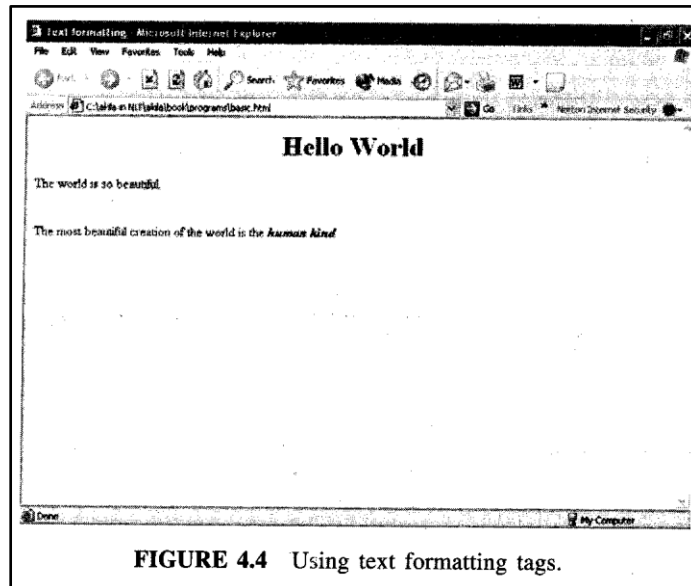
The page content in the body of the Web page can be presented in a neat way using different tags like ``, ``, ``. The `` tag is used to make the text in italics. The `` tag is used to make text as bold. An example is shown in Figure 4.4.

```

<html>
<head>
<title> Text formatting </title>
</head>
<body>
<h1 align = center> Hello World </h1>
<p> The world is so beautiful. </p>

```

```
<br>
<p> The most beautiful creation of the world is the
    <strong><em>human kind</strong></em>
</p>
</body>
</html>
```



The tag is also used for text styling. This tag contains various attributes like face, color and size. An example is given below.

```
<p> <font color="red" size="+1" face="Arial"> ...</font> </p>
```

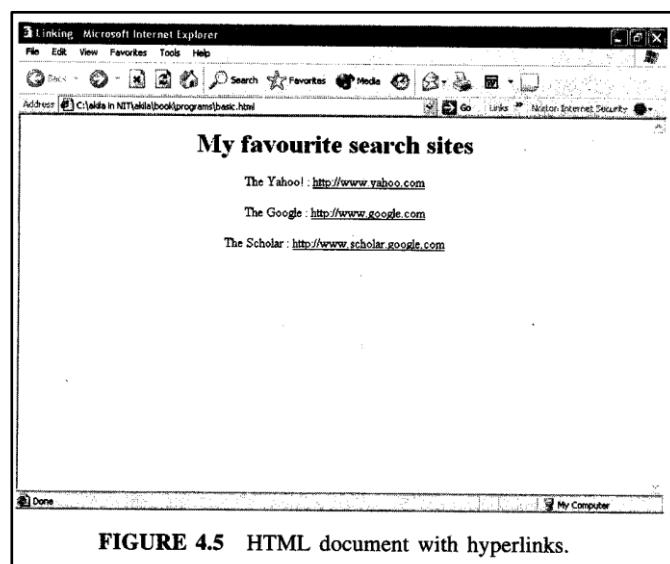
The color attribute indicates the color of the text, size attribute is used to change the size of the text being formatted. To make the text larger, set size="+x" and similarly set size="-x" to make the text smaller, where x is the number of font point sizes. The attribute face refers to the type of font.

LESSON-4

Linking

The most important capability of the HTML is its ability to create hyperlinks to documents and thereby make the World Wide Web as a collection of linked documents. The links are created using anchor `<a>...` element. This element requires an attribute to mark the location of the object to get linked. The address of the object is specified using href attribute. The href attribute of the tag tells the browser to get another html document on the Web. An example of how to link a Web page is given below (see Figure 4.5)

```
<html>
<head>
<title> Text formatting </title>
</head>
<body>
  <h1 align = center> My favourite search sites </h1>
  <p align=center> The Yahoo ! : <a href=http://www.yahoo.com>
http://www.yahoo.com </a> </p>
  < p align =center >The google : <a href =htt[: //www.google.com "
  http:// www.google .com </a> </p>
  <p align=center > The Scholar : <a href="http://
  www.scholar.google.com">
  http:// www.scholar.google.com </a> </p>
</body>
</html>
```



The anchor tag can also be used with other protocols like mailto and FTP. The format is

```
<a href = mailto:emailaddress> ...<a>
```

Internal linking

In the previous section. it is discussed on how to create links to external document HTML has facilities to include internal links also by assigning a location name to any individual point in an HTML document. The location name can then be added to the page's URL. For internal linking also. the anchor tag is used. To create name, the following tag is used

```
<a name="name"> </a>
```

To refer to the location use

```
<a href = "# name"> text </a>
```

The following example illustrates the use of <a> as internal linking (see Figure 4.6).

```
<html>
<head>
<title> Internal Linking </title>
</head>
<body>
  <h1 align = center > My Book </h1>
  <p> My book contains the following chapters </p>
  </br>
    <a href="#chap 1"> 1. Introduction </a><br>
    <a href="#chap 2"> 2. Topics 1 </a><br>
    <a href="#chap 3"> 3. Topics 2 </a><br>
    <a href="#chap 4"> 4. Topics 3 </a><br>
  <a name = "chap 1"></a>
  <h3 align ="center"> Chapter 2 </h3>
  <p> the text is here </p><br>
  <p> the text is here </p><br>
  <p> the text is here </p><br>
```

```

<p> the text is here </p><br>
<p> the text is here </p><br>

<a name = "chap 2"></a>
<h3 align = "center"> Chapter 2 </h3>
<p> the text is here </p><br>
<p> the text is here </p><br>
<p> the text is here </p><br>
<p> the text is here </p><br>
<p> the text is here </p><br>
<a name = "chap 3"></a>
<h3 align = "center"> Chapter 3 </h3>
<p> the text is here </p><br>
<p> the text is here </p><br>
<p> the text is here </p><br>
<p> the text is here </p><br>
<p> the text is here </p><br>
<a name = "chap 4"></a>
<h3 align = "center"> Chapter 4 </h3>
<p> the text is here </p><br>
<p> the text is here </p><br>
<p> the text is here </p><br>
<p> the text is here </p><br>
<p> the text is here </p><br>
</body>
</html>

```

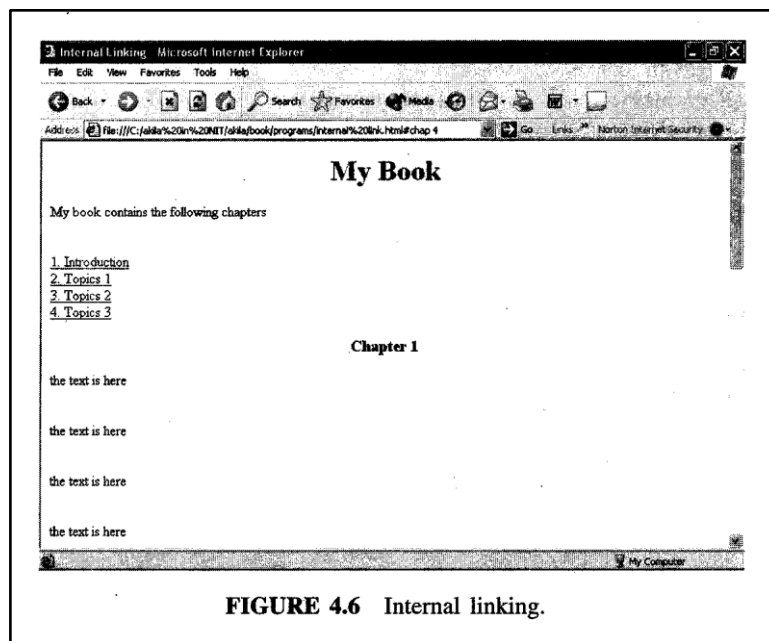


FIGURE 4.6 Internal linking.

Embedding Images

Images can be embedded into the Web page. To add images as a background for the whole Web page, the following format is used.

```
<body background= "bg.gif" >
```

To include the image as a normal figure as a normal figure in the Web page the tag is used. The image tag has the following attributes.

- Src
- Height
- Width
- border
- Align
- Alt

The src attribute is used to specify the location of the image, height and width, specify the size, and another important attribute is alt which is provided for browsers that have images turned off or that cannot have images. The value of alt attribute will appear on screen in place of the images. An example for embedding images follows.

```
<html>
<head>
<title> Embedding images </title>
</head>
<body>

    <h1 align =center> My favourite search sites </h1>
    <p align = center>. The Yahoo! :
    <a href=http://www.yahoo.com>
http://www.yahoo.com </a> </p>
    <p align =center> The google :
    <a href=http://www.google.com>
http://www.google.com </a> </p>
    <p align = center> The Scholar : <a href = "http://
www.scholar.google.com">
http:// www.scholar.google.com </a></p>
</center>
```

```


</center>
</body>
</html>

```

The output of the above code is rendered as in Figure 4.7

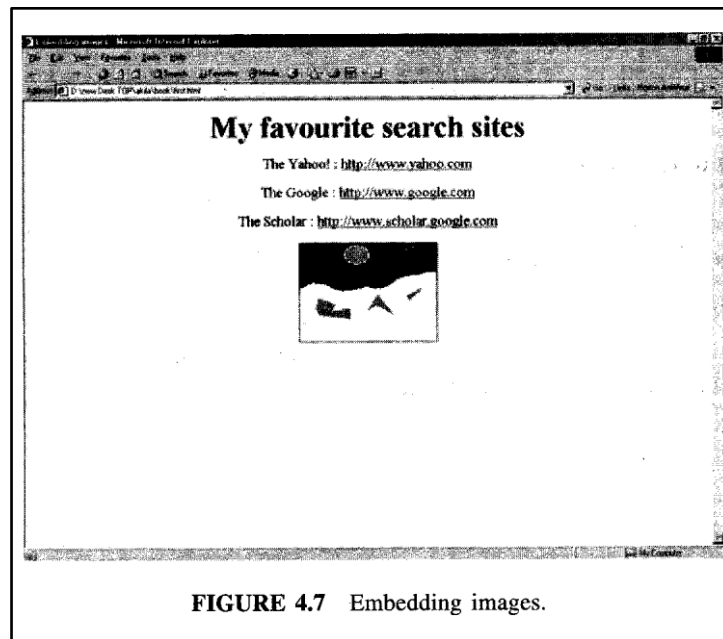


FIGURE 4.7 Embedding images.

The other related tag is `<map>` which is used to create hot spot. The syntax is

```
<map name="name">
```

```
.....
```

```
</map>
```

For example, the following code creates a hot spot of rectangular shape and has the hyperlink in the coordinates mentioned.

```

<map>
<area shape="rect"
      cords="23,45,56,89"
      href = "source.html">
</map>

```

Lists

Lists are a way of creating several kinds of outlines, menus, etc. There are three different kinds of lists to choose from: ordered, unordered, and definition. These tags are used to present the context in a readable form.

Unordered

For including unordered list of items start with ``. and end with ``. The items inside the `` tag should be included within ``. The `` tag has an attribute `type` with values like `disc`, `circle`, or `square`. For example,

```
<html>
<head>
<title> Unordered Listing </title>
</head>
<body>
  <h1 align = center> My favourite search sites </h1>
  <ul>
    <li type="circle"> The Yahoo! : <a href="http://
      <u>www.yahoo.com</u>">
      http://www.yahoo.com </a> </li>
    <li type="circle"> The Google : <a href="http://
      www.google.com">
      http://www.google.com </a> </li>
    <li type="circle"> The Scholar : <a href="http://
      <u>www.Scholar.google.com</u>">
      http:// <u>www.Scholar.google.com</u> </a> </li>
  </ul>
</body>
</html>
```

Ordered:

For including ordered list of items start with `` and end with ``. It has attributes like `compact`, `start`, `type`, etc. The attribute `compact` specifies the display compact form, `start` indicates number to start at and `type` specifies the kind of numbering to use like. 1: numeric, a, A: lower and uppercase alphabetic. i, I: lower and uppercase roman. The items should get tagged within `` and `` having the attributes as value which says what number to use now and `type` which specifies the kind of numbering to switch to. An example is shown below.

```
<ol type=I>
  <li> this is in roman, a heading.
<ol type=1>
  <li> subheadings are numeric.
  <li> another numeric subheading.
<ol type=a>
  <li> subsubheadings are alpha.
  <li> another one.
  <li> and another one.
</ol>
  <li> another subheading.
</ol>
  <li> another heading.
</ol>
```

The result of the above code is

- I. this is in roman, a heading.
 1. Subheadings are numeric.
 2. another numeric subheading.
 - a. subsubheadings are alpha.
 - b. another one.
 - c. and another one.
 3. another subheading.
- II. another heading

The numbering in ordered lists can be controlled with start values or individual item numbers.

```
<ol start=9>
<li> This list starts at number 9.
<li> This should be 10.
<li> This should be 11.
<li value=1> This should be 12, but set it to 11.
<li> counting .....
</ol>
```

The result is

9. This list starts at number 9.
10. This should be 10.
11. This should be 11.
1. This should be 12, but set it to 11.
2. counting.

Definition lists:

Specifically used for lists in which each element is labelled with a word rather than a bullet or number. It should start with `<dl>` and end with `</dl>`. The items are included within pairs: `<dt>` and `<dd>` where `<dt>` to include term and `<dd>` to include actual definition.

```
<dl>
<dt> start tag
<dd> the symbol indicating the start of a region of interest.
<dt> end tag
<dd> a symbol indicating the end of that region.
<dt> content
<dd> the text between the start and end tags.
</dl>
```

The above code will present the content as the following:

```
start tag
    the symbol indicating the start of a region of interest.
end tag
    a symbol indicating the end of that region.
content
    the text between the start and end tags.
```

Tables:-

This form tag is the HTML's best of arranging information in space, and controlling layout. It allows specification of arrangement of cells but does not allow cell content to be aligned as flexibly as one would sometimes like.

The syntax is

- `<table> .. </table>`
 - . `align = left, center, right`
 - . `border`: make a border around the table and its cells.
- Within this, `<tr> .. </tr>` specifies a table row.
 - . `align= left, center, right` applies to cells.
- Within this, `<td> .. </td>` specifies a table cell, while `<th> .. </th>` specifies a table heading.
 - . `align= left, center, right` applies to cells.

A simple example is shown below.

```
<table>
<tr>
  <td>row=1,col=1</td>
  <td>row=1,col=2</td>
  <td>row=1,col=3</td>
</tr>
<tr>
  <td>row=2,col=1</td>
  <td>row=2,col=2</td>
  <td>row=2,col=3</td>
</tr>
<tr>
  <td>row=3,col=1</td>
  <td>row=3,col=2</td>
  <td>row=3 ,col=3</td>
</tr>
</table>
```


The result of the above code will look like:

```
row=1,col=1 row=1,col=2 row=1,col=3
row=2,col=1 row=2,col=2 row=2,col=3
row=3,col=1 row=3,col=2 row=3,col=3
```

```
<table border=1>
<tr>
  <th align=right>Head1</th>
  <th align=center>Head2</th>
  <th align=left>Head3</th>
</tr>
<tr>
  <td align=right>A1 </td>
  <td align=center>A2</td>
  <td align=left>A3</td>
</tr>
<tr>
  <td align=right>A4</td>
  <td align=center>A5</td>
  <td align=left>A6</td>
</tr>
<tr>
  <td align=right>A7</td>
  <td align=center>A8</td>
  <td align=left>A9</td>
</tr>
</table>
```

Head1	Head2	Head3
A1	A2	A3
A4	A5	A6
A7	A8	A9

We can use an attribute called colspan and rowspan. For example, <td colspan=2> will take up two columns and <td rowspan=2> will take up two rows.

For example.

```
<table border=1>
<tr>
  <td colspan=2 align=center>Report</td>
</tr>
<tr>
  <td rowspan=2>football commentators
  <td>good taste=5%</td>
</tr>
<tr>
  <td>bad taste=95%</td>
</tr>
<tr>
  <td rowspan=2>other
  <td>good taste=90%</td>
</tr>
<tr>
  <td>bad taste=10%<td>
  <td>test</td>
</tr>
</table>
```

Report		
football commentators	good taste=5%	
	bad taste=95%	
other	good taste=9%	
	bad taste=10%	test

Instead of items, images can also be used as items with the same `` tag.

Frames

Framed layout is the one in which the browser window is broken into multiple regions called frames. Each frame can contain different HTML documents. The `<frameset>` tag is a container for frames and replaces the body tag and `<frame>` tag is used to place the contents into the frame.

```
<frameset rows="value" cols="value">
```

The above tag takes attribute rows or cols. If the window is to be divided in horizontal stripes then the row attribute should be used. If in the vertical stripes, then cols attribute should be used. Both the attributes should not be used at the same time. For example, the code shown below divides the window into two regions in row wise.

```
<frameset rows ="30%, 70%"
```

The frame tag is used to place different HTML documents in each frame.

```
<frame src="URL"  
      name="name"  
      frameborder="0 | 1"  
      scrolling="Yes | no | auto">
```

Another tag is `<noframes>` used in the case where the browser does not support frames. If the browser does not support, the content placed between `<noframes>` tag is displayed.

An example is given to demonstrate the use of frames (see Figure 4.8).

```
<html>  
<head>  
<title> Frames </title>  
</head>  
<frameset cols+"40%,*">  
<frame src="list.html" >  
<frame src="table.html" >  
< /frameset>  
<noframes>
```

provides an alternative text of the browser does not support frames

```
</noframes>  
</html>
```

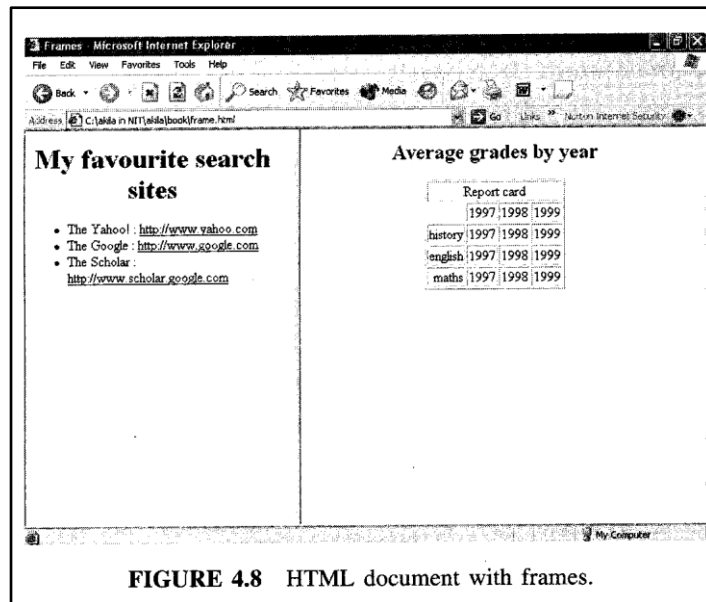


FIGURE 4.8 HTML document with frames.

Other Special Tags and Characters

All special characters are inserted in their code form. The format is `& code`. An example is `& amp` is used to insert an ampersand. The tag `` is used to strike through text with a horizontal line. Similarly, the tags `<sub>` and `<sup>` are used to turn the text into subscript and superscript respectively.

To draw a horizontal line in the Web page `<hr>` tag is used. This tag has the following attributes:

- Width-to specify the width of the line.
- Size-to specify the height of the line.
- Noshade-eliminates the default shading effect and instead displays the horizontal rule as a solid color bar.

```
<html>
<head>
<title> Special tags </title>
</head>
<body>
  <h3 align = center> All Information on this & copy of XYZ
    & amp companu,
    2005. </h3>
  <br>
  <p><del> To strike through the text use the del element
  </del></p>
```

```

<p> Use <sup>sup </sup> element to turn text into superscript
  </p>
<p>Use <sub>sub  element to turn text into subscript
  </p>
<p>Use <big>Big </big> element to turn text into big text and
  </p>
<p>Use <small>small </small> element to turn text into small
  text </p>
<br>
<hr eidth="25%" size=8>
<br>
<hr noshade width="50%" size=5 align="centre">
</body>
</html>

```

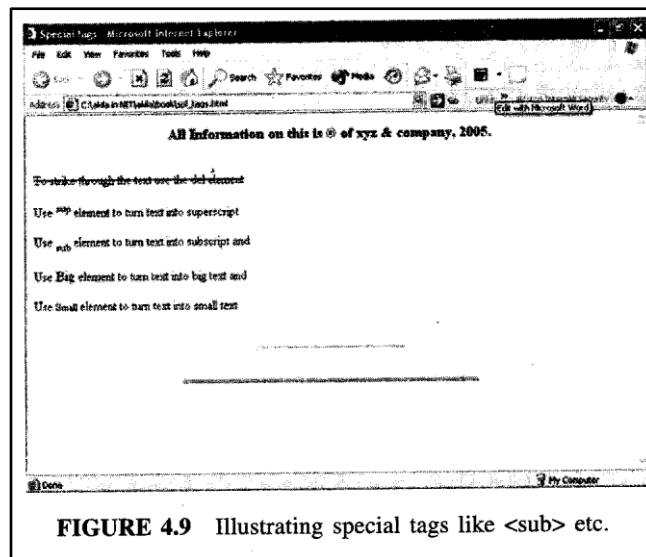


FIGURE 4.9 Illustrating special tags like <sub> etc.

See Figure 4.9 for the output of the HTML code above. There are other block level tags like <blockquote> and <div> apart from <p>, <center>,
. <blockquote> is the tag that contains the quoted text that is to be displayed indented from the regular body text. The <div> tag is used to define a section or division of a document that requires a special alignment. The syntax is,

```

<div align="left|right|center|justify">
-----
</div>

```

The above said tags are used to format the text in blocks.

LESSON-5

HTML FORMS

There are a special kind of tags called form tags which are used to create interactive content. In all the above discussed tags, user cannot give inputs or press buttons. These form tags are used to collect information from the user and then a script or program on a Web server uses that information to compute a custom response to the form submission. A typical example is the form filled during the creation of a mail account. The syntax is

```
<form action="URL_of processing script"
      method"get |post"
      target="frame_name"
      enctype="MIME type of file to upload"
      Accept_charset="acceptable char ser"
      Accept="acceptable MIME TYPES">
```

For example, the following form tag can be used in the HTML code.

```
<form action="shopping.cgi" method="post"
      target="response_frame">
-----
</form>
```

The method attribute is used to send the form data to the Web server. The default attribute value for method is get which appends the data to the end of the processing script URL. If the method has the value post, then the data is sent to the Web server as a separate transaction. The value post is used when the form data is to be stored in the database or as a processing data in the Web server.

The following are some of the tags which should be used only within the form tag.

Input:

This tag places the following form controls.

- Text, password or hidden fields
- Check boxes
- Radio buttons
- Image based buttons
- Scripted buttons
- Submit or reset buttons

The syntax is

```
<input type="text | passwd" name="name" value="default_val"
      size="field size"
      maxlength="mas_inlength" DISABLED READONLY>
or
<input type="hidden" name="fieldname" value="default_val">
or
<input type="checkbox" name="fieldname" CHECKED DISABLED>
or
<input type="radio" name="fieldname" CHECKED DISABLED>
or
<input type="button" name="fieldname" value="button_lable"
      onclick="button_lable" onclick="processing script" DISABLED>
```

Select

This form tag is used to set up a list of choices from which a user can select one or many. The syntax is

```
<select name="name" size="visible_rows" MULTIPLE DISABLED
      accesskey="shortcut access key" >
-----
</select>
```

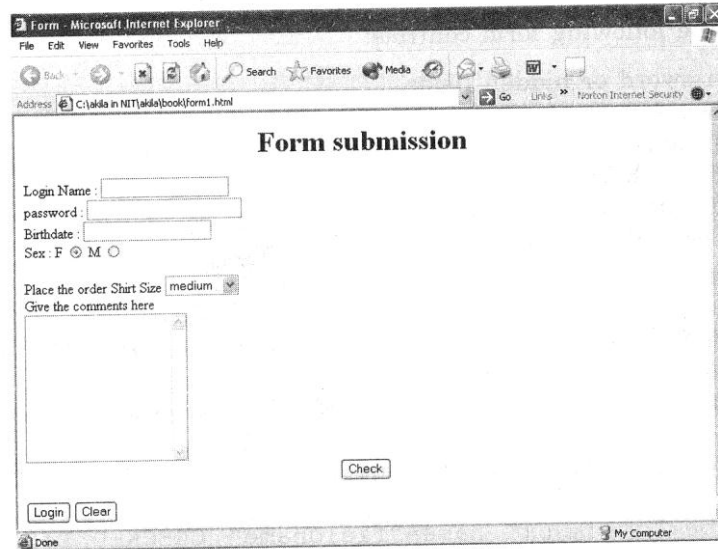
The attribute size is used to specify the number of rows to be visible in the list. The keyword MULTIPLE enables the user to choose more than one of the options by holding down the control key and clicking.

Text Area

This tag sets up a multiple line input text window (see Figure 4.10). The syntax is

```
<textarea name="name" rows="no of rows"
      cols="no_of_cols" DISABLED READONLY
      accesskey="shortcut-key" >
```

```
-----
</textarea>
```



The following code demonstrates the use of the various tags of the form (see figure 4.10).

```
<html>
<head>
<title> form </title>
</head>
<body>
  <h1 align = center> Form submission </h1>
  form action=" /cgi-bin/submit.cgi" method="post">
  Login Name:
  <br>
  Password:
  <input type="text" name="login" size=20>
  <br>
  Birth date:
  <input type = "text" name="login" size=20>
  <br>
  Sex: F
  <input type="radio" name = "sex" value="F" CHECKED>
<br>
<br>
```


<p> Place the order

Shift Size

```
< Select name="size"
<option >small </option>
<option> medium </option>
    <option> large </option>
<option> xtra large </option>
</select>
```


Give the comments here


```
<Textarea name="comments" rows=10 cols=20>
```

```
</textarea>
```

```
<input type="button" value="Check" onclick ="validate()">
```



```
<input type ="submit" value ="Login" onclick="submit()">
```

```
<input type ="reset" value="Clear" onclick="clear">
```

```
</form>
```

```
</body>
```

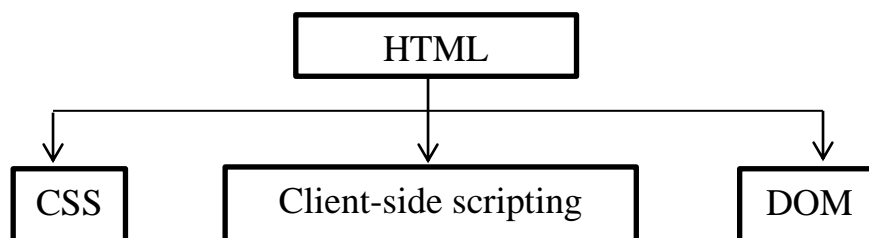
```
</html>
```

DYNAMIC HTML (DHTML)

INTRODUCTION

Dynamic HTML or DHTML is the name given to the technique of dynamically changing the rendering content of an HTML code. Traditionally, HTML is static, because images once defined cannot be changed or a pre-designed layout cannot be altered. In other words, the document coded in the HTML is not interactive and will not respond to the user actions. DHTML gives the designers to create visually outstanding HTML documents that directly interact with the user.

DHTML makes it possible for the Web pages to react to the user actions or to the changes. The images can be animated when the mouse moves over it or to certain parts of the document, and a separate design can be applied. These, however, are some of the uses of DHTML. With DHTML, we can hide text and images of a document for a given period of time, create tickers that automatically refresh the contents of a document with the latest data, create a form which can be readily filled up with data, which can be processed and accessed immediately. DHTML is not a technology; rather, it is the combination of the three different technologies, client-side scripting (JavaScript or VBScript), cascading style sheets and document object model.



The following list gives an idea of the contents of a dynamic Web page:

- animation, audio and video files
- page transitions, automatic refresh and forwarding.
- changing text(moving, scrolling, blinking, etc)
- visual change (in text, images, colors, etc.) in response to user actions
- automatically generated HTML
- automatically generated data
- forms processing (validation, calculation)

REVIEW QUESTIONS

1. Describe the structure of an HTML document.
2. Mention the elements located within `<head>` and `</head>` element.
3. What is the difference between tag and attributes? Explain with an example.
4. What is the significance of `<title>` tag in HTML?
5. What are the basic tags of HTML?
6. List the attributes that the `<Body>` tag can take.
7. Explain with examples the various text formatting attributes.
8. How will you create hyperlinks in Web documents? Discuss on different attributes of creating a hyperlink.
9. Describe how will you embed images in a Web document?
10. Describe `` tag.
11. How will you create unordered and ordered lists?
12. Explain in detail about the `<table>` tag. Also explain all its attributes and related tags with examples.
13. What do you mean by internal linking? How will you implement it?
14. What are image maps?
15. State the importance of `<meta>` tag?
16. Describe with examples the frame model of HTML document.
17. What is the important of `<pre>` tag?
18. Explain the various tags used to create HTML form with suitable examples.
19. What are the actions of the submit button in a form?
20. Mention the used of external and internal linking. Also mention how internal linking affects the crawling process of search tools.
21. Differentiate ROWSPAN and COLSPAN.
22. List the properties of form elements and the methods of form element.