

Web Basics – HTML5



Copyright © 2011 IGATE Corporation. All rights reserved. No part of this publication shall be reproduced in any way, including but not limited to photocopy, photographic, magnetic, or other record, without the prior written permission of IGATE Corporation.

IGATE Corporation considers information included in this document to be Confidential and Proprietary.

Document History

Date	Course Version No.	Software Version No.	Developer / SME	Change Record Remarks
1-Oct-09	1.0	1.0	Kumar Bramhadande	Initial Document
May 2011	2.0		Karthik M	Integration Refinements
20-May-2013	3.0		Vinay Gupta	Revamped according to new curriculum
31-Mar-2015	4.0		Rathnajothi P	HTML material upgraded from version 4 to 5

Keep this as a hidden slide.

Note to coordinators: Not to be printed for the class book.

Course Goals and Non Goals

➤ Course Goals

- Understand Static web page creation.
- Create Static web pages.



➤ Course Non Goals

- DHTML (not covered in this course).

Pre-requisites

- **None**

Intended Audience

- Novice User
- Developer



Day Wise Schedule

➤ Day 1

- Lesson 1: Introduction to Internet
- Lesson 2: HTML Basics
- Lesson 3: Tables
- Lesson 4: Lists
- Lesson 5: Working with Links
- Lesson 6: Image Handling
- Lesson 7: Frames

➤ Day 2

- Lesson 8: HTML Forms for User Input
- Lesson 9: New Form Elements

Table of Contents

➤ **Lesson 1: Introduction to Internet**

- Understand the history of Internet.
- Understand Web terminology.
- Understand IP addresses
- TCP/IP Protocol
- Domain Name System
- HTTP Protocol
- Servers – Web Servers
- Web Browsers
- Working of WWW
- HTML – Static and Dynamic Web Pages

➤ **Lesson 2: HTML Basics**

- Understand the structure of an HTML page.
- New Semantic Elements in HTML 5
- Learn to apply physical/logical character effects.
- Learn to manage document spacing.

Table of Contents

➤ **Lesson 3: Tables**

- Understand the structure of an HTML table.
- Learn to control table format like cell spanning, cell spacing, border

➤ **Lesson 4: List**

- Numbered List
- Bulleted List
- Directory List
- Glossary List

➤ **Lesson 5: Working with Links**

- Understand the working of hyperlinks in web pages.
- Learn to create hyperlinks in web pages.
- Add hyperlinks to list items and table contents.

Table of Contents

- **Lesson 6: Image Handling**
 - Understand the role of images in web pages
 - Learn to add images to web pages
 - Learn to use images as hyperlinks
- **Lesson 7: Frames**
 - Understand the need for frames in web pages.
 - Learn to create and work with frames.
- **Lesson 8: HTML Form for User input**
 - Understand the role of forms in web pages
 - Understand various HTML elements used in forms.
 - Single line text field
 - Text area
 - Check box
 - Radio buttons
 - Password fields
 - Pull-down menus
 - File selector dialog box

Table of Contents

➤ **Lesson 9: New Form Elements**

- Understand the new HTML form elements such as
 - date
 - number
 - range
 - email
 - search
 - datalist
- Understand audio, video, article tags

References

➤ Books

- Head First HTML with CSS
- HTML 5 in action
- HTML 5 and CSS 3
- Html Pocket Reference

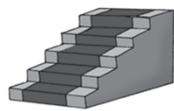


➤ Links

- <http://imboldc.ucc.ie/~pflynn//books/htmlcard.html>
- <http://www.w3.org/MarkUp/>

Next Step Courses (if applicable)

- Working with any Scripting Language
- Working with Dynamic HTML



Other Parallel Technology Areas

- None

Web Basics – HTML5

Lesson 1. Introduction to the Internet

Lesson Objectives

➤ After completing this module you will be able to:

- Understand the history of Internet.
- Understand Web terminology.
- Understand IP addresses
- TCP/IP Protocol
- Domain Name System
- HTTP Protocol
- Servers – Web Servers
- Web Browsers
- Working of WWW
- HTML – Static and Dynamic Web Pages



What is Internet?

➤ **Internet:**

- ‘Network of networks’ or “world’s largest network”.
- A concept, like the economy.
- Collection of inter-networked regional networks.
- Not owned by anyone.
- Based on TCP/IP.

June 5, 2015

Proprietary and Confidential

- 3 -

IGATE
Speed.Agency.Imagination

Internet:

Internet is a network of networks in which many machines are interconnected with each other using TCP/IP protocol suite.

No one person runs the Internet, and no single organization pays all the costs; there is no Internet Corporation. It exists as a result of the cooperation from people all over the world, who work in various types of organizational and computing environments. Internet never closes down, mostly because of its decentralized structure. Today, it is reliable and predictable. Individual servers may close for upgrades or hardware replacement, but the network is always available without interruption.

Services offered by internet are:

Communication : Users can easily communicate with each other at any time from anywhere. For an Example, Email Communication, GTALK, etc..

Data Transfer : Users can transfer file in the format of picture, word document, pdf, etc.. as an attachment in the E-Mail. Also files are possible to be shared through FTP servers

1.2: History of Internet

History of Internet

- Need to share information.
- Advanced Research Projects Agency (ARPA)
 - ARPANET
 - Comprised individual packet switching computers interconnected by leased lines.

June 5, 2015

Proprietary and Confidential

- 4 -

IGATE
Speed.Agency.Imagination

History of Internet

Internet was established more than 25 years ago to meet research needs of the U. S. defence industry. However, it has grown into a huge global network serving universities, academic researchers, government agencies, and commercial interests, both in the United States and in more than 100 other countries.

In 1969, Advanced Research Projects Agency (ARPA) of the U. S. Department of defense established ARPAnet, an experimental four-computer network, so that research-scientists could communicate among themselves. By 1971, ARPAnet comprised almost two dozen sites. By 1974, that number grew to 62, and by 1981, it comprised more than 200 sites.

As more and more computers using different operating systems were connected, the need for a common communications protocol became apparent. Theory required that any computer on the network should be able to talk to any other computer, as a peer.

1.3: Internet Basic Definition

Internet – Basic Definitions

- **Internet Service Provider (ISP)**
- **Network Information Center (NIC)**
- **Internet Address**
- **Internet Domain Name**
- **Routing**
- **Gateways**
- **Protocols**
 - TCP/IP
 - HTTP

June 5, 2015

Proprietary and Confidential

- 5 -



Internet Service Provider (ISP): Internet Service Providers (ISPs) are companies that enable user to connect to the Internet . They offer an account on their systems and access to the Internet.

Network Information Center (NIC): Network Information Center (NIC) assigns and regulates IP addresses on the Internet. You can get one directly from the NIC, or you can ask your ISP to secure an IP address on your behalf.

Internet Address: TCP/IP requires each host on a TCP/IP network have their own unique IP address.

Internet Domain Name: A domain name maps or translates the actual numeric IP address used for your Web server into an easy-to-remember alphanumeric name. Domain refers to a collection of network host computers, known by the same name. Your domain name should reflect your organization or corporation, for example, .com , .edu , .gov , .int , .mil , .net, and so on.

Routing: Process of getting your data from point A to point B.

Gateway: Forwards datagrams to a destination if it knows where the destination is.

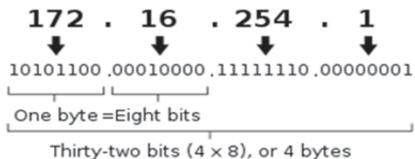
Protocol: specify interactions between the communicating entities. Example for protocol are TCP/IP, HTTP, telnet, etc..

1.4: Internet Address

Internet Address

- Every device (eg: computer, printer) that participates in a computer network is assigned a numeric label called as Internet Protocol address (IP address).
- The designers of the Internet Protocol defined an IP address as a 32-bit number and this system is known as Internet Protocol Version 4 (IPv4).
- IP addresses are binary numbers, but they are usually stored in text files and displayed in human-readable notations, such as 172.16.254.1
- IP Address is a 32-bit address, in the form of x.x.x.x.

An IPv4 address (dotted-decimal notation)



1.4: Internet Address

Internet Address (Contd...)

➤ Internet address classification

- Class A N.H.H.H Used for very large networks
- Class B N.N.H.H Used for medium sized networks
- Class C N.N.N.H Used for smaller networks
- Class D Multicast Address
- Class E Reserved for future use.

➤ Addresses that start with a value between:

- 1 and 126 Class A (First bit value is 0)
- 128 and 191 Class B (First two bit values are 10)
- 192 and 223 Class C (First three bit values are 110)
- 224 and 239 Class D (First four bit values are 1110)
- 240 and 255 Class E (First five bit values are 11110)

June 5, 2015

Proprietary and Confidential

- 7 -



Internet Address Classification:

Internet address is classified into 5 classes such as Class A, B, C, D and E. IP address start and end value will vary based on classification. IP address for each classification is described using N and H.

N stands for Network (Assigned by the NIC)

H stands for Host (Assigned by the local administrator)

IP address classification : start and finish address

Class	Start address	Finish address
A	0.0.0.0	127.255.255.255
B	128.0.0.0	191.255.255.255
C	192.0.0.0	223.255.255.255
D	224.0.0.0	239.255.255.255
E	240.0.0.0	255.255.255.255

Depends on classification, IPAddress can be in the range value between start address and finish address.

1.5: TCP/IP Basics

TCP/IP Basics

➤ **Transmission Control Protocol (TCP):**

- Connection-oriented transport layer protocol.
- Sets up a connection between the sender and receiver.
- Uses the services of IP to send and receive data.
- Re-orders received information.

June 5, 2015

Proprietary and Confidential

- 8 -

IGATE
Speed.Agency.Imagination

TCP (Transmission Control Protocol) is a connection-oriented protocol. TCP ensures that data arrives and that it arrives in the correct order. It reorders information that is received out of order and requests the information that is not received is to be resent.

Internet uses TCP/IP to link computers. TCP/IP stands for Transmission Control Protocol/Internet Protocol, which are two significant parts of what is now also known as the Internet Protocol Suit.

The feature that makes TCP/IP different from many other networking protocols is that it was designed to link networks instead of simply linking computers in a network.

1.6: Domain Name System

Domain Name System

- **Computers work best with numbers.**
 - Synonymous to identifying people with names.
- **Domain Name:**
 - Maps or translates the actual numeric IP address into an easy-to-remember alphanumeric name.
- **Internet Network Information Center (InterNIC) Registration Service:**
 - Manages IP addresses and domain name assignment to internet users.

June 5, 2015 | Proprietary and Confidential | - 9 -

IGATE
Speed. Agility. Imagination

Domain Name System (DNS)

It is as easy to identify computers with numbers, as it is to identify people with names. To bridge this dichotomy, the Domain Name System (DNS), a distributed database, was invented. Domain name maps or translates an actual, numeric IP address into an easy-to-remember alphanumeric name that your Web server uses.

Before DNS can do this for you, you must register any name you want to use.

Domain refers to a collection of network host computers, known by the same name. Your domain name should reflect your organization or corporation.

InterNIC (Internet Network Information Center) Registration Service, manages the task of assigning IP addresses and domain names to Internet users.

Note: InterNIC (Internet Network Information Center) lets you apply for any domain name you like, regardless of your company name. The only restriction is that the name must be available and not already reserved by someone else.

Rightmost part of a name is called its zone. The next part is the name of the company. The part to the left of the company name is the particular machine within the company. Seven domains were established originally.

1.6: Domain Name System

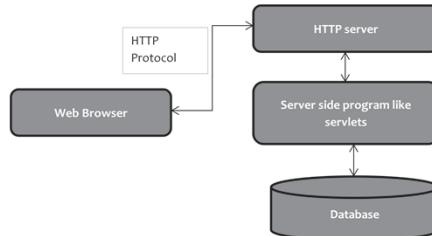
Domain Name System (Contd...)

- Few domain types are as listed below:

Domain type name	Description
.com	Commercial organization. Most companies end up as a part of this domain.
.edu	Educational establishment such as university
.gov	Branch of the U. S. government.
.int	International organization, such as NATO or the United Nations.
.mil	Mil is a branch of the U. S. military.
.net	Network organization.
.org	Non-profit organization.

1.7: HTTP Basics

- HTTP is the fundamental means of communication used by WWW.
- It defines formal syntax that allows user agents, such as browsers, to interact with web servers.
- It is one of the many protocols designed to allow clients to store and retrieve files from servers.
- HTTP requests can specify the language the browser would like to see in a page as well as information about how the data is encoded.



IGATE
Speed.Agency.Imagination

HTTP – Hyper Text Transfer Protocol

A browser works as an HTTP client because it sends requests to an HTTP server which is called Web server. The Web Server then sends responses back to the client. Diagram on above slide shows where HTTP Protocol fits in communication.

Apart from HTTP, other examples of protocols are :

- File Transfer Protocol(FTP)
- Common Internet File System (CIFS)
- Network File System (NFS)
- Simple Network Management Protocol (SNMP): Most widely used protocol for monitoring network devices such as hubs, routers, workstations, and computers. Windows NT supports SNMP.

1.7: HTTP

Client Server Interaction in HTTP

➤ **Client Server interaction in HTTP includes following four basic steps:**

- The client opens Transmission Control Protocol (TCP) connection.
- Client then sends a HTTP request such as 'GET index.HTML'
- Then, the server sends an HTTP response, including a status and a requested object.
- Finally, a TCP connection is ended.

June 5, 2015

Proprietary and Confidential

- 12 -



HTTP Version:

Earlier versions were 0.9 and 1.0 – Rarely used

Almost all clients and browsers support 1.1 today and it offers:

- Better performance
- Better support for proxies and cache

1.7: HTTP

HTTP – Stateless Nature

- **HTTP is connection less.**
- **HTTP being stateless is direct implication of HTTP being connectionless.**
 - The server and client are aware of each other only during a request. Afterwards, each forgets the other. For this reason neither the client nor the browser can retain information between different request across the web pages.
- **Advantage**
 - Simple design
 - If transaction fails, then server state is not required to be cleaned.
- **Disadvantage**
 - Additional information need to be sent in every request.
 - Information need to be interpreted



June 5, 2015

Proprietary and Confidential

- 13 -

HTTP- stateless nature

After a request is made, the client disconnects from the server and waits for a response and the server needs to re-establish the connection after it processes the request and hence HTTP is connection less.

In addition to being connectionless and stateless HTTP is also media independent which means that any type of data can be sent by HTTP as long as both the client and server know how to handle the data content. How content is handled is determined by the MIME specification.

Advantage : The stateless design simplifies the server design because there is no need to dynamically allocate storage to deal with conversations in progress. If a client dies in mid-transaction, no part of the system needs to be responsible for cleaning the present state of the server.

Disadvantage: A disadvantage of statelessness is that it may be necessary to include additional information in every request, and this extra information will need to be interpreted by the server

Examples of stateful protocol:

- TCP (Transmission Control Protocol)
- IP (Internet Protocol)
- BGP (Border Gateway Protocol)

1.7: HTTP

URL and Parts of URL

- **HTTP clients use Uniform Resource Locator (URL) to interact with resources**
- **For HTTP, URL is composed of:**
 - Scheme
 - It is implied by the fact that it is HTTP message.
 - Host
 - In HTTP 1.1, it is included in HOST header.
 - Port
 - Port is used by TCP and not HTTP. By default it is 80.
 - Path
 - Relative path of the requested resource
 - Query
 - Both Path and Query are contained in request start line.
 - Scheme and Host are case insensitive
 - Path and query are case sensitive

June 5, 2015

Proprietary and Confidential

- 14 -



Parts of URL:

Consider the given URL as

http://localhost:8080/Example/test.html?username='test'

Scheme : http

Host: localhost

Port : 8080

Path : /Example/test.html

Query String: username='test'

In HTTP 1.0, the host was not always include in the HTTP messages

**1.7: HTTP
URL and Parts of URL**

- Consider following example:
`http://www.example.com:80/path/hello.jsp?k=10&h=40`

- See below the detailed description of the given URL:

URL Parts name	Example	Comparison
Scheme	http	Equals to HTTP
Host Name	www.example.com	Equals to WWW.EXAMPLE.COM
Path	/hello.jsp	Not equals to /HELLO.jsp
Query	k=10&h=40	Not equals to K=10&H=40

URL parts for example given on slide:

Scheme : http

Host : www.example.com

Port : 80

Path : /path/hello.jsp

Query : k=10&h=40

1.7: HTTP

HTTP Request Methods

➤ **GET**

- It is used to retrieve a resource from server

➤ **POST**

- POST is used to pass information to the server.
- POST allows clients to send messages to forums or update databases.

June 5, 2015

Proprietary and Confidential

- 16 -

IGATE
Speed. Agility. Imagination.

HTTP Request Methods

GET: The GET method is used to retrieve information from the given server using a given URI. Requests using GET should only retrieve data and should have no other effect on the data.

POST: A POST request is used to send data to the server, for example, customer information, file upload, etc. using HTML forms

1.8: Servers Servers

- A server is a system (software and suitable computer hardware) that responds to requests across a computer network to provide, or help to provide, a network service.
- Servers operate within a client-server architecture, servers are computer programs running to serve the requests of other programs, the clients.
- Thus, the server performs some task on behalf of clients. The clients typically connect to the server through the network but may run on the same computer



June 5, 2015

Proprietary and Confidential

- 17 -

IGATE
Speed.Agency.Imagination

Above diagram shows clients communicating with a server via internet.

Servers often provide essential services across a network, either to private users inside a large organization or to public users via the Internet.

1.8: Servers

Servers - Example

- Application Server
- Database Server
- File Server
- Print Server
- Web Server

June 5, 2015

Proprietary and Confidential

- 18 -



Examples of Servers

Application server - A server dedicated to running certain software applications

Database server – It provides database services to other computer programs or computers

File server – It provides remote access to files

Print server – It provides printer services

Web server – A server that HTTP clients connect to in order to send commands and receive responses along with data contents

1.8: Servers
Web Servers

- Web server can refer to either the hardware (the computer) or the software (the computer application) that helps to deliver web content that can be accessed through the Internet.
- The primary function of a web server is to deliver web pages on the request of clients using the Hypertext Transfer Protocol (HTTP).

Delivery of web pages over the web means HTML documents and any additional content that may be included by a document, such as images, style sheets and scripts

Web servers are not always used for serving the World Wide Web. They can also be found embedded in devices such as printers, routers, webcams and serving only a local network. The web server may then be used as a part of a system for monitoring and/or administering the device in question. This usually means that no additional software has to be installed on the client computer, since only a web browser is required (which now is included with most operating systems).

1.8: Servers

Web Servers - Features

- **Virtual Hosting**
 - To serve many websites using one IP address.
- **Large File Support**
 - To be able to serve files whose size is greater than 2GB or 32 bit OS.
- **Bandwidth throttling**
 - To limit the speed of responses in order to not saturate the network and to be able to serve more clients.
- **Server-side scripting**
 - To generate dynamic web pages, still keeping web server and website implementations separate from each other.

June 5, 2015

Proprietary and Confidential

- 20 -



A website is a set of related web pages served from a single web domain and is hosted on at least one web server, accessible via a network such as the Internet or a private local area network through an Internet address known as a Uniform Resource Locator(URL).

All publicly accessible websites collectively constitute the World Wide Web.

1.8:Servers

Web Servers – Path Translation

- Web servers are able to map the path component of URL into:
 - A local file system resource (for static requests)
 - An internal or external program name (for dynamic requests)
- Consider URL :
`http://www.exam.com/path/file.html` is requested by client.
- The client's user agent will translate it into a connection to www.example.com with the following HTTP 1.1 request:
`GET /path/file.html HTTP/1.1`
Host: www.example.com
- The web server on www.example.com will append the given path to the path of its root directory.
 - On an Apache server, this is commonly /home/www
 - On Unix machines, usually /var/www
 - The result is the local file system resource: /home/www/path/file.html
- The Web Server then reads the file, if it exists and sends a response to the client's web browser.

June 5, 2015

Proprietary and Confidential

- 21 -



Web Server Path Translation :

Once a request is made by a client using the below URL, the user agent will translate URL into a connection to a web server.

`http://www.exam.com/path/file.html`

The Web Server then reads the file, if it exists and sends a response to the client's web browser. The response will describe the content of the file and contain the file itself or an error message will return saying that the file does not exist or is unavailable.

1.8: Servers

Web Servers – Examples

➤ Examples of web servers

- Apache by Apache
- IIS by Microsoft
- nginx by NGINX Inc
- GWS by Google

Developer	December 2014	Percent	January 2015	Percent
Apache	358,159,405	39.11%	348,460,753	39.74%
Microsoft	273,967,294	29.81%	241,276,347	27.52%
nginx	132,467,763	14.47%	128,083,920	14.61%
Google	20,011,260	2.19%	20,209,649	2.30%

The above table by netcraft (Netcraft is an internet services company providing research data and analysis on many aspects of the internet.) shows number of websites hosted on each of the web servers listed in above slide

1.9: Web Browser

Web Browser

- A web browser (commonly referred to as a browser) is a software application for retrieving, presenting and traversing information resources on the World Wide Web.
- An information resource is identified by a Uniform Resource Identifier (URI) and may be a web page, image, video or other piece of content
- A web browser can also be defined as an application software or program designed to enable users to access, retrieve and view documents and other resources on the Internet.

1.9: Web Browser

Web Browser

- Available web browsers range in features from minimal, text-based user interfaces with bare-bones support for HTML to rich user interfaces supporting a wide variety of file formats and protocols.
- Browsers which include additional components to support e-mail, Usenet news, and Internet Relay Chat (IRC), are sometimes referred to as "Internet suites" rather than merely "web browsers".
- Most browsers can be extended via plug-ins, downloadable components that provide additional features.

1.9: Web Browser

Web Browser – Examples

- Google Chrome
- Mozilla Firefox
- Microsoft Internet Explorer
- Opera by Opera Software
- Apple Safari

June 5, 2015

Proprietary and Confidential

- 25 -

IGATE
Speed.Agency.Imagination

1.10: Working of www

Working of WWW

- **WWW (World Wide Web) refers to all of the publicly accessible web sites in the world, in addition to other information sources that web browsers can access.**
 - These other sources include FTP sites, USENET newsgroups, and a few surviving Gopher sites.
- **Typically Internet follows client/server model where:**
 - Web-browsers acts as client software on the remote machine.
 - The server software is hosted on the webserver which acts as host.
- **Whenever you view a web page on the internet, you are requesting that page from a web server. When you type a URL into your browser (for example, "http://www.igate.com/igate-profile.aspx"), your browser requests the page from the web server and the web server sends the page back:**

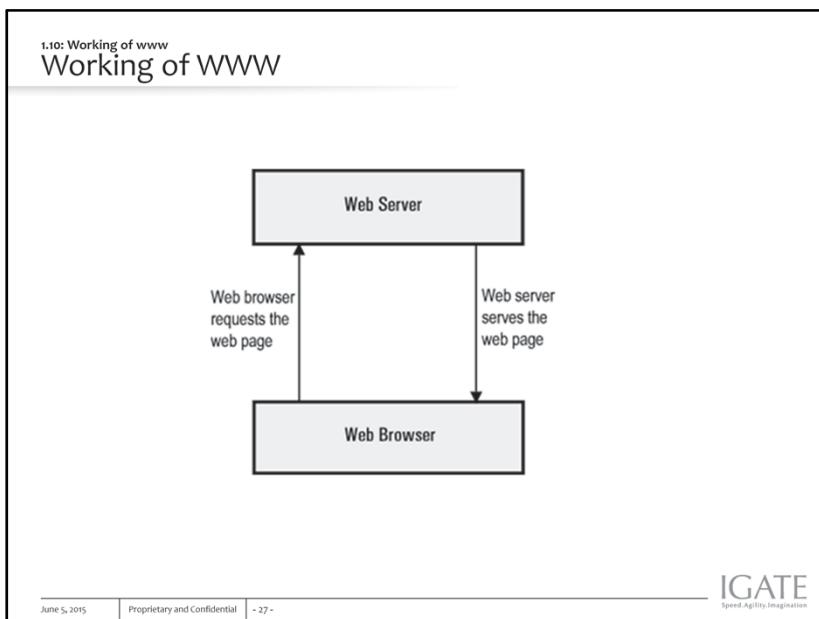


Diagram shown on above slide is simplistic version, Here is the detailed description:

Your web browser first needs to know which IP address the website "www.igate.com" resolves to. If it doesn't already have this information stored in its cache, it requests the information from one or more DNS servers (via the internet). The DNS server tells the browser which IP address the website is located at. Note that the IP address was assigned when the website was first created on the web server.

Now that the web browser knows which IP address the website is located at, it can request the full URL from the web server.

The web server responds by sending back the requested page. If the page doesn't exist (or another error occurs), it will send back the appropriate error message.

Your web browser receives the page and renders it as required.

Multiple Websites : A web server can contain more than one website. In fact, many hosting companies host hundreds, or even thousands of websites on a single web server. Each website is usually assigned a unique IP address which distinguishes it from other websites on the same machine. This IP address is also what the DNS server uses to resolve the domain name.

It is also possible to configure multiple websites without using different IP addresses using host headers and/or different ports.

Page Not Found : If the requested page isn't found, the web server sends the appropriate error code/message back to the client. You can create user friendly error messages, then configure your web server to display that page instead of the usual error page. This can add a nice touch to your website.

Default Documents: If you've ever created a website, you may have found that if you have an "index" file (index.html for example), you don't need to specify the name of the file. For example, the following URLs both load the same page:

<http://www.example.com/html/tutorial>

<http://www.example.com/html/tutorial/index.html>

In this example, "index.html" is the *default document*. You can configure your web server so that any file name can be the default document.

For example, you could configure your web server to use "index.html" in the event no filename has been specified. You could even specify different default documents for different directories if you like.

SSL Certificates

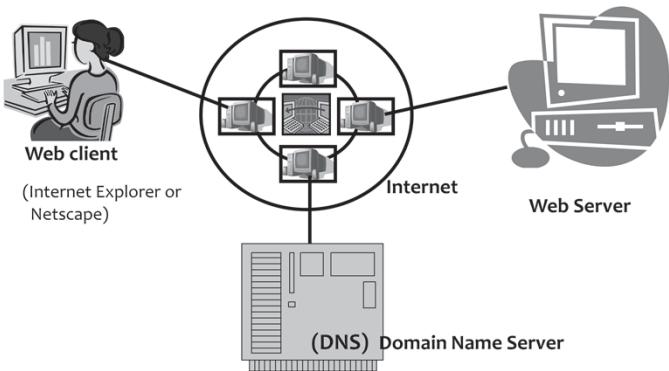
You can apply SSL certificates against a website via the web server. First you need to generate the certificate either by yourself (i.e. using a certificate generator), or by a Certificate Authority (CA). Then, once it has been generated, you apply it to your website via your web server. Applying an SSL certificate to a website is a straight forward task.

Once you've applied an SSL certificate against a website, you can navigate it using HTTPS (as opposed to HTTP). HTTPS encrypts any data that is transferred over the internet. This reduces the possibility of some malicious person being able to read your users' sensitive information.

To navigate a website using HTTPS, you simply replace the HTTP with HTTPS at the start of the URL in your browsers' location bar ("https://www.example.com")

1.10: Working of www
Working of WWW

- Open location <http://www.google.com/index.html>



Consider <http://www.google.com/index.html> is typed in web client by the user.

1.10: Working of www
Working of WWW

➤ Open location <http://www.google.com/index.html>

The diagram shows a 'Web client' (a person sitting at a computer) sending a request to a 'Domain Name Server (DNS)'. The DNS is represented as a server unit with multiple ports. A dotted line labeled 'What is IP address of google.com' connects the client to the DNS. The DNS then returns the IP address '10.10.10.9' to the client. The client then connects to a 'Web Server' (a computer with a monitor and keyboard) located at the IP address '10.10.10.9'. The 'Internet' is shown as a network of interconnected nodes between the client and the web server.

Web client

Connecting host google.com

What is IP address of google.com

(DNS) Domain Name Server

10.10.10.9

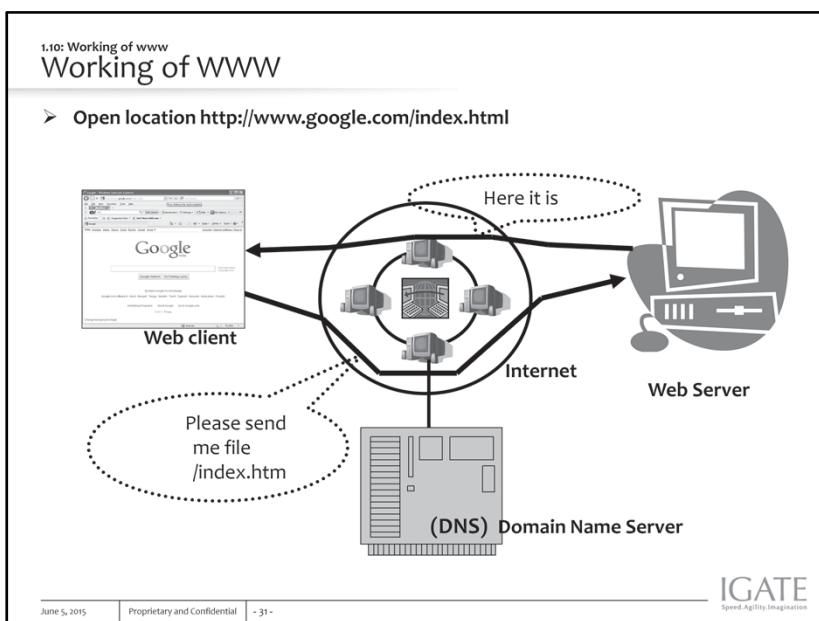
Internet

Web Server

IGATE
Speed.Agency.Imagination

June 5, 2015 | Proprietary and Confidential | - 30 -

After the request is made, Domain Name server will be connected via Internet to find the ipaddress (for the domain name) of the web server which need to be connected for processing the request.



Once Ipaddress of the webserver is found, request from the webclient will be sent to the web server. In the webserver , request will be processed and response will be sent back to the web client.

1.10: HTML

HTML and WWW

➤ **What is HTML?**

- HTML stands for Hyper Text Markup Language
- HTML is a markup language used to design web pages.
- A markup language is a set of markup tags
- The tags describe document content
- HTML documents contain HTML tags and plain text
- HTML documents are also called web pages

➤ **Web Pages are of two types:**

- Static Web Page
- Dynamic Web Page

June 5, 2015

Proprietary and Confidential

- 32 -



HTML tags are keywords (tag names) surrounded by angle brackets like <html>

1.10: HTML

Static Web Page

- A static web page is a web page that is delivered to the user exactly as stored, in contrast to dynamic web pages which are generated by a web application.
- A static web page displays the same information for all users, from all contexts, subject to modern capabilities of a web server to negotiate content-type or language of the document where such versions are available and the server is configured to do so.
- Static web pages are often HTML documents stored as files in the file system and made available by the web server over HTTP.
- **Disadvantages:**
 - Any personalization or interactivity has to run client-side, which is restricting.
 - Maintaining large numbers of static pages as files can be impractical without automated tools.

1.10: HTML

Dynamic Web Page

- A dynamic web page is a web page with web content that varies based on parameters provided by a user or a computer program.
- For dynamic behavior, client side scripting and server side scripting are used.
- Client-side scripting is changing interface behaviors within a specific web page in response to mouse or keyboard actions, or at specified timing events.
- Server side scripting involves program running on a web server and is used to change the web content on various web pages, or to adjust the sequence or reload of the web pages. Server responses may be determined by such conditions as data in a posted HTML form, parameters in the URL, the type of browser being used, the passage of time, or a database or server state.

June 5, 2015

Proprietary and Confidential

- 34 -



In Client side scripting, the dynamic behavior occurs within the presentation. The Client-side content is generated on the user's local computer system.

Such web pages use presentation technology called rich interfaced pages. Client-side scripting languages like JavaScript or ActionScript, used for Dynamic HTML (DHTML) and Flash technologies respectively, are frequently used to orchestrate media types (sound, animations, changing text, etc.) of the presentation.

Web pages that need server side scripting are often created with the help of server-side languages such as ASP, ColdFusion, Perl, PHP, Ruby, WebDNA and other languages. These server-side languages often use the Common Gateway Interface (CGI) to produce dynamic web pages. Three notable exceptions are ASP.NET, JSP, and LSP, which reuse CGI concepts in their APIs but actually dispatch all web requests into a shared virtual machine.

Dynamic web pages are often cached when there are few or no changes expected and the page is anticipated to receive considerable amount of web traffic that would create slow load times for the server if it had to generate the pages on the fly for each request.

Lesson Summary

➤ **In this lesson, you have learnt about:**

- Internet: Connection of interrelated networks.
- Protocols: Developed to maintain communication standard across:
 - Different computers and operating systems (Platforms).
 - TCIP/IP concepts
 - HTTP concepts
- WWW:
 - Webservers
 - Web Browsers
 - Working of WWW
- HTML
 - Static Web Page
 - Dynamic Web Page

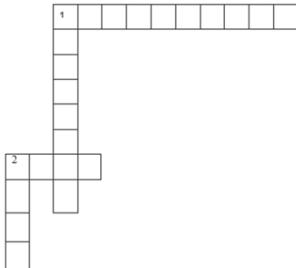


Review Question

- **Question 1:** URL is a networked extension of the standard filename concept.
 - True/False
- **Question 2:** HTTP is a Stateless Protocol
 - True/False
- **Question 3:** Which of the following Webservers host maximum number of websites?
 - Apache
 - IIS
 - Nginx
 - GWS



Review Question : Crossword



June 5, 2015

Proprietary and Confidential

- 37 -

IGATE
Speed.Agency.Imagination

Clues:

ACROSS

- 1 Client for accessing webpage
- 2 Stateless & connectionless protocol

DOWN

- 1 Set of related web pages
- 2 Markup Language

Web Basics - HTML

Lesson 2. HTML Basics

June 5, 2015

Proprietary and Confidential

- 1 -

IGATE
Speed. Agility. Inspiration.

Lesson Objectives

➤ After completing this lesson, you will be able to:

- Understand the structure of an HTML page.
- Learn to apply physical/logical character effects.
- Learn to manage document spacing.
- New Semantic Elements in HTML 5



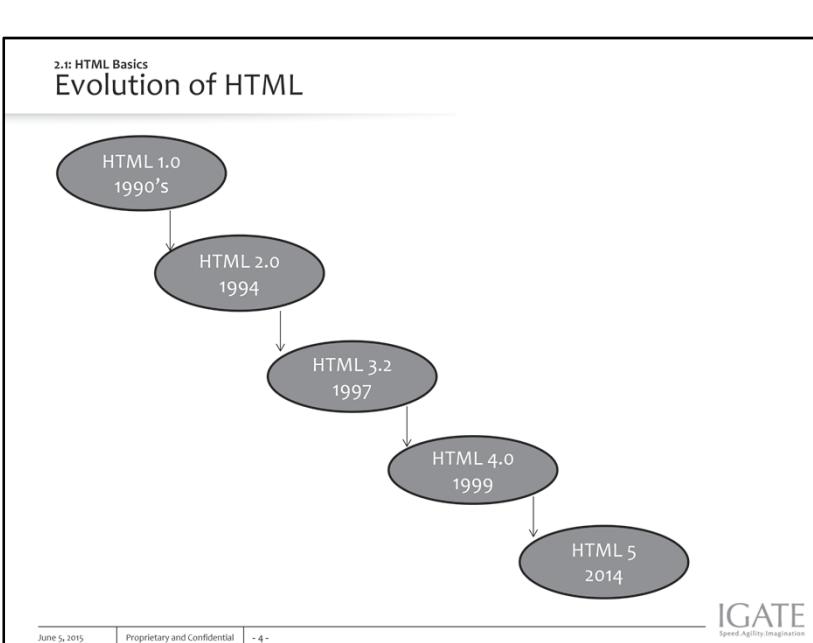
2.1: HTML Basics

What is HTML?

- **HTML is a language for describing web pages.**
 - It stands for Hyper Text Markup Language
 - HTML is a markup language and not a programming language
 - HTML uses markup tags to describe web pages.

HTML is a markup language and these documents describe web pages. Any HTML document contains HTML tags and plain text. HTML documents are also called web pages.

A web browser (like Internet Explorer or Firefox) reads HTML documents and display them as web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page:



HTML is a markup language and these documents describe web pages. Any HTML document contains HTML tags and plain text. HTML documents are also called web pages.

A web browser (like Internet Explorer or Firefox) reads HTML documents and display them as web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page:

2.1 HTML Basics

HTML5 – The Good News !!!

- Good support on modern mobile devices (iOS, Android)
- Simpler, more intuitive syntax
- Video and Audio can be included without requiring a plug-in
- Incremental improvements to previous HTML challenges
- Much needed next step in HTML evolution
- Creative enhancements: Rounded corners, gradients, text layout
- Promising support of Mobile JS Frameworks (Sencha, jQTouch)

2.1 HTML Basics

HTML5 - Browser Support

Android 2.2 Chrome 6 Firefox 4.0
Opera 10.6 IE 9 Safari 5
Safari Mobile

June 5, 2015 | Proprietary and Confidential | - 6 -

IGATE
Speed. Agility. Imagination.

Browser Support:

HTML5 is still a work in progress. However, most modern browsers have some HTML5 support. The latest versions of Safari, Chrome, Firefox, and Opera support

some HTML5 features. Internet Explorer 9 will support some HTML5 features.

Some Statistics:

Chrome 10.0 is the big winner coming in. They have already implemented most of

the functionality to be a compatible HTML5 browser. In addition, many of the features

that are not added are partially added. So Google Chrome is definitely way out in front in the race for HTML compatible browsers.

Firefox 4.0 is next in line. They are still missing a lot of key elements but got bonus

points for the audio and video implementation as well as their parsing rules.

Internet Explorer 8 is pretty far behind the curve. Pretty dismal showing for what

used to be the top internet browser in the world. Internet Explorer has been playing

catch up with rendering design since the implementation of CSS and their poor showing here tells us it does not seem like much will change in the future.

So those are the statistics. At the moment the only HTML5 browser that is going to

get you very far is Google Chrome and until browsers catch up with the newer language it is probably a good idea to use it sparingly in your designs until it actually

is a true and tested standard.

2.2: HTML Elements

HTML Elements

- Most Web documents are created using HTML.
- Documents are saved with extension .html or .htm.
- Tags are strings in the language surrounded by a less-than (<) and a greater-than (>) sign.
 - Opening tag: <html> Ending tag: </html>
- Can have Attributes
 - Attributes are Name-Value pairs added to HTML start tags.

June 5, 2015

Proprietary and Confidential

-7-

IGATE
Speed. Agility. Imagination

HTML documents are text files made up of HTML elements. These files are saved with an extension of .htm or .html A file with this extension indicates to the browser that the file is an HTML document.

You can check for the document structure and look for the presence of HTML element or tag. HTML elements are defined using HTML tags. Following are features of HTML tags:

Used to mark-up HTML elements.

Surrounded by the two characters “<“ and “>”.

Normally come in pairs.

HTML element starts with a start tag . Then, the content of the HTML element is placed. This ends with an end tag

Not case sensitive.

For HTML, the usual filename extension is .html
(.htm for PC-based servers).

Q: Do we need to write the end tag for all the elements?

A: In HTML, there are some tags which do not have a closing tag. For example,
 and <hr> tags. So, it is not necessary to write the closing tags for such tags. But it is a good idea to write an empty tag by providing a “/” in the end tag.

Rules applicable for HTML5 Tags

➤ Rules for HTML5 Tags are:

- The document must include with an HTML5 DOCTYPE.
- Tags and attributes are case-insensitive.
- Attributes do not need to be quoted.
- End tags are not required for every element.
- Some attributes may be empty such as checked and disabled .
 - For example, <input type=checkbox checked>
- Only void elements such as br, img and link may be "self-closed" with />.

We stated that HTML tags are not case sensitive, but let's always use lowercase tags. To prepare yourself for the next generations of HTML, start using lowercase tags.

Tag Attributes

Tags can have attributes. Attributes provide additional information about HTML elements on your page.

For an Example:

This tag defines the <body> element of your HTML page. With an added bgcolor attribute, you can tell the browser that the background color of your page should be red, like this:

```
<body bgcolor="red">
```

Attributes always come in name/value pairs like this: name="value". Attributes are always added to the start tag of an HTML element.

Quote Styles

Always enclose values in quotes. Double quotes are more common, but single quotes are also allowed. In some rare situations, like when the attribute value itself contains quotes, it is necessary to use single quotes:

```
name='John "ShotGun" Nelson'
```

2.2: HTML Elements

HTML Elements (Code)

➤ **Code Snippet**

An HTML document appears as follows:

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Title of page</title>
  </head>
  <body>
    This is my first homepage. <b>This text is bold</b>
  </body>
</html>
```

June 5, 2015 | Proprietary and Confidential | - 9 -

IGATE
Speed. Agility. Imagination

Save the file as "mypage.html".

Click "Browse" (or "Choose File") and locate the HTML file you just created - "mypage.html". Select it and click "Open". Now you should see an address in the dialog box, for example "C:\MyDocuments\mypage.htm". Click OK, and the browser will display the page.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
```

It tells type, version, language of particular document. The first tag in your HTML document is **<html>**. This tag tells your browser that this is the start of an HTML document. The last tag in your document is **</html>**. This tag tells your browser that this is the end of the HTML document.

Text between the **<head>** tag and the **</head>** tag is the header information. This information is not displayed in the browser window. Text between the **<title>** tags is the title of your document. Title is displayed in your browser's title bar. Text between the **<body>** tags is the text that is displayed in your browser. Text between the **** and **** tags is displayed in a bold font.

2.3. Document Sections

HTML Document Sections

- **HTML Head Section:**
 - <head>...</head>
 - Page Title, Base URL, Meta Information
- **HTML Body Section:**
 - <body>...</body>
 - Text, Images, Tables Colors, etc.

June 5, 2015 | Proprietary and Confidential | - 10 -

IGATE
Speed. Agility. Imagination

Head element can contain information about the document. The browser does not display this information to the user. Following tags can be in the head section: `<base>`, `<link>`, `<meta>`, `<script>`, `<style>`, and `<title>`.

Body element defines the document's body. It contains all the contents of the document (like text, images, colors, graphics, etc.).

2.3: Document Sections
Head Section: Page Title (Code)

➤ **Code Snippet**

Document Title is displayed using <title>.....</title> tag.

```
<!DOCTYPE HTML>
<html>
    <head>
        <title>
            My First Page
        </title>
    </head>
</html>
```

June 5, 2015 | Proprietary and Confidential | - 11 -

IGATE
Speed. Agility. Imagination

<TITLE> element does not have any visible effect within a browser's client area; however, the enclosed title appears in the title bar of the browser window.

2.1: Html Basics
Demo

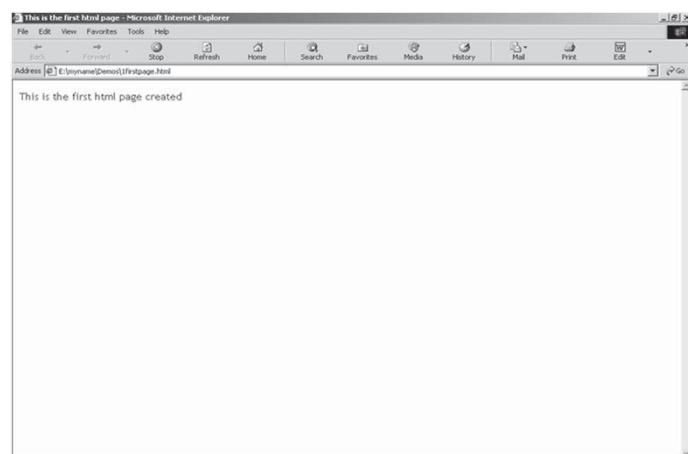
➤ Firstpage.html



June 5, 2015 Proprietary and Confidential - 12 -

IGATE Speed-Agility-Imagination

```
<html>
<head><title>This is the first html page</title>
<body>This is the first html page created</body>
</head>
</html>
```



2.3: Document Sections

Head Section: Document Base URL

➤ **Code Snippet**

```
<!DOCTYPE HTML>
<html>
  <head>
    <title> Document Base URL Manipulation </title>
    <base href="URL/">
  </head>
</html>
```

More Examples:

```
<base href="http://www.state.edu/images/">
<base href="ftp://ftp.state.edu/images/">
```

June 5, 2015 | Proprietary and Confidential | - 13 -

IGATE
Speed. Agility. Imagination.

Base element specifies a base URL for all the links in a page.
Note: The `<base>` tag must go inside the head element.

You can set the `<BASE>` element only once in a document, in the header. `<BASE>` element does not work outside the header.

Assume that the absolute address for an image is:
``

Now, we insert the `<base>` tag, which specifies a base URL for all of the links on a page, in its head section:

```
<head>
  <base href="http://www.state.edu/images/" />
</head>
```

When we insert images on the page, in the example above, we just specify the relative address, `` and the browser looks for that file using the full URL, "http://www.state.edu/images/smile.gif".

2.3: Document Sections

Head Section: Meta Information

- The <meta> tag provides metadata about the HTML document such as descriptions and keywords for search engine.
- Metadata will not be displayed on the page, but will be machine parsable.
- Examples:

```
<meta .....>
<meta name="keywords" content="HTML, CSS" />
<meta name="author" content="Username">
<meta http-equiv=refresh content=60 />
<meta http-equiv=refresh content="20;url=c:/html/htm3.htm" />
<meta charset="UTF-8">
```

June 5, 2015 | Proprietary and Confidential | - 14 -

IGATE
Speed. Agility. Imagination

META: This element is used to specify additional meta information and appears in the <HEAD> tag. You can specify multiple <META> tags for the document. It provides meta-information about your page, such as descriptions and keywords for search engines and refresh rates. Following are some uses of META:

Define keywords for search engines:

```
<meta name="keywords" content="HTML, DHTML, CSS,
XML, XHTML, JavaScript, VBScript" />
```

Provide a description of your web page:

```
<meta name="description" content="Free Web tutorials
on HTML, CSS, XML, and XHTML" />
```

Define the last revision of your page:

```
<meta name="revised" content="Hege Refsnes,
6/10/99" />
```

Refresh page every 5 seconds:

```
<meta http-equiv="refresh" content="5" />
```

2.3: Document Sections
Demo

➤ **Meta.html**



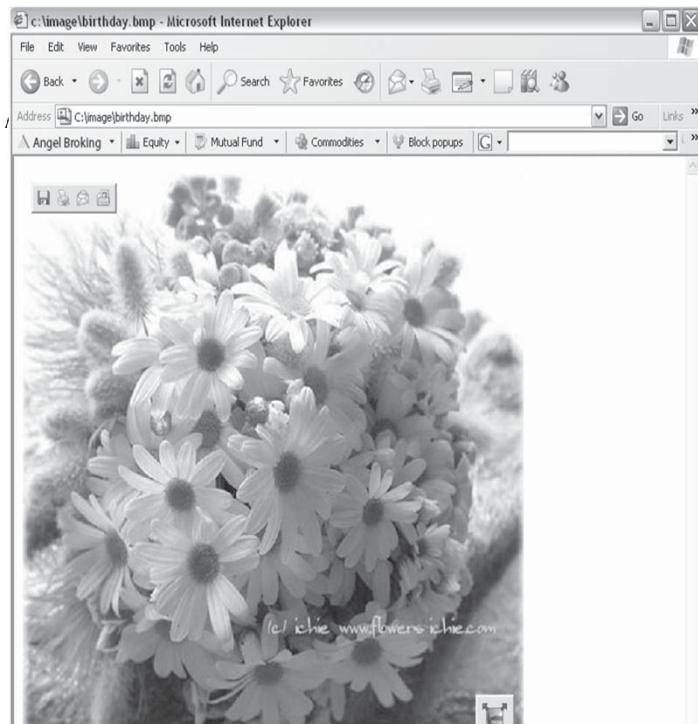
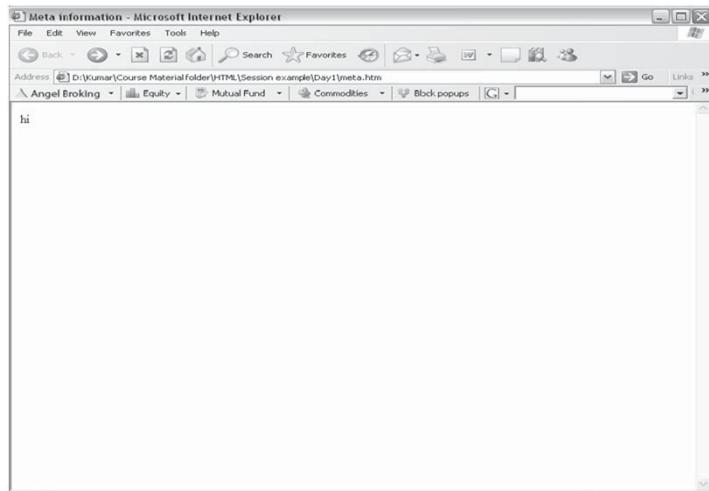
June 5, 2015 | Proprietary and Confidential | - 15 -

IGATE
Speed. Agility. Imagination

```
<!DOCTYPE HTML>
<html>
<head>
<!-- in meta tag in name attribute user can specify the value like keywords, description -->
<meta name="generator" content="microsoft visual studio 6.0">
<title>meta information</title>
<!-- this meta information help you to refresh the page after 2 sec and load image -->
<meta http-equiv=refresh content="2;url=c:\image\birthday.bmp"/>
</head>
<body>
Hi <p>&nbsp;</p>
</body>
</html>
```



Before Automatic Refresh



2.3: Document Sections

HTML Body Section

➤ <body> Element:

- Represents information content.
- Each document can have at most one <body> element.
- Body element is placed between </head> and </html> elements.
- Attributes supported in <body> element are:
 - Event Handler attributes like ononline, onoffline, onunload, onpagehide, onpageshow, etc..
 - Global attributes like id, style, class, hidden, lang, etc..

<BODY> tag defines the HTML element containing the body of the HTML document.

2.3: Document Sections

Document (Body) Contents

- **Body Text**
 - HTML truncates spaces in your text.
 - Use
 to insert new lines.
 - Use <p> tag to create paragraphs.
- **Other Elements of Body Section:**
 - <table> tags are used to create tables.
 - tags are used to insert images.

June 5, 2015 | Proprietary and Confidential | - 18 -

IGATE
Speed. Agility. Imagination

When you write HTML text, you are never sure how text is displayed in another browser. Some computers have large displays whereas some have small ones. Text is reformatted every time the user resizes the window.

Avoid reformatting text in your editor by adding empty lines and spaces. HTML truncates spaces. Any number of spaces as well as a new line counts as one space. Use
 to insert blank lines. You might have noticed that
 tags can be written without the closing tag </br>.

HTML automatically adds an extra blank line before and after some elements, like before and after paragraphs and headings.

Use a <P> tag to start a new paragraph, to which you can assign new attributes. Most browsers also place an extra space after a <P>. A
 causes the browser to maintain the current paragraph attributes but to start placing text on a new line.

</TABLE> is used to close the table tag.

To insert an image, the tag is used. Image should be saved with extension .gif. The tag is empty, which means that it contains attributes only and it has no closing tag.

2.3: Document Sections

Comments in HTML Document

- Increase code readability.
- Ignored by the browser.
- Example of HTML comment:
– <!-- This is a Sample HTML Comment -->

June 5, 2015 | Proprietary and Confidential | - 19 -

IGATE
Speed. Agility. Imagination

Comments are used for better code readability. Browser will ignore a comment. You can use comments to explain your code, or to store program-specific information, which can help you when you edit the source code at a later date.

Comments are not visible to the user, but they are still available to the program. A good practice is to comment text inside the script and style elements to prevent older browsers, that do not support scripting or styles, from showing it as plain text.

To include comments in the document, use the comment tag. Comment is a special tag starting with a “<“ sign, followed by “!” and two hyphens. Then, type the commented text. It ends with two hyphens and a “>” sign.

2.3: Document Sections

Demo

➤ Body.html



June 5, 2015 Proprietary and Confidential - 20 -

IGATE
Speed. Agility. Imagination

```
<html>
<head>
<title>body tag</title>
</head>
<body bgcolor="pink" text="red" alink="green" link="yellow">
<a href="body.html">background</a> color of the page is pink.
and text color is red
</body>
</html>
```

2.3: Home Page

Home Page Concepts

- Most popular home pages reflects the personality of the sponsoring organization or corporation.
- Keep the initial home page short and to the point.
- First element visitor sees is a collection of navigation buttons to navigate to other pages.
- When you publish a URL in print or any other marketing material, it points to the location of your home page.

Physical Character Effects

Tags applies physical character effects by formatting the characters are listed below:

Tag Name	Description	Example
..	To specify bold text	Language
<i>..</i>	Displays content in italic to define a part of text in an alternate voice or mood	Language
<u>..</u>	Tags make text between them underlined	Language
<s>..</s>	Tags make text between them appear struck out	Language
_{..}	To specify subscript text	H ₂ O
^{..}	To define superscript text	2 nd May

June 5, 2015 | Proprietary and Confidential | - 22 -

IGATE
Speed. Agility. Imagination

BOLD

... : Tags make text between them bold e.g. igate is displayed as “igate”.

ITALIC

<i>... </i>: Tags make text between them italicized e.g. <i>igate </i> is displayed as “igate”.

UNDERLINE <u>... </u>: Tags make text between them underlined e.g. igate is displayed as “igate”.

STRIKETHROUGH

<strike> or <s> <strike> or <s>: Tags make text between them appear struck out e.g. <s>igate </s> is displayed as “igate” (with a strike across it.)

Subscripts and Superscripts

_{text} subscripts text e.g. Chemical formula of water is H₂O is displayed as “ H₂O”

Formula for a parabola is y=x² is displayed as “ Y=X²”.

2.4: Physical Character Effects

Demo

- [PhysicalCharacterEffect.html](#)



June 5, 2015

Proprietary and Confidential

- 23 -

IGATE
Speed. Agility. Imagination

2.5: Logical Character Effects

Logical Character Effects

- **Heading Styles:**
- `<hn>.....</hn>`
- **Value of n can range from 1 to 6**

```
<h1>This is level 1 heading</h1>
```

- **Syntax**

June 5, 2015

Proprietary and Confidential

- 24 -



Heading Style

`<hn>.....</hn>`: Tag identifies headings and subheadings in a document. Here, n is the size of the heading ranging from 1 to 6 where 1 is largest and 6 is smallest.

For example:

```
<H1>This is an example of level 1 heading<H1>
<H6>This is an example of level 6 heading<H6>
```

```
<html>
<body>This is the first html page created
<h1>This is level 1 heading</h1>
<h2>This is level 2 heading</h2>
<h3>This is level 3 heading</h3>
</body>
</html>
```

2.5: Logical Character Effects

Logical Character Effects (Contd...)

<code><code></code>	Displays any code part in the web page.
<code><var></code>	Displays any variable on the web page.
<code><kbd></code>	Displays computer commands and arguments.
<code></code>	Highlights specific areas of text enclosed within it.
<code></code>	Used for defining important text.
<code><dfn></code>	Designed specifically for words and phrases defined in text.
<code><address></code>	Specifies information about the creator e.g. Address, mailing address.

Code : `<code>..</code>`: Tag displays any code part on the web page.

e.g. `<code>perform 2000-modify-para</code>` displays
“perform 2000-modify-para”

Var : `<var>....</var>`: Tag displays any variable on the web page

e.g. `<var>count</var>` displays count.

Keyboard

`<KBD>....</KBD>`: Tag to display computer commands and arguments, especially those to be entered by the user. The text tagged by `<KBD>` is usually shown in a fixed-width font. Indicating text as keyboard input.

E.g. `<KBD>copy *.exe c:\dir1</KBD>` displays

copy *.exe c:\dir1.

Emphasis

`.....`: Tag provides generic emphasis or highlights specific areas of text without requiring the browser to set a specific physical style for them e.g. `This is igate` displays “*This is igate*”.

Strong

`...`: Tag for strong emphasis e.g. `This is an L2 question` displays “*This is an L2 questions*”.

Emphasis and strong emphasis tags highlight specific areas of text without the browser having to use a specific physical style for them.

Definition

`<dfn>.....</dfn>`: Tag is designed specifically for words and phrases that are defined in the text. Physical method of emphasizing the words is left to the browser. This tag changes the style of the text contained between the `<DFN>` and `</DFN>` tags. Style used is dependent on the browser, but is usually either bold or italic e.g.

`<dfn>Definition</dfn>`.

Address

`<address>...</address>`: Tag is used to specify information about the creator or maintainer of a Web page. This information can include the name, e-mail address, phone number, mailing address, or other relevant information.

2.5: Logical Character Effects

Special Characters in HTML

➤ Character Entities

- Comprise following three parts:
 - Ampersand (&),
 - Entity name or a #
 - Character code
 - Semicolon (;)
- Included in HTML page using:
 - Character code/Entity number: Include any character using its ISO Latin 1 character code.
 - To display ">" symbol, character code is 62 i.e. >

June 5, 2015

Proprietary and Confidential

- 27 -

 IGATE
Speed. Agility. Imagination.

Some characters like the “<“ character, have a special meaning in HTML, and therefore cannot be used in text. To display a less than sign (<) in HTML, we need use a character entity.

Character Entities

Some characters have a special meaning in HTML, like the less than sign (<) that defines the start of an HTML tag. If we want the browser to actually display these characters we must insert character entities in the HTML source.

A character entity has three parts: an ampersand (&), an entity name or a # and an entity number, and finally a semicolon (;). To display a greater than sign in an HTML document we write: > or >

The advantage of using a name instead of a number is that a name is easier to remember. The disadvantage is that not all browsers support the newest entity names, while the support for entity numbers is very good.

Note: Entities are case sensitive.

Non-Breaking Space

The most common character entity in HTML is the non-breaking space. Normally HTML will truncate spaces in your text. If you write 10 spaces, HTML will remove 9 of them. To add spaces, use the character entity.

2.6: Managing Document Spacing

➤ Horizontal Spacing

- Use `<hr>` tag for including horizontal rule in an HTML document

➤ Vertical Spacing

- `<p>`: Paragraph Break
- `
`: Line Break

`<p> ... </p>`: Tag to start and end the new paragraph.

`
... </br>` Tag maintains current paragraph but enters text in the new line.

2.6: Managing Document Spacing

Demo

- LCharacterEffects.htm
- Spacing.html



June 5, 2015

Proprietary and Confidential

- 29 -

IGATE
Speed. Agility. Imagination

```
<html>
<head><title>Horizontal and Vertical spacing</title></head>
<body>
<p>
This paragraph contains a lot of lines
in the source code,
but the browser ignores it.
</p>
<hr size="2" width="50%" color="blue">
<p>
Notice the horizontal rule occupying 50 % of the window width.
</p>
This paragraph contains <br> line breaks in the
source code <br> so this
is the third line displayed within the paragraph.
</body>
</html>
```

2.6: Managing Document Spacing

Preformatted Text (Code)

- <PRE>...</PRE> Tags display preformatted blocks of text with a fixed-space font.
- <PRE> tag displays text with white space, line breaks, and tabs.
- Examples:

```
<!DOCTYPE html>
<html>
    <head><title> Preformatted Text</title>
    <body>
        <pre>Browser would display this paragraph
        as you are viewing here.
        No need to provide line or paragraph breaks. </pre>
    </body>
</html>
```

June 5, 2015

Proprietary and Confidential

- 30 -

IGATE
Speed. Agility. Imagination

<PRE>...</PRE> Tags display preformatted blocks of text with a fixed-space font. When they appear inside <PRE> tag, white space, line breaks, and tabs are also displayed. It is mainly useful for simulating program listings.

Note: The <pre> tag takes care of carriage returns, tabs, spaces and even the bold tag.

E.g.:

```
<html>
    <head><title>PREFORMATTED TEXT EXAMPLE</title></head>
    <body>
        <h3>EMPLOYEE DETAILS PREFORMATTING</h3>
        <hr>
        <pre>
            <b>EMPLOYEE DETAILS</b>
            EMPID          EMPNAME           DEPT
            1001            JOHN              ACCOUNTS<br>
            1002            TOM               PURCHASE<br>
            1003            TAMMY             SALES<br>
        </pre>
    </body></html>
```

2.6: Managing Document Spacing

Divisions in an HTML Document (Code)

- The `<div>` tag defines a division or a section in an HTML document.
- The `<div>` tag is used to group block-elements to format them with CSS.
- Syntax:

```
<div>.....</div>
```

- Example:

```
<!DOCTYPE html>
<html>
<body>
<div style="text-align: center">Text is center aligned.</div>
<div style="text-align: left">Text is left aligned.</div>
</body>
</html>
```

June 5, 2015

Proprietary and Confidential

- 31 -

IGATE

Speed. Agility. Imagination

`<div>... </div>`: Tag creates divisions in Web pages. These can be used to set the alignment for an entire section of the page.

`<div align=center>`This text is at the center of the browser window.`</div>`:

Tag to align the text to the center of web page.

`<div align=left>`This text is at the left side of the browser window.`</div>`:

Tag to align the text to the left of the web page.

`<div align=right>`This text is at the right side of the browser window.`</div>`: Tag to align the text to the right of the web page.

Difference Between PRE and DIV Tags:

`<pre>` is used for formatting the entire text.

`<div>` is used to divide the webpage and use center, left or right attribute to align the text of the divided web page.

Difference Between DIV and SPAN Tags:

DIV is used as a paragraph break as it creates a logical division of the document. In contrast, SPAN simply applies style and alignment as specified.

DIV has ALIGN attribute in it which is not present in case of SPAN.

DIV is generally used for a block of text. SPAN is used for words or sentences.

Primary difference between them is that `` does not format by itself. The `<div>` tag includes a paragraph break, as it defines a logical division in the document.

2.6: Managing Document Spacing
** in an HTML Document**

- The HTML tag is used for grouping and applying styles to inline elements.
- The span tag is used with inline elements whilst the div tag is used with block-level content.
- Example:

```
<!DOCTYPE html>
<html>
<body>
HyperText Markup Language is the standard markup language used to create
<span style="color:blue;font-weight:bold"> static </span> web pages </body>
</html>
```

SPAN is used for words or sentences.

Primary difference between them is that does not format by itself. The <div> tag includes a paragraph break, as it defines a logical division in the document.

Block level element

- Block elements separate content into blocks
- Each block element is displayed on its own with a line break before and after it
- Example for block elements are <p>,<div>, , , etc..
 - <div> element is a block level element that can be used as a container for other HTML elements

Inline element

- Inline elements won't separate content into blocks
- Inline elements are normally displayed without line breaks.
- Inline element is just displayed in the flow of the paragraph
- Example for Inline elements are ``,`<i>`,`<u>`,``,`<a>`,`<td>`,``, etc..
 - `` element is an inline element that can be used as a container for text.

Rules

- All inline elements and text need to be nested inside another block element before they can go in the <body> element
- Block elements are not allowed inside an inline element
- Keep block elements out of your <p> element
- Put text and inline elements inside block elements before adding them to a <blockquote>
- For an example:
 - and can have only
 - We can put text, inline elements or block elements inside

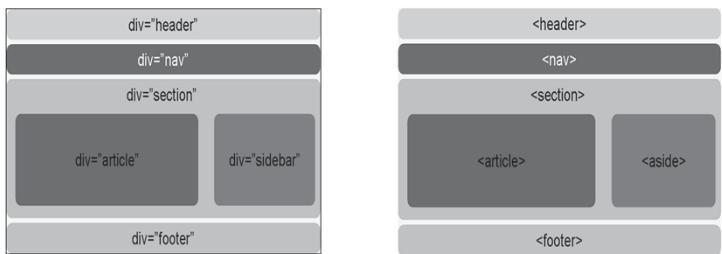
Example of Inline and Block elements

```
<!DOCTYPE html>
<html>
<body>
<p>HyperText Markup Language is the standard markup language
used to create <span style="color:blue;font-weight:bold"> static
</span> web pages. JavaScript is a scripting language used to make
web page content as <span style="color:orange;font-
weight:bold">dynamic</span>.</p>
</body>
</html>
```

2.7 New Semantic Elements in HTML 5

Laying out a page with HTML5

- Most HTML 4 pages include a variety of common structures, such as headers, footers and columns
- It's common to mark them up using div elements, giving each a descriptive id or class
- HTML 5 addresses this issue by introducing new elements for representing each of these different sections
- Elements that make it much easier to structure pages



June 5, 2015 | Proprietary and Confidential | - 37 -

IGATE
Speed. Agility. Imagination.

HTML5 Page Structure

HTML 5 introduces a whole set of new elements that make it much easier to structure pages. Most HTML 4 pages include a variety of common structures, such as headers, footers and columns and today, it is fairly common to mark them up using div elements, giving each a descriptive id or class. The use of div elements is largely because current versions of HTML 4 lack the necessary semantics for describing these parts more specifically. HTML 5 addresses this issue by introducing new elements for representing each of these different sections. The div elements can be replaced with the new elements: header, nav, section, article, aside, and footer.

Page Simplification

Small degree of freedom, but very detailed rule for parsing
Validation will matter

HTML5 is designed so that old HTML4 browsers can safely ignores new HTML5 constructs

Documents must not use deprecated features –extensive error handling by user agents

2.7 New Semantic Elements in HTML 5

HTML Tag	Description
<article>	Defines a complete, self-contained block of related elements.
<aside>	Defines some content aside from the content it is placed in (like a sidebar)
<footer>	Defines a footer for a document or section which contains the author of the document, copyright information, links to terms of use, contact information
<header>	Defines a header for a document or section. It can be used as a container for introductory content.
<nav>	Defines a set of navigation links
<section>	Defines a section in a document

2.7: New Semantic Elements in HTML5

Demo

➤ LayoutDemo.html



June 5, 2015 | Proprietary and Confidential | - 39 -

IGATE
Speed. Agility. Imagination.

```
<html>
<head><title>Horizontal and Vertical spacing</title></head>
<body>
<p>
This paragraph contains a lot of lines
in the source code,
but the browser ignores it.
</p>
<hr size="2" width="50%" color="blue">
<p>
Notice the horizontal rule occupying 50 % of the window width.
</p>
This paragraph contains <br> line breaks in the
source code <br> so this
is the third line displayed within the paragraph.
</body>
</html>
```

Lab Session

➤ Lab 1



June 5, 2015

Proprietary and Confidential

- 40 -

IGATE
Speed. Agility. Imagination

Summary

➤ In this lesson, you have learnt about

- The structure of an HTML page.
- Physical/logical character effects
- Managing document spacing.
- New Semantic elements in HTML5



Review Question

- **Question 1:** Which of the following are newly added content tags in HTML5?
 - Option1: article
 - Option2: div
 - Option3: section
 - Option4: p
- **Question 2:** HTML document is saved with an extension .xml.
 - True/False
- **Question 3:** A Var tag is used to display the _____ in the web page.



Review Question: Match the Following

1. Code

2. Var

3. Kbd

4. Emphasis

5. Strong

6. Definition

a) Tag displays user-entered computer commands and arguments

b) Tag is used for defining important text.

c) Tag displays any code part on the web page.

d) Tag displays any variable on the web page.

e) This tag designed specifically for words and phrases that are defined in the text.

f) Tag for generic emphasis means this tag highlights specific areas of text.



Web Basics – HTML5

Lesson 3. Tables

June 5, 2015

Proprietary and Confidential

- 1 -

IGATE
Speed. Agility. Imagination

Lesson Objectives

➤ After completing this module, you will be able to:

- Understand the structure of an HTML table.
- Controlling table format



Creating Tables

- Table contains data in the format of rows and columns.
- For an example, department information's are displayed in the tabular format as shown below

Deptno	Dname	Location
10	Accounting	New York
20	Research	Dallas
30	Sales	Chicago
40	Operations	Boston

Column named as “Deptno”

→ Row

- The above “Department” table contains 4 rows and 3 columns.

June 5, 2015

Proprietary and Confidential

- 3 -

IGATE
Speed.Agency.Imagination

Tables

Tables are defined with the `<table>` tag. A table is divided into rows (using `<tr>`), and each row is divided into data cells (using `<td>`). Letters `td` stand for "table data," which is the content of a data cell. A data cell can contain text, images, lists, paragraphs, forms, horizontal rules, tables, etc.

Creating Tables

➤ An HTML table can be created using `<table>` elements

- `<table>`
 - Define an HTML table
 - Other elements like `<tr>`, `<caption>`,.. can be nested inside `<table>` element

➤ An HTML table has two kinds of cells

- Header Cells
 - `<th>`
 - Defines a table header
- Standard Cells
 - `<tr>`
 - Defines a table row
 - A row can have one or more `<td>` or `<th>` elements
 - `<td>`
 - Defines a table cell data

Employee Name	Salary
Lathika	55000
Sandeep	32000
Ajay	99322

An HTML table has two kinds of cells:

- Header Cells: Contain header information (created with the `th` element).
- Standard Cells: Contain data (created with the `td` element).
 - The text in a `th` element is bold and centered.
 - The text in a `td` element is regular and left-aligned.

`<tr>` stands for Table row

`<td>` stands for table data

`<th>` stands for table header

3.1:Tables
Creating Tables

➤ **Syntax**

```
<table>
  <tr> <th>Column1 Header</th> <th>Column2 Header</th></tr>
  <tr> <td>Cell 1,1</td> <td>Cell 1,2</td> </tr>
  <tr> <td>Cell 2,1</td> <td>Cell 2,2</td> </tr>
</table>
```

Creating Tables

➤ Some more elements which can be used while creating tables are:

- <thead>
 - Group header content in an HTML table
- <tbody>
 - Group the body content in an HTML table
- <tfoot>
 - Group footer content in an HTML table
- <caption>
 - Defines a caption for the table
 - <caption> element should follow with <table> element immediately.
 - <caption> element value will be center aligned and displayed above the table

The table also has a few more tags to layout your data.

The <thead> tag is used to group the header content in an HTML table. The <thead> element should be used in conjunction with the <tbody> and <tfoot> elements.

The <tbody> element is used to group the body content in an HTML table and the <tfoot> element is used to group the footer content in an HTML table.

<tfoot> must appear before <tbody> within a table, so that a browser can render the foot before receiving all the rows of data.

Note that the <thead>, <tbody> and <tfoot> elements are seldom used, because of bad browser support. Expect this to change in future versions.

3.1: Tables Creating Tables

➤ Syntax

Table column headings:

```
<table>
    <caption>This is table caption</caption>
    <tr>
        <th>COLUMN 1</th>
        <th>COLUMN 2</th>
        <th>COLUMN 3</th>
    </tr>
</table>
```

June 5, 2015

Proprietary and Confidential

- 7 -

IGATE
Speed. Agility. Imagination

Table Caption:

The `<caption>` tag defines a table caption.

The `<caption>` tag must be inserted immediately after the `<table>` tag.

You can specify only one caption per table. Usually the caption will be centered above the table.

Table Headers:

The `<th>` tag defines a header cell in an HTML table.

Creating Tables

➤ Syntax

```
<table>
<thead>
<tr><td>.....</td></tr>
</thead>
<tfoot>
<tr><td>.....</td></tr>
</tfoot>
<tbody>
<tr><td>.....</td></tr>
</tbody>
</table>
```

3.1:Tables

Blank Data Cell (Code)

➤ **Inserting Blank Data Cell :**

- `<td></td>`
- `<td>
</td>`

June 5, 2015

Proprietary and Confidential

- 9 -

IGATE
Speed.Agency.Imagination

Blank Data Cell:

You can avoid missing a border around empty cells. Add a non-breaking space () to empty data cells, to make the borders visible.

3.1:Tables

Creating Tables

- [Tableheading.html](#)



3.1: Tables

Nested Tables (Code)

- A cell can contain another table within it.

```
<table>
<tr> <th>Zone</th> <th>State</th> </tr>
<tr> <td>South</td>
<td><table>
<tr> <th>Name</th> <th>Capital City</th> </tr>
<tr> <td>Karnataka</td> <td>Bangalore</td> </tr>
<tr> <td>Tamilnadu</td> <td>Chennai</td> </tr>
<tr> <td>Andhra Pradesh</td> <td>Hyderabad</td>
</tr>
</table></td></tr>
</table>
```

Zone	State	
	Name	Capital City
South	Karnataka	Bangalore
	Tamilnadu	Chennai
	Andhra Pradesh	Hyderabad

June 5, 2015

Proprietary and Confidential

- 11 -

IGATE
Speed.Agency.Imagination

Nested Tables:

Nesting tables within tables lets you create some complex effects from just basic HTML table tags. Nesting tables places one table inside of another. To nest a table, start the table as you normally would. When you come to the cell into which you want to place the nested table, enter the table data tag. Then, instead of typing the contents of the cell as you normally would, start the nested table. The table becomes that cell's content. After you end the nested table, be sure to end the table cell it is in with the `</td>` tag.

3.1:Tables
Demo

➤ tabnest4.htm



June 5, 2015 | Proprietary and Confidential | + 12 -

IGATE
Speed. Agility. Imagination

```
<html>
<head><title>Nested Tables</title></head>
<table width=100% height=100%>
<tr><th bgcolor="red">col1<th bgcolor="green">col2
<tr align=center><td bgcolor="aqua">data1
<td bgcolor="black"><font color="yellow">data2</font>
<table width=100% height=100%>
<tr><th bgcolor="pink">col1<th bgcolor="blue">col2
<tr align=center><td bgcolor="orange">data1<td bgcolor="purple">
<font color="yellow">data2</font>
</table>
</table>
</html>
```

3.2: Table Formatting

Table Formatting

➤ Cell Spanning

- Table cells can span across more than one column or row.
- Types of cell spanning
 - Row spanning
 - Column spanning

➤ Example of Colspan

Employee Name	Salary
HR	
Lathika	55000
Sandeep	32000
SALES	
Sonia	43000
Raju	30022
Ajay	99932

➤ Example of Rowspan

Department Name	Employee Name	Salary
HR	Lathika	55000
	Sandeep	32000
Sales	Sonia	43000
	Raju	30022
	Ajay	99932

June 5, 2015

Proprietary and Confidential

- 13 -

IGATE
Speed.Agency.Imagination

Cell Spanning:

There are two types of cell spanning. Row and column spanning.

3.2: Table Formatting

Cell Spanning (Code)

➤ Row spanning/Column spanning:

- Use rowspan and colspan attribute either in <td> or <th> element.

```
<table>
  <tr>
    <th rowspan=m>Multiple Column Header</th>
    <th colspan=n>Multiple Row Header</th>
  </tr>
</table>
```

➤ m & n are integers specifying number of rows and columns respectively.

June 5, 2015

Proprietary and Confidential

- 14 -

IGATE
Speed.Agency.Imagination

colspan="number of columns" ~

By default, the number of columns in a table is defined by the number of table data cells appearing in the table row that contains the most data. You would typically place the same number of data cells in each table row. If a table row does not contain the requisite number of table cells, then it will essentially be in 'error' and will be displayed with a missing cell.

rowspan="number of rows" ~

Rowspan attribute works just like the colspan attribute except that you may find the situation a little more difficult to visualize when working with the source code. But once again the principle is the same. By using the rowspan attribute, you can force a table cell to span the number of rows specified by the respective value.

3.2: Table Formatting

Cell Spanning Demo

➤ Table-span.html

Deptno	Name	Salary
10	Sita	34000
	Ram	24000
20	Laksman	45000
	Lava	32000
	Kusha	42000
Total Salary		1222



3.2: Table Formatting

Grouping of Columns

- <colgroup> tag specifies a group of one or more columns in a table for formatting
- The <col> tag specifies column properties for each column within a <colgroup> element.
- Use <colgroup> and <col> tags to group columns with common properties like
 - Span attribute :
 - Identifies number of columns in the current group.
 - Default value is 1
 - Provide span attribute and omit <col> tag
- Example for grouping 3 columns and applying background color as green

```
<table>
  <colgroup span="3" style="background-color:green"> </colgroup>
  <col>
  <col>
  <tr>
    table contents.....
  </tr>
</table>
```

<COLGROUP> and <COL> are another attempt to help define parts of a table more easily. Before these tags were defined, you had to keep track of how many <td> cells were in each row to know how many columns were in your table, which cells were in the same column, and how to format the cells in a given column.

Span: Defines the number of columns the <colgroup> should span

If you wish to put the first three columns in a group and format them the same way you need to write the following HTML code:

```
<table>
  <colgroup span="3"> </colgroup>
  <col>
  <col>
  <col>
  <tr>
    table contents.....
  </tr>
</table>
```

<COL> tag defines attribute values for one or more columns in a table. You can only use this element inside a colgroup. Use this element when you specify different attribute values to a column inside a colgroup. Without a col element a column will inherit all its attribute values from the column group.

3.2: Table Formatting
Demo

➤ tabcol3.htm



June 5, 2015 | Proprietary and Confidential | - 17 -

IGATE
Speed. Agility. Imagination

```
<html><head><title>Table with Colgroup</title></head>
<body>
<table width="100%" frame="void">
<tr><th colspan="3">Ecommerce Stream</th>
<th colspan="4">Mainframe Stream</th>
<colgroup span="3" width="20%" bgcolor="orange">
<colgroup span="4" width="10%" bgcolor="lightgreen">
<tr><td>Internet/HTML<td>Javascript<td>Java
<td>CICS<td>COBOL<td>JCL<td>IMS
<tr><td>JSP<td>Servlet<td>Struts
<td>ADS/O<td>VSAM<td>IDMS<td>MVS
</table>
</html>
```

Lab

➤ Lab 2



June 5, 2015

Proprietary and Confidential

- 18 -

IGATE
Speed. Agility. Imagination

Lesson Summary

- After completing this module you know:
- Structure of an HTML table
 - Control table format such as cell spanning
 - Use tables to format contents of an HTML Page.



Review - Questions

- Question 1: Cell spanning is used to joining cells together to make a larger cell.
 - True/False
- Question 2: The <col> tag defines the attribute values for one or more columns in a table.
 - True/ False
- Question 3: In which tag usage of rowspan/colspan attribute is valid?
 - <th>
 - <tr>
 - <td>
 - None of the above



Web Basics – HTML 5

Lesson 4. List

June 5, 2015

Proprietary and Confidential

- 1 -

IGATE
Speed. Agility. Imagination.

Lesson Objectives

➤ In this lesson, you will learn about:

- Numbered List
- Bulleted List
- Glossary List



HTML provides support for ordered, unordered and other types of lists.

4.1: Numbered List

- List of items can be numbered or ordered is called numbered/ordered List
- Tags can be used for creating ordered list are:
 -
 -
- tag
 - Defines a list item.
 - Should be used as a child tag of either or tag.
- tag
 - defines an ordered list.
 - An ordered list can be numerical or alphabetical.
 - Attributes in tag are
 - start
 - type

Numbered List:

HTML enables you to create a numbered, or ordered list that will automatically generate numbers in front of each item in the list.

The number placed in front of an item depends on the location of the item in the list.

At the beginning of the list, place the (for ordered list) tag.
At the end of the list, place the tag and in front of each item in the list, place the (for list item) tag.

End tag for is optional.

4.1: Numbered List

➤ Example (Code)

```
<!DOCTYPE html>
<html>
<body>
<h2>My favorite cricket teams</h2>
<ol>
<li>INDIA</li>
<li>SRILANKA</li>
</ol>
</body>
</html>
```

➤ Output

My favorite cricket teams

1. INDIA
2. SRILANKA

4.1: Numbered List

Attributes in tag

➤ Type Attribute:

- Specifies the kind of marked to use in the list
- Valid values for the Type Attribute
 - A-Uppercase letters
 - a-Lowercase letters
 - I-Uppercase Roman letters
 - i-Lowercase Roman letters
 - 1-Standard numbers, default

➤ Start Attribute

- Specifies the start value of an ordered list

You can even select the type of numbering system to be used with the type attribute. For example:

A-Uppercase letters: <OL TYPE=A>

a-Lowercase letters: <OL TYPE=a>

I-Uppercase Roman letters: <OL TYPE=I>

i-Lowercase Roman letters: <OL TYPE=i>

1-Standard numbers: <OL TYPE=1>

Type attribute enables you to alter the style of numbers used in the lists, whereas start attribute sets the starting number of the list.

To start an ordered list at a number other than 1, place the START attribute on the tag at the beginning of the list.

You can also use the VALUE attribute within tag to change the numbering sequence within a list.

4.1: Numbered List

Example

➤ Example:

```
<!DOCTYPE html>
<html>
  <body>
    <h2>My favorite cricket teams</h2>
    <ol start="2" type="a">
      <li>INDIA
      <li>SRILANKA
      <li>PAKISTAN
      <li>AUSTRALIA
      <li>SOUTH AFRICA
    <ol>
  </body>
</html>
```

➤ Output

My favorite cricket teams

- b. INDIA
- c. SRILANKA
- d. PAKISTAN
- e. AUSTRALIA
- f. SOUTH AFRICA

4.1 Numbered List

➤ [NumberedList.html](#)



4.2: Bulleted List

Bulleted List

- List of items prefixed with bullets is called **Bulleted /Unordered List**
- tag
 - defines an unordered list.
 - An unordered list items will be marked with bullets (small black circles).
- Example

```
<!DOCTYPE html>
<html>
  <body>
    <h2>My favorite cricket teams</h2>
    <ul>
      <li>INDIA</li>
      <li>SRILANKA</li>
    </ul>
  </body>
</html>
```

➤ Output

My favorite cricket teams

- INDIA
- SRILANKA

To create a bulleted list use the (unordered list) tag at the opening of the list and at the end of the list.

4.2: Bulleted List

Attributes in tag

➤ Style Attribute:

- Can be used in tag to define the style of the marker.

Style Attribute Value	Description
list-style-type:disc	The list items will be marked with bullets (default)
list-style-type:circle	The list items will be marked with circles
list-style-type:square	The list items will be marked with squares
list-style-type:none	The list items will not be marked
list-style-image: url('logo.gif');	The list items will be marked with an image

➤ Example

- <UL style="list-style-type:circle"> will be used to prefix unordered list of items with square.

4.2: Bulleted List
Example (Code):

➤ Example

```
<!DOCTYPE html>
<html>
<body>
<h2>My favorite cricket teams</h2>
<ul style="list-style-type:square">
<li>INDIA</li>
<li>SRILANKA</li>
</ul>
</body>
</html>
```

➤ Output

My favorite cricket teams

- INDIA
- SRILANKA

4.2 Bulleted List

➤ **BulletedList.html**



4.3: Glossary List Glossary List

- List of terms, with a description of each term is called Glossary/ Description Lists.
- Tags can be used for creating glossary list are:
 - <DL> tag: Defines a description list.
 - <DT> tag: Defines the term (name).
 - <DD> tag: Defines the data (description).
- Example for Glossary/Description List:

HTML Tag

HTML tags are used to markup HTML elements. They normally come in pairs.

Title Tag

Title tag is used to display the page title in the browser window.

Glossary List:

In the given example, term is defined using <DT> tag are:

HTML Tag

Title Tag

In the given example, defined data are:

HTML tags are used to markup HTML elements. They normally come in pairs.

Title tag is used to display the page title in the browser window.

4.3: Glossary List
Glossary List (Code)

```
<!DOCTYPE html>
<html>
  <body>
    <dl>
      <dt>HTML Tag</dt>
      <dd>HTML tags are used to markup HTML elements. They normally
come in pairs. </dd>
      <dt>Title Tag</dt>
      <dd>Title tag is used to display the page title in the browser
window.</dd>
    </dl>
  </body>
</html>
```

4.3 Glossary List

➤ [GlossaryList.html](#)



Lab

➤ Lab 4



June 5, 2015

Proprietary and Confidential

- 15 -

IGATE
Speed. Agility. Imagination.

Lesson Summary

➤ **In this lesson, you have learnt about:**

- Types of available list items
- Create bulleted, numbered and glossary listed items



Review Question: Questions

➤ **Question 1:** Which tag is used at the beginning of an Ordered List

-
-
-

➤ **Question 2:** To create a bulleted list tag is used

- True/ False

➤ **Question 3:** _____ attribute is used to modify the appearance of bullets



Review Question: Match the Following

1. <dd>	a. 1-Standard numbers
2. <dt>	b. Square Form
3. <OL TYPE=I>	c. A-Uppercase letters
4. <OL TYPE=A>	d. I-Uppercase Roman letters
5. <OL TYPE=1>	e. Data Definition
6. <UL style="list-style-type:circle">	f. Data Term



Web Basics – HTML5

Lesson 5. Working with Links

June 5, 2015

Proprietary and Confidential

- 1 -

IGATE
Speed. Agility. Imagination

Lesson Objectives

➤ After completing this module, you will be able to:

- Understand the working of hyperlinks in web pages.
- Learn to create hyperlinks in web pages.
- Add hyperlinks to list items and table contents.



5.1 Hyperlink in HTML

- Links allow users to navigate and find information on the Internet as easy as possible.
- HTML links are hyperlinks.
- A hyperlink is a text or an image you can click on, and jump to another document.
- Tag to create a link is `<a>` which stands for anchor.
- Syntax:

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

- By default, links will appear as this in all browsers:
 - An unvisited link is underlined and text is highlighted in blue color (Click me)
 - A visited link is underlined and text is highlighted in purple color (Click me)
 - An active link is underlined and text is highlighted in red color (Click me)

June 5, 2015

Proprietary and Confidential

- 3 -

IGATE
Speed. Agility. Imagination

Hyperlinks make navigating and finding information on the Internet as easy as possible. When information is published on the Internet, hyperlinks allow readers to access related information on other Web pages or Web sites.

What makes HTML Hyper?

Hyper means more than or outside of. A hyperlink takes you to another page that gives you more than what was on the original page. That page is outside the main page, although it is connected to it.

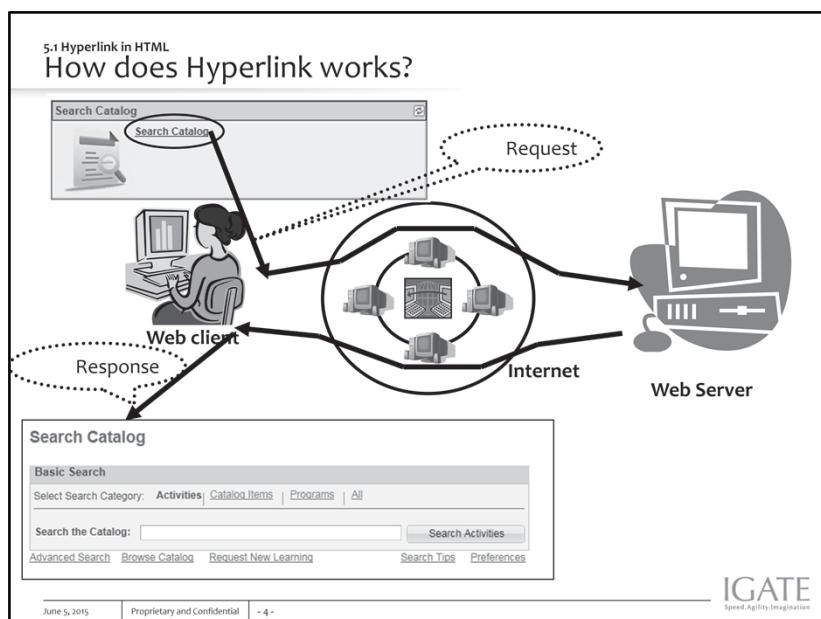
Create Link to a Local Page

Anchor Tag

Tag to create a link is `<A>`, which stands for anchor. Anchor tag begins with `<A>` and is followed by a reference comment (HREF) and the name of the page or the URL where the page to link resides. After closing the initial anchor command, you type in the text you wish to highlight as a link, and then close the tag with ``.

E.g.: The `news ` on Indian Festival.

The link above displays the word news in blue with an underline. When you click it, you see the web page named news.htm.



How does hyperlink works?

In the given example, when user clicks on search catalog link request is sent to the server via internet. In the server, request is processed and “Search catalog” page as response sent back to the web client for users view.

5.2 Hyperlink Work

Create Links

- Link to local document in an application(Relative linking)

```
<a href="contactus.html">Click Here</a>
```
- Link to local document in an application(Absolute linking)

```
<a href="http://localhost:8080/Example/contactus.html">Click Here</a>
```
- Link to web document

```
I have some <a href="http://www.state.edu/info/info.htm">information</a> about education.
```

June 5, 2015 | Proprietary and Confidential | - 5 -

IGATE
Speed.Agency.Imagination

Create Link to Other Pages

Use Hypertext Transfer Protocol (HTTP), to connect to one computer and transfer a copy of documents from that computer to a browser that runs on another computer. Place the document URL in the anchor of a link using the HREF attribute.

For example:

I have some <A HREF=<http://www.stateu.edu/info/info.txt>> information about education

5.2 Hyperlink Work

Links to an E-Mail

➤ **Link to an electronic mail:**

- An URL tag called “mailto” identifies the address to which an e-mail has to be sent.
- Syntax:

```
<a href="mailto:mailid?subject =content&cc=cc-mailid&bcc=bcc-mailid">Link  
text</a>
```

- Examples:

```
<a href="mailto:eDude@igate.com?subject =More Info">Contact Me</a>  
Please <a href="mailto:author@igate.com>mail</a>your comments to me
```

IGATE
Speed. Agility. Imagination

June 5, 2015 | Proprietary and Confidential | - 6 -

Create Link to an Electronic Mail

HTML includes a URL tag called mailto, which identifies the address to which an e-mail is to be sent to. If the link is selected, the browser starts an email program to send a message to the recipient listed in the URL.

For example:

Send me <A HREF="mailto:abc@igate.com?subject=My
comments">e-mail with your comments about my page.

CC and BCC in Email (Code)

```
<!DOCTYPE html>
<html>
<body>
<!----add the CC and BCC's-->
<a
href="mailto:edude@igate.com?Subject=Hello&Cc=jane@igate.com&bcc=joe@i
gate.com"> Mail us </a>
</body>
</html>
```

5.2 Hyperlink Work

Links to Specific Part of a Page

- id attribute, used in the anchor tag identifies a section of a page.
- Set HREF to the section name. Precede the name with a # symbol

```
<a id="address1"></a>
IGATE Global Solutions, Unit No. 134/135, SDF V,
Seepz, Andheri-East, Mumbai 92
... Contents of the page...
I am working in <a href="#address1">IGATE</a>
```

June 5, 2015 | Proprietary and Confidential | - 8 -

IGATE
Speed. Agility. Imagination

For documents located on other computers, attach the section name, preceded by the # symbol, at the end of the document URL.

E.g. I am working in

`IGATE`

Create Link to A Specific Part of A Page

ID attribute, used in the anchor tag, identifies a section of a page. Subsequently, users can access this section via a link within the document or other documents.

To include a link to the named section elsewhere in your page, create a link and set HREF to the name of the section as defined by the ID attribute. Precede the name with a # symbol to differentiate it from the name of another document.

For example:

``

`< -- Text of section 1-- >`

You find the relevant information in

`section 1.`

To include a link to the named section from another local document, create a local link and include the name of the section, preceded by the # symbol, in the location of the link defined by the HREF attribute.

Demo

➤ Linkall.html



June 5, 2015

Proprietary and Confidential

- 9 -

IGATE
Speed. Agility. Imagination

```
<!DOCTYPE html>
<html>
<head>
<title>Using links</title>
</head>
<body >
<a href="6image1.html" target="_blank" > link to a local page</a>
<p>
<a href="http://eclassroom.igate.com" > link a site</a>
<p>
<p>
<a href="mailto:eDude@igate.com?subject=More Info"> mail</a>
your comments to me
</body>
</html>
```

5.3: Hyperlinks in Lists Items

Hyperlinks in Lists Items (Code)

- Add links to list items and table contents:

```
<ul>
<li><a href=home.html>mumbai</a></li>
<li><a href=home.html>pune</a></li>
<li><a href=home.html>nasik</a></li>
</ul>
```

5.4: Hyperlinks in Table Elements

Hyperlinks in Table Elements (Code)

- Add links to table contents:

```
<table border=1>
<tr><th>team</th><th>points</th><th>grade</th>
<tr><td>
<a href=home.html>mumbai</a></td><td>90</td><td>a</td>
</tr>
<tr><td>
<a href=home.html>pune</a></td><td>86</td><td>b</td>
</tr>
<tr><td>
<a href=home.html>nasik</a></td><td>80</td><td>c</td>
</tr>
</table>
```

5.4: Hyperlinks in Table Elements

Demo

➤ Linkintable.html



June 5, 2015 | Proprietary and Confidential | - 12 -

IGATE
Speed.Agency.Imagination

```
<!DOCTYPE html>
<html>
<head>
<title>Links in lists and table Example</title>
</head>
<body>
<ul>
<li><a href=home.html>mumbai</a>
<li><a href=home.html>pune</a>
<li><a href=home.html>nasik</a>
</ul>
<table border=1>
<tr><th>team</th><th>points</th><th>grade
<tr><td><a href=home.html>mumbai</a><td>90<td>a
<tr><td><a href=home.html>pune</a><td>86<td>b
<tr><td><a href=home.html>nasik</a><td>80<td>c
</table>
</body>
</html>
```

5.5: Providing Target for a Hyperlink

Providing Target for a Hyperlink

- **Target attribute of the <a> tag specifies where to load the linked document.**
`Document 2`
- **It takes following values:**
 - name of the existing window/target
 - _blank
 - _self
 - _parent
 - _top
- **By default it loads the linked document in the current window.**

5.5: Providing Target for a Hyperlink

_blank (Code)

- “_blank”:
 - opens a new document in a new window.
- This code produces this:

a new window

IGATE
Speed. Agility. Imagination

June 5, 2015 | Proprietary and Confidential | - 14 -

Using target="_blank"

This causes the link to open in a totally new browser window, with the page with the referring link still open behind it. Unlike JavaScript pop ups, the developer has no control over the size of the resulting window. It depends on what the browser did the last time before it was closed.

5.5: Providing Target for a Hyperlink

_self (Code)

➤ “_self”:

- Puts new document in the same window and frame as the current document.
- Works the same as if you had not used TARGET at all.

➤ This code produces this go to:

```
<A HREF="selftarget.html" TARGET="_self">next</A> page
```

Using target="_self"

This loads the page within the same frame as the link tag.

5.5: Providing Target for a Hyperlink

_top (Code)

➤ “_top”:

- loads linked document in the topmost frame. This means that the new page fills the entire window.

```
<A HREF="selftarget.html" TARGET="_top">top</A>
```

Using target="_top"

This, within a link tag causes the new page to load in the full body of the window. It is useful if you want to break out of the frameset you have created in order to have a frameless page.

5.5: Providing Target for a Hyperlink

_parent (Code)

➤ “_parent”:

- Used when a frameset file is nested inside another one
- A link in one of the inner frameset documents which uses “_parent” loads the new document where the inner frameset file had been

```
<A HREF="bigframe.html" TARGET="_parent">bigframe</A>
```

June 5, 2015 | Proprietary and Confidential | - 17 -

IGATE
Speed. Agility. Imagination

Using target=_parent"

This is similar to target=_top" but refers to the immediate parent of a frame. In more advanced frame usage there may be several nested frames. This allows more control over which frames are specified. (It is actually something developers rarely need to use).

5.5: Providing Target for a Hyperlink

Demo

- [Linktarget.html](#)



LAB

➤ Lab 4



June 5, 2015

Proprietary and Confidential

- 19 -

IGATE
Speed. Agility. Imagination

Summary

➤ In this lesson, you have learnt about

- How hyperlink works in web pages
- Creation of hyperlinks in web pages
- Including hyperlinks to list items and table contents



Review Question

- **Question 1:**The id attribute along with # used in the anchor tag identifies:
 - Page section.
 - Path to load the linked document.
 - Address to send the e-mail to.
- **Question 2:**URL tag mailto, identifies the address from where e-mail is to be received.
 - True/ False
- **Question 3:**_____ attribute of the <a> tag specifies where to load the linked document.



Review Question: Match the Following

1. “_blank”

2. “_self”

3. “_top”

4. mailto

5. Relative Linking

6. Absolute Linking

a. Identifies address to send the e-mail to.

b. Only filename is needs to be specified.

c. Gives entire address of the page.

d. Puts the new document in the same window and frame as the current document

e. Opens the new document in a new window.

f. Loads linked document in the topmost frame.

Web Basics – HTML5

Lesson 6. Image Handling

June 5, 2015

Proprietary and Confidential

- 1 -

IGATE
Speed. Agility. Imagination.

Lesson Objectives

- After completing this module, you will be able to:
- Understand the role of images in web pages
 - Learn to add images to web pages
 - Learn to use images as hyperlinks



6.1 Roles of images in web pages

- Web pages are comprised of text and images.
- Some of the roles of images in web pages are
 - To make web page content more interesting to be read
 - To increase clarity on web page content by transmitting right message
 - Provoking reactions
 - To ensure that the user response is anticipated
- Some of the widely used image formats
 - GIF : Graphical Interchangeable Format
 - JPEG : Joint Photographic Expert Group
 - PNG : Portable Network Graphics

GIF images are quickly downloadable, because its size is less. It is used for animated images, thumbnails, B/W image . It supports only 256 colors

JPEG images uses **lossy** compression method, meaning that some original image information is lost and cannot be restored, possibly affecting image quality.

PNG images are lossless, portable, well-compressed storage of raster images. It supports image transparency

6.2 Adding images in webpages

- Images are added into a HTML document using tag.
- tag has 2 required attributes such as src and alt
- Syntax:

```

```

- The src attribute defines the url (web address) of the image
- Alt attribute defines alternative text to display for non-graphical browsers. The alt attribute is required

- Example:

```

```

GIF images are quickly downloadable, because its size is less. It is used for animated images, thumbnails, B/W image . It supports only 256 colors

JPEG images uses **lossy** compression method, meaning that some original image information is lost and cannot be restored, possibly affecting image quality.

PNG images are lossless, portable, well-compressed storage of raster images. It supports image transparency

6.2: Adding images in webpages

Attributes of an Inline Image

- Other attributes of tag are:
 - Width: Sets the width of the image.
 - Height: Sets the height of the image.
 - ismap: Specifies an image as a server-side image-map
 - usemap: Specifies an image as a client-side image-map

June 5, 2015

Proprietary and Confidential

- 5 -

IGATE
Speed.Agency.Imagination

After you have the image, save it in GIF format. Then, use the tag to include the image in your document. The tag is empty, which means that it contains attributes only and it has no closing tag.

Following list shows attributes and options you can include inside the tag:

Alt: Alternative text to display for non-graphical browsers

Align: Aligns graphic with surrounding text. Values include left, right,

and center

6.2: Adding images in webpages

Text & Image Alignment (Code)

➤ **Style Attribute:**

- Can be used in tag to define the alignment of text and image.

Style Attribute Value	Description
vertical-align:bottom	The bottom of the element is aligned with the lowest element on the line
vertical-align:middle	The element is placed in the middle of the parent element
vertical-align:top	The top of the element is aligned with the top of the tallest element on the line
float:right	The element floats to the left
float:left	The element floats to the right
float:none	The element floats to the none

➤ <BR CLEAR=LEFT/RIGHT> can be used to stop text flowing around an image

June 5, 2015

Proprietary and Confidential

- 6 -

IGATE
Speed. Agility. Imagination.

Insert the <BR CLEAR=LEFT/RIGHT> command at the spot in the descriptive text where text is to stop flowing around the image.

6.2: Adding images in webpages

Demo

- [ImageDemo.html](#)
- [ImageAlign.html](#)

The Example of image alignment

HTML allows for simple aligning of text and images by adding the style attributes to the tag. HTML 5 expanded this attribute by specifying float:left and float:right options. When you use these options, text flows down the right or left side of an image aligned to them. The text flow continues until either the text passes the image, the text ends, or the browser encounters a tag.



HTML5 allows vertical alignment of text and images by adding the style attribute with values as vertical-align:bottom, vertical-align:top... When you use these options, text is vertically aligned top/bottom/middle. The text flow continues until either the text passes the image, the text ends, or the browser encounters a br tag with clear attribute.



6.3: Image as Link

Use of Image as a Hyperlink (Code)

- Images used as hyperlinks:

```
<A href="6image1.html"></a>
```

- Images contained within a table:

```
<table>
<tr><th>Product</th><th>Cost</th><th>Image</th></tr>
<tr> <td>Pencil</td> <td>$8</td>
      <td></td> </tr>
<tr> <td>Brush</td> <td>$15</td>
      <td></td></tr></table>
```

June 5, 2015

Proprietary and Confidential

- 8 -

IGATE
Speed.Agency.Imagination

Use Image as Link

To create an image link, place the **** tag with the image URL between **.....** tags.

```
<H2>MY FIRST IMAGE LINK</H2>
<A HREF="BALLOON.HTM"><IMG
SRC="BALLOON.GIF"></A>Click on it to have
additional information on it.<P>
```

6.3: Image as Link
Demo

➤ [ImageLink.html](#)



June 5, 2015

Proprietary and Confidential

- 9 -

IGATE
Speed. Agility. Imagination

6.4: Image Maps

Image Maps

➤ **Image Maps come in two flavours:**

- Server-side maps: Contain linking information on the server
- Client-side maps: Linking information is bundled along with the HTML document

June 5, 2015

Proprietary and Confidential

- 10 -

IGATE
Speed. Agility. Imagination

Difference Between Client Server-side Image Maps:

In Server-side image maps, when you click an image leading to multiple links, you actually invoking a cgi-bin program. The program reads a separate map file that tells it where to go based on the section of the image you click. The program returns the destination to the web browser, which opens the page. Image map is server-side because the web browser needs to contact the remote host to know which site it should contact.

Client-side image maps, in contrast, do not require a cgi-bin program to function. The image map is actually an HTML construct that can be contained on the same page as the clickable image. Rather than the program on the remote host figuring out what site to go to, the web browser itself knows what regions of the image are associated with which sites. The browser, rather than the remote host, interprets the image map.

Client-side image maps are usually faster than server-side image maps and, in general, easier to create.

Their main disadvantage is that older browsers, and less fully featured ones, may not support them. Fortunately, it is usually possible to create a clickable image that can have both a server-side and a client-side image map.

6.4: Image Maps

Clickable Image Maps – Client Side

➤ Steps to include client-side image maps:

- Identify areas on the image that you need hyperlink to a document.
 - Get their pixel locations
- In HTML documents, do the following:
 - Insert the <map> tag and specify the name.
 - Insert <area> tag between <map> tag. Specify the shape attribute value.
 - Specify coords attribute, to identify corners of the image map area.
 - Attribute href identifies the document to load if this area is selected.

June 5, 2015

Proprietary and Confidential

- 11 -

IGATE
Speed. Agility. Imagination.

Include Client-side Image Maps

Netscape and Microsoft's Internet Explorer support client-side image maps in which the different regions of the image are specified within the document or in another HTML document. No CGI applications are required to read client-side image maps. Therefore, they work even while not connected to the Internet. Perform the following steps:

1. Open your document, and then load an image with a viewer such as Photoshop. This allows you to get the pixel locations from the image so you can specify them in links.
2. Identify areas on the image that you wish should serve as links.
3. Note pixel locations of the upper-left and lower-right corners of the rectangle.
4. In the HTML document, insert the <MAP> and </MAP> tags.
5. Use the NAME attribute for <MAP> to give a unique name to the image map.
6. Between <MAP> and </MAP> tags, insert an <AREA> tag. There should be one <AREA> tag for each link on the map.
7. SHAPE attribute of <AREA> identifies the shape of the area in the image map. As only rectangles are currently supported, SHAPE takes only one value, "RECT".
8. COORDS attribute identifies boundaries of the area on the image map. For RECT, COORD has four values, separated by commas: x and y coordinates of the upper-left and lower-right corners.

6.4: Image Maps

Clickable Image Maps – Client Side (Contd) (Code)

- **Usemap attribute of refers to the image map**

```
<map name="mymap">
<area shape="RECT" coords="10,10,30,50" href="info1.htm"/>
<area shape="RECT" coords="50,50,70,70" href="info2.htm"/>
<area shape="RECT" coords="90,90,120,95" href="info3.htm"/>
</map>

```

For each area, the HREF attribute identifies the URL of the document that should be loaded if the user selects that area. Document can be a local file or a document on another server

You can reference the image map information to an image on the page by adding the USEMAP attribute to the element. USEMAP is set equal to the name of map information from the NAME attribute of <MAP>.

```
<IMG SRC="IMAGEMAP.GIF" ALT="An Imagemap"
      USEMAP="#mymap">
```

6.4: Image Maps Client-Side Maps (Code)

➤ Example:

```
<A Href="http://www.webdevelopersnotes.com/">  
<Img Src="home.gif" Width="152" Height="25" Alt="Back to homepage"/>  
</A>
```

Create Server-side Clickable Image Map

Server side maps also have a main viewable image (IMG) like client-side maps. Instead of the usemap attribute, they set only the Boolean attribute ismap.

When a user activates a region of a server-side image map with a mouse, pixel coordinates of the click are sent to the server-side agent specified by the href attribute of the 'A' element. Server-side agent interprets the coordinates and performs some action.

For server-side image maps, the client browser just sends the coordinates of a mouse click to the server where calculations are made and the appropriate page is sent back to the client. Because of this essentially visual (and mouse) method of selecting a new page, there is no way to make server-side image maps accessible.

With server-side image maps, "active regions" in the picture only display a set of coordinates. Browsers cannot indicate to the user the URL that will be followed when a region of the map is activated. Therefore, a redundant text link is necessary to provide access to the page.

Demo

➤ [clientsideimagemap.html](#)



June 5, 2015

Proprietary and Confidential

- 14 -

IGATE
Speed. Agility. Imagination

```
<!DOCTYPE html>
<html>
<head><title>Client-Side Image Map</title></head>
<body>
<map name="menu">
<area shape=rect coords="4,4,96,36" href="home.htm" >
<area shape=poly coords="5,39,95,104,95,39,5,39" href="fun.htm">
<area shape=poly coords="4,44,95,113,6,114,4,44" href="games.htm">
<area shape=circle coords="51,164,36" href="work.htm">
<area shape=rect coords="16,215,89,284" href="6clientsideimagemap.htm">
<area shape=default href="6clientsideimagemap.htm">
</map>

</body></html>
```

Lab

➤ Lab 5



June 5, 2015

Proprietary and Confidential

- 15 -

IGATE
Speed. Agility. Imagination.

Lesson Summary

➤ In this lesson, you have learnt about:

- Role of images in web pages
- How to add images in web pages
- How to use images as hyperlinks



Review - Questions

- Question 1: Client-side image maps are usually faster than server-side image maps.
 - True/ False
- Question 2: _____ contain the linking information on the server.



Web Basics – HTML5

Lesson 7. Frames

June 5, 2015

Proprietary and Confidential

- 1 -

IGATE
Speed. Agility. Imagination

Lesson Objectives

- **After completing this module you will be able to:**
- Understand the need for frames in web pages.
 - Learn to create and work with iframes.



7.1 Introduction to FRAME

- With frames, more than one HTML document content can be nested in a web page.
- Need for frames in webpage:
 - Reusability
 - Static navigation can be visible all the time
 - Design issues can be resolved.
 - Advertisement can be displayed which promotes web site

June 5, 2015

Proprietary and Confidential

- 3 -

IGATE
Speed.Agency.Imagination

Iframe:

- Inline Frame Element (`<iframe>`) is used to embed another HTML page into the current page.
- `<iframe>` can be used within a HTML document body.
- A section that contains embedded content is called the parent context. For an example, if `<iframe>` is used as a nested tag in `<body>`, then body is the parent context and browser window is the top level parent context.

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

Disadvantage of using frames is that the web developer must keep track of more HTML documents. Also, it is difficult to print the entire page.

7.2 Introduction to IFRAME

- **Inline Frame Element (`<iframe>`)** is used to embed another HTML page into the current page.
- `<iframe>` can be used within a HTML document body just like an `` tag.
- **Syntax:**

```
<iframe src="url"></iframe>
```

- **Types of documents which can be embedded using IFRAME** are
 - Image(GIF, JPEG, etc..)
 - PDF
 - Audio
 - Video
 - HTML document, etc..

- *Inline Frame Element (`<iframe>`)* is used to embed another HTML page into the current page.
- `<iframe>` can be used within a HTML document body just like an `` tag.
- A section that contains embedded content is called the parent context.
- Types of documents which can be embed using IFRAME are
 - Image(GIF, JPEG, etc..)
 - PDF
 - Audio
 - Video
 - HTML document, etc..
- Advantages of using IFRAME
 - Common content like header, footer, .. For multiple pages can be described using IFRAME
 - By displaying advertisement, web page can be promoted easily.

7.2 Introduction to IFRAME

IFRAME Tag Attributes

Attribute Name	Attribute Value	Description
name	Name of the frame	Mentions the name of iframe.
height	pixels	Denotes the height level of iframe.
width	pixels	Denotes the width level of iframe.
src	URL of the file	The address/URL of the content to be fixed in iframe.
sandbox	"", allow-forms, allow-same-origin, allow-scripts, allow-top-navigation	Creates limitations for iframe contents.
seamless	seamless	Defines that iframe should look like the part of the containing document.
srcdoc	HTML code	The HTML content of a page to be shown in iframe.

June 5, 2015 | Proprietary and Confidential | - 5 -

IGATE
Speed.Agency.Imagination

Frame :

<frame> tag defines which HTML document to put into each frame.

If a frame has visible borders, the user can resize it by dragging the border. To prevent a user from doing this, you can add noresize="noresize" to the <frame> tag.

Second tag used to create frames is the <FRAME> tag. It defines a frame within a frameset and has six possible attributes. <FRAME> tag syntax is:

```
<frame src="url" name="window_name" marginwidth="value"
marginheight="value"
scrolling="yes/no/auto" {noresize}>
```

SRC is the URL of the source document to be displayed in this frame.

NAME is an optional attribute that assigns a name to a frame so that it can be used by links in other documents or by JavaScripts.

MARGINWIDTH is an optional attributes used to set the width of the margins of the frame. The value is in pixels.

7.2 Introduction to IFRAME

Example

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

<iframe src="HTML5.pdf" width="600" height="400">
<p>Your browser does not support iframes.</p>
</iframe>

<p>Inline Frame Element is used to embed another HTML page into the current page. can be used within a HTML document body just like an tag. A section that contains embedded content is called the parent context.</p>

</body>
</html>
```

June 5, 2015 | Proprietary and Confidential | - 6 -

IGATE
Speed.Agency.Imagination

Frame :

`<frame>` tag defines which HTML document to put into each frame.

If a frame has visible borders, the user can resize it by dragging the border. To prevent a user from doing this, you can add `noresize="noresize"` to the `<frame>` tag.

Second tag used to create frames is the `<FRAME>` tag. It defines a frame within a frameset and has six possible attributes. `<FRAME>` tag syntax is:

```
<frame src="url" name="window_name" marginwidth="value"
marginheight="value"
scrolling="yes/no/auto" {noresize}>
```

SRC is the URL of the source document to be displayed in this frame.

NAME is an optional attribute that assigns a name to a frame so that it can be used by links in other documents or by JavaScripts.

MARGINWIDTH is an optional attributes used to set the width of the margins of the frame. The value is in pixels.

Demo

- [iframesDemo1.html](#)
- [iframesDemo2.html](#)



June 5, 2015

Proprietary and Confidential

- 7 -

IGATE
Speed...Agility...Imagination

```
<HTML>
<HEAD><TITLE>Frames</TITLE></HEAD>
<!—Dividing the Web document into three parts using frameset
First frameset divides the browser horizontally into two parts
First half contain the Top. Html page
Second half again gets divided vertically into two parts ‘A’ Part
showing the ‘left.htm’ page and ‘B’ part further get divided horizontally
into two parts x part showing the ‘main.htm’ and y part showing ‘blank ’ -->
<FRAMESET rows=10%,*>
    <FRAME border=0 name=top src="top.htm" noResize
scrolling=no>
    <FRAMESET cols=20%,*>
        <FRAME name=left src="left.htm">
<FRAMESET rows=70%,*>
    <FRAME name=main src="main.htm">
    <FRAME name=details src="about:blank">
</FRAMESET>
</FRAMESET>
</FRAMESET>
</HTML>
```

Lab

➤ Lab 7



June 5, 2015

Proprietary and Confidential

- 8 -

IGATE
Speed. Agility. Imagination

Lesson Summary

➤ In this lesson, you have learnt about

- Need of frames in webpages
- Embedding of content in HTML document using <iframe> tag.



Review – Match the Following

1. SRC	a. Sets the height of the frame.
2. NAME	b. Sets the width of the frame.
3. WIDTH	c. Creates limitations for iframe contents.
4. HEIGHT	d. Assigns a name to a frame.
5. SANDBOX	e. Source document's URL is displayed in this frame.



Web Basics – HTML5

Lesson 8. HTML Forms for User Input

June 5, 2015

Proprietary and Confidential

- 1 -



Lesson Objectives

➤ After completing this module you will be able to:

- Understand the role of forms in web pages
- Understand various HTML elements used in forms.



8.1 HTML Forms for User Input

- HTML forms are used to accept user inputs and then submit data for processing.
- A **form** is an area that contains form elements.
- Types of elements which can be included in a form are
 - Label
 - Single line text field
 - Password field
 - Text area
 - Drop down menu
 - Radio button
 - Checkbox
 - File selector box, etc..

Forms:

HTML forms are used to accept of user input.

A form is an area that contains form elements.

Form elements are elements that allow users to enter information

(text fields, text area fields, drop-down menus, radio buttons, checkboxes, etc.) in a form.

Define a form with the <form> tag.</FORM>

8.1 HTML Forms for User Input

- User input forms are created using <form> tag.

- Syntax:

```
<form method="get/post" action="URL" enctype="Encryption Type">  
    Field definitions  
</form>
```

- **action:** the URL of the script
- **method:** the HTTP request method to use, sometimes GET, but usually POST
- **enctype:** Specifies how the data is to be encoded.

Basic syntax for the <FORM> tag is:

```
<FORM METHOD="Get or Post" ACTION="URL" ENCTYPE="type">  
    Field definitions  
</FORM>
```

<FORM> tag tells a browser that there is a *fill-in-the-blank* form in this HTML document.

Method:

METHOD attribute states the method to use when you send the form to the server. Two acceptable methods are GET and POST.

GET sends information entered in the form to the server at the end of the URL.

POST sends information entered in the form to the server as a data body/document.

ACTION attribute:

Gives the address of the script that processes the form.

Defines the name of the file to send the content to. File defined here typically does something with the received input.

Enctype:

Specifies how the data is to be encoded.

Applies only if you use the POST method. There is only one possible value, the default value “application/x-www-form-urlencoded”.

8.1 HTML Forms for User Input

- Some more attributes which can be used in <form> tag are

Attribute Name	Attribute Value	Description
name	Form name as a string	Mentions the name of a form.
autocomplete	On, off	Specifies whether a form should have autocomplete on or off
target	_blank, _self, _parent, _top	Specifies where to display the response that is received after submitting the form
novalidate	novalidate	Specifies that the form should not be validated during form submission.

8.2 HTML Form Elements

- <input> element is the most used form tag.
- An <input> tag includes the following attributes
 - name: Name of the field which is required to send data(Key/Value pair) during form submission
 - id: A unique identified of the field
 - value: Sets a default value of the field
 - maxlength: Specifies the maximum number of characters allowed in an <input> element
 - readonly: Specifies that an input field is read-only
 - size: Specifies the width, in characters, of an <input> element
 - Disabled: specified that an input element should be displayed.
 - type attribute of <input> tag specifies the field type

Input:

The most used form tag is the <input> tag. Input type is specified with the type attribute. Type attributes are:

- Text
- Password
- Hidden
- Radio
- Checkbox
- File
- Button
- Submit/reset

8.2 HTML Form Elements

Text- related Elements

- Text related elements can be created as shown below:

Code	Element
<input type="text">	Single line text box
<input type="password">	Password field
<input type="hidden">	Hidden field

➤ Multiple line text input control

- If input exceeds more than one line, then create Multi-line input control using HTML <textarea> tag
- Syntax:

```
<textarea rows="" cols="" name="">
```

- Rows : Number of rows of text area box
- Cols: Number of columns of text area box
- Name: name of the element

8.2 HTML Form Elements

Checkbox Element

- If more than one option is required to be selected from multiple options, then create checkbox as shown below:

- <input type="checkbox">
- Use checked attribute for selecting any checkbox to be selected by default

➤ Example:

```
<input type="checkbox" name="hobbies" value="Reading Books"> Reading Books  
<input type="checkbox" name="hobbies" value="Net Surfing"> Net Surfing  
<input type="checkbox" name="hobbies" value="Singing" checked> Singing
```

Select your Hobbies: Reading Books Net Surfing Singing

8.2 HTML Form Elements

Radio Button

- If only one option is required to be selected from multiple options, then create radio button as shown below:

- <input type="radio">
- Use checked attribute for making a radio button to be selected by default

➤ Example:

```
<input type="radio" name="sector" value="Public"> Public  
<input type="radio" name="sector" value="Private"> Private
```

Select your sector in which you are working: Public Private

8.2 HTML Form Elements

Drop down list

- Drop down list allow the user to select one or more values from a pre-determined options
- Tags for creating drop down list with options are:
 - <SELECT> : Creates drop down list
 - <OPTION>: Defines an option in a select list.

Tag name	Attribute	Description
<SELECT>	Name	Defines a name for the drop down list
	Size	Defines the number of visible options in a drop down list
	Multiple	Allow to select multiple options at once
	Disabled	Disable drop down list
<OPTION>	value	Specifies the value to be sent to a server
	Selected	Makes option to be selected by default

8.2 HTML Form Elements

Drop down list - Example

```
<!DOCTYPE html>
<html>
<body>Select a country:
<select name="country">
<option value="Germany">Germany</option>
<option value="India" selected>India</option>
<option value="China">China</option>
<option value="Japan" >Japan</option>
</select>
</body>
</html>
```

Select a country: India ▾

June 5, 2015

Proprietary and Confidential

- 11 -

IGATE
Speed.Agency.Imagination

8.2 HTML Form Elements
File Upload

- File upload will allow the user to upload a file from the desktop to an application in browser.
- The below code is used to define a file-select field and a "Browse..." button (for file uploads):
 - <input type="file"/>

Attribute	Description
Name	Defines a name for the file upload dialog box
Disabled	Disable element
Accept	Specify MIME type to describe the file type which accepts by a server

- In <form> tag, use enctype="multipart/form-data" if file need to be uploaded using file selector form element.

8.2 HTML Form Elements
File Upload- Example

```
<!DOCTYPE html>
<html>
<body>
    <form method="post" action="success.html"
          enctype="multipart/formdata">
        Select a photo to upload:
        <input type="file" name="photo"/>
    </form>
</body>
</html>
```

Select a photo to upload:

8.2 HTML Form Elements

Button

- Different types of button which is possible to be created in HTML5 are as shown below:

Field type	Element	
<input type="button">	Button	A clickable button, that activates a JavaScript when it is clicked
<input type="submit">	Submit button	Defines a button for submitting a form
<input type="reset">	Reset button	Define a reset button (resets all form values to default values)

Single Line Text Field

<INPUT> tag: Add an <INPUT> tag to accept some input from the reader. This is a standalone tag. This is the default input type .Syntax is as follows:

```
<input type="text" name="UserID" size=20  
value="Guest" maxlength=unlimited [disabled] [readonly]>
```

Give the user the option to submit this form or reset it and enter fresh values. To do this, use the <INPUT> tag and set TYPE to either “SUBMIT” or “RESET”.

```
<input type="submit" name="submit_button" value="Submit">  
<input type="reset" name="reset_button" value="Reset">
```

TYPE attribute tells the browser the type of button used. NAME is a variable that you can access later when referring to this information. VALUE is button caption.

Text Fields: These are used to type letters, numbers, etc. in a form.

Example:

```
<form>  
    First name: <input type="text" name="firstname">  
    <br>  
    Last name:<input type="text" name="lastname">  
</form>
```

Text Area:

The <textarea> tag defines a multi-line text input control.

A text area can hold an unlimited number of characters. Text renders in a fixed-width font (usually Courier). You can specify text area size with cols and rows attributes. An even better way to do it is through CSS height and width properties.

```
<textarea rows="4" cols="20">  
<textarea name="name" rows="10" cols="50" [disabled][readonly]>  
    Default Text  
</textarea>
```

```
<textarea name="address" rows=5 cols=10>  
    Please write your address</textarea>
```

Demo

- [CheckboxExample.html](#)
- [FormExamples.html](#)
- [PasswordField](#)
- [RadioButtonExample](#)
- [RadioExample](#)
- [TextFieldExample.html](#)



```
<!DOCTYPE html>
<html>
<head>
<title>form examples</title>
</head>
<body bgcolor="ivory">
<form name="form1" action="store.html" method="post">
<p>
<b>Enter first name</b> : <input name="username"><BR><br>
<b>Enter lastname</b> : &nbsp; <input maxlength="30" name="surname">
</p>
<p>
<b>Enter&nbsp; address :</b>&nbsp;&nbsp;&nbsp;
<textarea name="addr" rows="3"></textarea>
<br>
<br> <b>Select the training programs attended :</b><input type="checkbox" value="internet/html" name="internet-html">
Internet/HTML <input type="checkbox" checked value="c programming" name="c-programming"> C Programming <input type="checkbox" value="dbms-sql" name="dbms-sql"> DBMS-SQL
</p>
<p>
<b>Select the stream you belong to :</b> <input type="radio" value="science" name="s-grp"> Science <input type="radio" value="arts" name="s-grp"> Arts <input type="radio" value="commerce" name="s-grp"> Commerce <input type="radio" value="oth2" name="s-grp">Engineering
</p>
```

```
<p>
<b>Select the stream you belong to :</b> <input
type="radio" value="science" name="s-grp"> Science <input
type="radio" value="arts" name="s-grp"> Arts <input
type="radio" value="commerce" name="s-grp"> Commerce <input
type="radio" value="oth2" name="s-grp">Engineering
</p>
<b>Which training program would you like to attend ?</b> <select
multiple size="3" name="pref">
<option value="ih" selected>Internet-HTML
<option value="js">javascript
<option value="vbs">VBscript
<option value="as">ASP
<option value="xm">XML
<option value="jv">JAVA
<option value="jsp">jsp</option>
</select> <br>
<br> <b>Select the location of your resume</b> <input
type="file" size="15" name="fnm"> <br> <input
type="hidden" value="patni" name="coname"> <br> <input
type="button" value="exit" name="but"> <input type="submit"
value="save"> <input type="reset" value="reset">

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

Demo

➤ FormSubmission.html



June 5, 2015

Proprietary and Confidential

- 18 -

IGATE
Speed. Agility. Imagination

```
<HTML>
<HEAD>
<TITLE>Form Submission</TITLE>
</HEAD>
<BODY>
<form method="post" action="store.html">
<P>Enter UserID:<INPUT id=text1 name=text1></P>
<P>Enter Password:
<INPUT id=password1 type=password name=password1></P>
<P>
<INPUT id=submit1 type=submit value=Submit name=submit1> &nbsp;
<INPUT id=reset1 type=reset value=Reset name=reset1></P>
</form>
</BODY>
</HTML>
```

Lab

➤ Lab 6



June 5, 2015

Proprietary and Confidential

- 19 -

IGATE
Speed. Agility. Imagination

Summary

➤ **After completing this module you now:**

- Understand the role of forms in web pages
- Understand various HTML elements used in forms like:
 - Single line text field
 - Text area
 - Check box
 - Radio buttons
 - Password fields
 - Pull-down menus
 - File selector dialog box



Review Question

- **Question 1:** Radio Buttons are used when you want the user to select:
 - Option 1: one of a limited number of choices.
 - Option 2: one or more options of a limited number of choices.
 - Option 3: many of unlimited number of choices.
- **Question 2:** METHOD attribute states the method to use when you send the form to the server.
 - True/ False
- **Question 3:** The _____ attribute of form specifies how the data is to be encoded.



Review Question: Match the Following

1. Check Boxes

2. Radio Buttons

3. Password fields

4. Hidden Input Type

5. <select> tag in
Drop Down List

6. <option> tag in
Drop Down List

a. Populate data in the list.

b. Create a select list.

c. Include form data without
rendering it to the user.

d. Entered text is shown as
asterisks.

e. Only one option of a limited
number of choices.

f. One or more options of a limited
number of choices



June 5, 2015 | Proprietary and Confidential | - 22 -

IGATE
Speed. Agility. Imagination

Add the notes here.

Web Basics – HTML5

Lesson 9. New Form Elements

June 5, 2015

Proprietary and Confidential

- 1 -

IGATE
Speed. Agility. Imagination.

Lesson Objectives

➤ In this lesson you will learn about:

- HTML5: Enhanced Form Elements
- An HTML5 Detection Library: Modernizr
- Using Canvas in HTML5



9.1: HTML5: Enhanced Form Elements

Introduction to HTML5 Enhanced Form Elements

- A Form is one of the most basic and essential feature of any web site
- HTML5 brings to the table several new input types, a total of 13
- HTML5 introduces these data types via the
`<input type="_NEW_TYPE_HERE_" />` format
- One of the key design decisions in HTML5 is backward compatibility
- It provides automatic validity of the fields as per the format
- The browser inspects the input type and if it finds that it is of a specific type

June 5, 2015

Proprietary and Confidential

- 3 -

IGATE
Speed.AGility.Imagination

Introduction to HTML Enhanced Form Elements:

A Form is one of the most basic and essential features of any web site.

The form elements available in HTML so far include the textbox, checkbox, radio, button, drop-down list, password and file picker. While these have sufficed so far, there is a clear need for newer form elements. The question is not just of newer form elements, but the ability to inject behaviour into existing form elements so that usability and validity, which is a cornerstone of any good UI, is given highest consideration.

One of the key design decisions in HTML5 is backward compatibility.

What this means is that if the new input types are not supported, then by default it falls back to `<input type="text"....>`, so it will be rendered as a plain text box, which the user can then fill data in.

Advantages of new input types:

- a) You get automatic validity of the fields as per the format. This means that the form is not going to get submitted if the value entered is not as per the default validation of that type
- b) The browser inspects the input type and if it finds that it is of a specific type, then it does something quite clear to aid the input of that data. For e.g. On the Smart Phones, which do not have a physical keyboard but instead a virtual keyboard, the keyboard that will be shown up will only contain keys that will aid the user in filling out the data.

9.1: HTML5: Enhanced Form Elements

Placeholder

- Place Holder - A placeholder is a textbox that hold a text in lighter shade when there is no value and not focused
- Syntax is –

```
<input id="first_name" placeholder="This is a placeholder">
```

- This is how place holder looks like on supporting browser

- Once the textbox gets focus, the text goes off and you shall input your own text

Firefox 3.7 Safari 4.0 Chrome 4.0 Opera 11 Android 2.2

IGATE Speed.Agency.Imagination

June 5, 2015 | Proprietary and Confidential | - 4 -

Place Holder:

The first improvement HTML5 brings to web forms is the ability to set placeholder text in an input field. Placeholder text is displayed inside the input field as long as the field is empty and not focused. As soon as you click on (or tab to) the input field, the placeholder text disappears.

Here's how you can include placeholder text in your own web forms:

```
<form>
  <input name="name" placeholder="Enter your name">
  <input type="submit" value="Search">
</form>
```

Browser's that don't support placeholder attribute will simply ignore it.
But if you

want to make it work in other browsers, you can use some **JavaScript** to create

The same behavior. There is an excellent **jQuery plugin** called **HTML5 Placeholder**

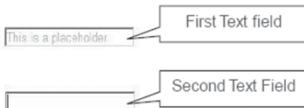
Plugin that will go through all input fields with placeholders attached to them and make them work in all browsers.

9.1: HTML5: Enhanced Form Elements

Auto Focus

- Auto Focus - Autofocus is a Boolean attribute of form field that make browser set focus on it when a page is loaded
- Syntax is –

<input id="Text2" type="text" autofocus/>



This is a placeholder

First Text field

Second Text Field



Firefox 4 Safari 4.0 Chrome 3.0 Opera 10 Android 2.2

June 5, 2015 | Proprietary and Confidential | - 5 -

IGATE
Speed.Agency.Imagination

Auto Focus

This attribute when applied to any form element, will result in the field receiving focus.

For e.g. consider the example shown below:

```
<form>
  <label for="firstname">First Name</label>
  <input type="text" id="firstname" name="firstname"
        autofocus>
  <label for="lastname">Last Name</label>
  <input type="text" id="lastname" name="lastname">
  <input type="submit" label="Go">
</form>
```

We have added the attribute autofocus to the firstname input field. When the form loads, you will find that the focus is already set on that field, thereby making it easier for the user to start filling the form.

9.1 : HTML5: Enhanced Form Elements

Required

- Required - A field with “required” attribute must be filled in with value before submission of a form
- Syntax is –

```
<input name="name" type="text" required/>
```

- The picture below shows us how Firefox and Opera prompt user to fill in value if a “Required field” is left blank upon submission

The screenshot displays two browser windows side-by-side. Both show a simple form with a text input field labeled "What is your favorite movie:" and a "Submit" button. In the Firefox 4 window, a red box-shadow is applied to the empty input field, and a tooltip-like message "Please fill out this field." appears below it. In the Opera 9 window, a similar red box-shadow is applied to the empty input field, and a tooltip-like message "This is a required field." appears below it. The browser logos and version numbers are visible at the bottom of each window.

Required:

If the **required** attribute is present, then the field must contain a value when the form is submitted. This informs the (HTML5-aware) web browser that the field is to be considered mandatory. Different browsers may mark the input box in some way (Firefox 4 Beta adds a red box-shadow by default), display a warning (Opera) or even prevent the form from being submitted if this field has no value. Hopefully these behaviours will converge in future releases.

Here's an example of an input field for a required email address that ensures that the field has a value and that the value is a valid email address.

Example:

```
<input type="email" id="email_addr" name="email_addr"
      required/>
```

9.1 : HTML5: Enhanced Form Elements

Pattern

- Pattern - A value filled in the field must be checked against the regular expression specified in pattern attribute.
- Syntax is –

```
Pincode: <input type="text" name="pin_code" pattern="[0-9]{6}" title="666666">
```

- The picture below shows us how browser prompt user to fill in valid value if a “field with Pattern attribute“ is filled with invalid value upon submission

A screenshot of a web browser showing an input field for a pincode. The field contains '773'. Below the field is a message: 'You must use this format: 666666'.

June 5, 2015

Proprietary and Confidential

- 7 -

IGATE
Speed. Agility. Imagination

Required:

If the **required** attribute is present, then the field must contain a value when the form is submitted. This informs the (HTML5-aware) web browser that the field is to be considered mandatory. Different browsers may mark the input box in some way (Firefox 4 Beta adds a red box-shadow by default), display a warning (Opera) or even prevent the form from being submitted if this field has no value. Hopefully these behaviours will converge in future releases.

Here's an example of an input field for a required email address that ensures that the field has a value and that the value is a valid email address.

Example:

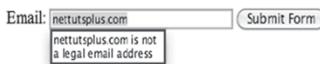
```
<input type="email" id="email_addr" name="email_addr"  
      required />
```

9.1 : HTML5: Enhanced Form Elements

Email

- Email - This field is used to check whether the string entered by the user is valid email id or not.
- Syntax is –

```
<input id="email" name="email" type="email" />
```
- Browser's that don't support this field will treat this as a simple text field
- This is how it looks like on form



The screenshot shows an Opera 10.63 browser window. A form has an email input field where "nettutsplus.com" is typed. Below the input field, a message box displays "nettutsplus.com is not a legal email address". To the right of the browser window, the Opera logo and version "10.63" are visible.

June 5, 2015 | Proprietary and Confidential | - 8 -

IGATE Speed. Agility. Imagination

Email:

The email type is used for input fields that should contain an email address. The value of the email field is automatically validated when the form is submitted.

Example

E-mail: `<input type="email" name="user_email" />`

Safari on the iPhone recognizes the email input type, and changes the on-screen keyboard to match it (adds @ and .com options).

9.1 : HTML5: Enhanced Form Elements

Data List

- Data List - Datalist is seems like type-ahead auto suggest textbox as you can see in Google search box
- Syntax is –

```
<input id="country_name" name="country_name"
      type="text" list="country" />
<datalist id="country">
  <option value="Australia">
  <option value="Austria">
  <option value="Algeria">
  <option value="Andorra">
  <option value="Angola">
</datalist>
```



Country : aus

Australia

Austria



User input

June 5, 2015 | Proprietary and Confidential | - 9 -

IGATE
Speed.Agency.Imagination

Data List:

The datalist element specifies a list of options for an input field.

The list is created with option elements inside the datalist. To bind a datalist to an input field, let the list attribute of the input field refer to the id of the datalist:

Example

Webpage: `<input type="url" list="url_list" name="link" />
<datalist id="url_list">
<option label="W3Schools" value="http://www.w3schools.com" />
<option label="Google" value="http://www.google.com" />
<option label="Microsoft" value="http://www.microsoft.com" />
</datalist>`

9.1: HTML5: Enhanced Form Elements

Search

- Search - In HTML5, we can define a textbox as search box instead of a normal textbox
 - Supported by Chrome 8 & Safari 5
 - Syntax is

```
<input id="mysearch" type="search" />
```



search : **Chrome**



search : **Safari**



Safari 5



Chrome 8

June 5, 2015
Proprietary and Confidential
- 10 -



IGATE
Speed.Agency.Imagination

Search:

Setting the type attribute of an `<input>` tag to **search** is mainly of cosmetic value. It doesn't automatically create a search field. As with all form elements, you still need to code the server-side logic yourself or use a third-party script. Most browsers simply display a search field as a normal text field. However, Safari and Chrome on Mac OS X automatically add rounded corners to match the operating system's standard look. Apple.com uses `<input type="search">` for their site-search box, to help give their Site a "Mac-like" feel. But there's nothing Mac-specific about it. It's just markup, so each browser on each platform can choose to render it according to platform-specific conventions. As with all the other new input types, browsers that don't recognize `type="search"` will treat it like `type="text"`.

Example:

```
<form>
<input name="q" type="search">
<input type="submit" value="Find">
</form>
```

9.1: HTML5: Enhanced Form Elements

Number

- Up and down button provided to increase and decrease the value.
- Min and max parameters provided to limit the values.
- Browser will treat it as simple textfield if it doesn't support this type.
- Syntax is

```
<input id="movie" type="number" value="0"/>
```

- Limiting the values for this field...

```
<input id="user_lic" type="number" min="5" max="30" step="5" value =""/>
```

Chrome

How often you watch movie in week :

Opera

How often you watch movie in week :

Chrome 8

Opera 11

IGATE

Speed. Agility. Imagination.

June 5, 2015

Proprietary and Confidential

- 11 -

Number:

Asking for a number is trickier than asking for an email address or web address.

First of all, numbers are more complicated than you might think.

You don't often

ask for “just a number.” It’s more likely that you’ll ask for a number in a

particular range. You may only want certain kinds of numbers within that range-

maybe whole numbers but not fractions or decimals.

Example:

```
<input type="number" min="0" max="10" step="2" value="6">
```

Let's take that one attribute at a time.

1. `type="number"` means that this is a number field.
 2. `min="0"` specifies the minimum acceptable value for this field.
 3. `max="10"` is the maximum acceptable value.
 4. `step="2"`, combined with the `min` value, defines the acceptable numbers in the range: 0, 2, 4, and so on, up to the maxvalue.
 5. `value="6"` is the default value.

That's the markup side of a number field. Keep in mind that all of those attributes are optional. If you have a minimum but no maximum, you can specify a min attribute but no max attribute. The default step value is 1, and you can omit the step attribute unless you need a different step value. If there's no default value, then the value attribute can be the empty string or even omitted altogether.

Visualizing Number field :

On the iPhone, where input is difficult to begin with, the browser once again optimizes the virtual keyboard for numeric input.

In the desktop version of Opera, the same type="number" field is rendered as a "spinbox" control, with little up and down arrows that you can click to change the value.

Opera respects the min, max, and step attributes, so you'll always end up with an acceptable numeric value. If you bump up the value to the maximum, the up arrow in the spinbox is greyed out.

9.1: HTML5: Enhanced Form Elements

Range

- Also known as slider
- Before HTML5, programmers used to write lines of code for range/slider
- Easy to implement
- Look and feel is different for different browser's
- Browser will treat it as simple textfield if it doesn't support this type
- Syntax

```
<input id="test" type="range"/>
```

The diagram shows six examples of the HTML5 `<input type="range">` element as rendered by different browsers:

- Chrome:** Shows a horizontal slider with a small circular handle.
- Safari:** Shows a horizontal slider with a small circular handle.
- Opera:** Shows a horizontal slider with a small circular handle and a visible numerical scale below it.
- Safari 5:** Shows a standard text input field.
- Chrome 8:** Shows a standard text input field.
- Opera 11:** Shows a standard text input field.

June 5, 2015
Proprietary and Confidential
- 13 -

Range:

Spinboxes are not the only way to represent numeric input.

You've probably also seen "slider" controls.

You can now have slider controls in your web forms, too.

Example:

```
<input type="range" min="0" max="10" step="2" value="6">
```

All the available attributes are the same as `type="number"` — `min`, `max`, `step`, `value` — and they mean the same thing. The only difference is the user interface. Instead of a field for typing, browsers are expected to render `type="range"` as a slider control. Safari, Chrome, and Opera all do this. (Sadly, the iPhone renders it as a simple text box. It doesn't even optimize its on-screen keyboard for numeric input.) All other browsers simply treat the field as `type="text"`.

9.1: HTML5: Enhanced Form Elements

Date

➤ Date

- Important and mostly used element
- Simple to implement
- Before HTML5, programmers used to write lines of javascript code for date picker
- Input type for date:- date, week, month, time, datetime (gives UTC time), datetime-local (local time)

➤ Syntax is

```
<input id="meeting" type="date" value="" />
```

Meeting Date : 2011-01-13

Opera 11

June 5, 2015
Proprietary and Confidential
- 14 -

Date:

HTML 4 did not include a date picker control. JavaScript frameworks have picked up the slack (Dojo, jQuery UI, YUI, Closure Library), but of course each of these solutions requires “buying into” the framework on which the date picker is built.

HTML5 finally defines a way to include a native date picker control without having to script it yourself. In fact, it defines six input types: date, month, week, time, date + time, and date + time - timezone.

So far, support is... sparse.

9.1 : HTML5: Enhanced Form Elements

Audio

- Audio - HTML5 is likely to put an end to audio plug-in such as Microsoft Windows Media player, Microsoft Silverlight , Apple QuickTime and the famous Adobe Flash
- MIME type's - *audio/mpeg*, is optional but its always better to provide
- Only .mp3, .wav, and .ogg (vorbis) formats are supported till date
- If quick time player is not available, then safari won't support this tag
- Other properties like auto play, loop, preload area also available
- Syntax is –

```
<audio controls>
<source src="vincent.mp3" type="audio/mpeg"/>
<source src="vincent.ogg" type="audio/ogg"/>
</audio>
```

June 5, 2015

Proprietary and Confidential

- 15 -

IGATE
Speed.Agency.Imagination

Audio & Video:

HTML5 is fairly intelligent about picking the right default for presenting the most optimum audio or video. Couple that with the absolute minimum coding that's needed to handle multimedia in HTML5 and you have a pretty good situation for developers.

The bad news is that because the people diligently working on the HTML5 specification tried to compromise between open formats and de facto standard formats and so on, support for native codecs in HTML5 is slightly lacking: there isn't any. It's up to the browser to support formats, and up to the developer to supply them. What's emerged from that are a few relatively new standards.

9.1 : HTML5: Enhanced Form Elements

Video

- Video – HTML5 video tag is exactly similar to audio but with few extra attributes
- Attributes
 - Width : Width of video area
 - Height : Height of video area
 - Poster : Still Image file projected on screen before video gets displayed
- Syntax is –

```
<video src="http://.....ogv" controls width="300" height="250"></video>
```

June 5, 2015Proprietary and Confidential- 16 -

Examples:

Audio

```
<audio controls preload="metadata">
<source src="sample.mp3">
<source src="sample.ogg"> </audio>
```

Video

```
<video controls width="640" height="480" src="sample.mp4"
       poster="sample.jpg">
</video>
```

Both the `<video>` and `<audio>` can take another important parameter, shown above, called `preload` (which was formerly implemented as `autobuffer` with slightly different syntax). The values for `preload` can be `auto` (download the media file to the browser in advance), `none` (do not preload the media), or `metadata` (just download enough metadata to discover the duration and other information of the media file). So with a little extra care in coding, you can make the user's experience much better.

9.1 : HTML5: Enhanced Form Elements

Article

➤ Article

- Represents independent component on page
- Separates itself from other elements on page
- Can be used most effectively in forum posts, blogs, magazine or newspaper etc

➤ Syntax is

```
<article>
<header>
<h1>The Very First Rule of Life</h1>
<p>Published : <time pubdate="pubdate">2009-10-09</time></p>
</header>
<p>If there's a microphone anywhere near you, assume it's hot
and sending whatever you're saying to the world. Seriously.</p>
<p>...</p>
<footer>
<a href="?comments=1">Show comments...</a>
</footer>
</article>
```

June 5, 2015

Proprietary and Confidential

- 17 -

IGATE
Speed. Agility. Imagination.

Article:

The article element represents a component of a page that consists of a self-contained composition in a document, page, application, or site and that is intended to be independently distributable or reusable, e.g. in syndication. This could be a forum post, a magazine or newspaper article, a Web log entry, a user-submitted comment, an interactive widget or gadget, or any other independent item of content.

9.1 : HTML5: Enhanced Form Elements

Demo

- Demonstration on the use of new form elements



June 5, 2015

Proprietary and Confidential

- i8 -

IGATE
Speed. Agility. Imagination.

9.2 : An HTML5 Detection Library

Modernizr

➤ **What's is Modernizr**

- It is (<http://www.modernizr.com>) an open source
- It's a JavaScript library that detects HTML5 & CSS3 features supported by Web Browser

➤ **How does this work?**

- Import JavaScript library in your HTML header

```
<script src="modernizr.min.js" type="text/javascript"></script>
```

```
if (Modernizr.audio)
/* properties for browsers that support audio */
else{
/* properties for browsers that does not support audio */
}
```

Automatically detects if audio is supported or not

June 5, 2015 | Proprietary and Confidential | - 19 -

IGATE
Speed.Aggility.Imagination

Modernizr

Modernizr is a JavaScript library that helps you add HTML5 capabilities to your websites. HTML5 is a combination of HTML, JavaScript, and CSS3. The problem with HTML5 is that not all browsers support HTML5 markup and those that do support it, do not always support it the same way. Modernizer helps you get a consistent experience for your end users across multiple browsers. Even older browsers like Internet Explorer 6 can benefit from Modernizr.

Modernizr detects the actual HTML5 features that a browser supports. It does this by first creating an element, setting a style instruction on the element, and then retrieving it. If a browser does not support the instruction, it will return an error or undefined.

The first thing you need to do is get the Modernizr library. There are two ways to do this. First, you can use a Content Delivery Network (CDN) like the one from Microsoft. The second (and best) way is to use a custom build from Modernizr.com. This option allows you to choose only the portions you want. This results in a smaller file.

Modernizr is referenced just like any JavaScript code. Modernizr should be placed after your CSS references.

Place a script reference in the head tag of your HTML document:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>Dive Into HTML5</title>
  <script src="modernizr.min.js"></script>
</head>
<body>
```

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

Modernizr runs automatically. There is no `modernizr_init()` function to call. When it runs, it creates a global object called Modernizr, that contains a set of Boolean properties for each feature it can detect. For example, if your browser supports the canvas API, the `Modernizr.canvas` property will be true. If your browser does not support the canvas API, the `Modernizr.canvas` property will be false.

```
if (Modernizr.canvas)
{ // let's draw some shapes! } else { // no native canvas support available :( }
```

9.3: Using Canvas in HTML5

Canvas

➤ Why do you need Canvas?

- A canvas is a rectangle in your web page within which you can use JavaScript to draw shapes
- Canvas can be used to represent something visually in your browser
 - Simple Diagrams
 - Fancy user interfaces
 - Animations
 - Charts and graphs
 - Embedded drawing applications
 - Working around CSS limitations

➤ Syntax is

```
<canvas id="myCanvas"></canvas>
<script type="text/javascript">
var canvas=document.getElementById('myCanvas');
var ctx=canvas.getContext('2d');
ctx.fillStyle='#FF0000';
ctx.fillRect(0,0,80,100);
</script>
```



HTML5 Training



IGATE
Speed. Agility. Imagination

June 5, 2015 | Proprietary and Confidential | - 21 -

Canvas

HTML5 defines the <canvas> element as “a resolution-dependent bitmap canvas that can be used for rendering graphs, game graphics, or other visual images on the fly.” A canvas is a rectangle in your page where you can use JavaScript to draw anything you want. HTML5 defines a set of functions (“the canvas API”) for drawing shapes, defining paths, creating gradients, and applying transformations. If your browser supports the canvas API, the DOM object it creates to represent a <canvas> element will have a getContext() method. If your browser doesn’t support the canvas API, the DOM object it creates for a <canvas> element will only have the set of common properties, but not anything canvas-specific.

```
function supports_canvas()
{ return !!document.createElement('canvas').getContext; }
```

This function starts by creating a dummy <canvas> element. But the element is never attached to your page, so no one will ever see it. It’s just floating in memory, going nowhere and doing nothing.

As soon as you create the dummy <canvas> element, you test for the presence of a getContext() method. This method will only exist if your browser supports the canvas API. This function will detect support for most of the canvas API, including shapes, paths, gradients & patterns. It will not detect the third-party explorercanvas library that implements the canvas API in Microsoft Internet Explorer.

Instead of writing this function yourself, you can use Modernizr to detect support for the canvas API as seen earlier.

Summary

➤ In this lesson, you have learnt about:

- HTML5 introduces new and enhanced form elements those who caters to every need of modern Web Designing
- It also introduces a new JavaScript library, Modernizr, that helps you detect HTML5 feature capabilities of your websites



Review Question

- Question 1: A _____ is a textbox that hold a text in lighter shade when there is no value and not focused
- Question 2: A _____ is a JavaScript library that helps you add HTML5 capabilities to your websites





Web Basics – HTML5

Lab Book

Copyright © 2011 IGATE Corporation. All rights reserved. No part of this publication shall be reproduced in any way, including but not limited to photocopy, photographic, magnetic, or other record, without the prior written permission of IGATE Corporation.

IGATE Corporation considers information included in this document to be Confidential and Proprietary.

Document Revision History

Date	Revision No.	Author	Summary of Changes
2-Jun-05	1.0	Kumar B	Initial Draft
1-Oct-09	1.0	Ummeaiman Diwanji	Quality review. Transferred to the new template.
1-Oct-09	1.0	Anu Mitra	Final review
16/06/10	2	Anu Mitra	Refinements
27/05/11	3	Anu Mitra	Integration Refinements
21-Apr-2015	4	Rathnajothi P	Revamp/Refinement as per revised TOC

Table of Contents

DOCUMENT REVISION HISTORY.....	2
TABLE OF CONTENTS	3
GETTING STARTED	4
Overview	4
Setup Checklist for HTML5	4
Instructions	4
Learning More (Bibliography)	4
LAB 1: HTML BASICS	5
1.1: Create HTML Page	5
1.2 Example: MyFirstPage.html	6
1.3 Example: Headers.html	6
1.4 Example: Address.html	7
1.5 Example: PreFormattedText.html	8
LAB 2: CREATING TABLES.....	10
Problem 1: Fun with Food	10
Problem 2: Table Heading << To Do>>	11
Problem 3: Calendar <<To Do>>	12
LAB 3: WORKING WITH LISTS	13
Problem 1: Types of Lists	13
LAB 4: WORKING WITH LINKS	16
Problem 1: Welcome to Big Company	16
Problem 2: Employee Details <<To Do>>	19
LAB 5: IMAGE HANDLING.....	20
LAB 6: WORKING WITH FRAMES.....	23
LAB 7: HTML FORMS FOR USER INPUT	26
LAB 8: NEW FORM ELEMENTS.....	29
Appendix A: HTML Standards	34
Appendix B: Table of Figures	37
Appendix C: Table of Examples	38

Getting Started

Overview

This lab book is a guided tour for learning HTML. It comprises solved examples and 'To Do' assignments. Follow the steps provided in the solved examples and work out the 'To Do' assignments given.

Setup Checklist for HTML5

Here is what is expected on your machine in order for the lab to work.

Minimum System Requirements

- Intel Pentium 90 or higher (P166 recommended)
- Microsoft Windows 95, 98, or NT 4.0, 2k, XP.
- Memory: 32MB of RAM (64MB or more recommended)
- Internet Explorer 6.0 or higher

Please ensure that the following is done:

- A editor like Notepad, Eclipse, Visual Studio 2008 is installed.

Instructions

- For all coding standards refer Appendix A. All lab assignments should refer coding standards.
- Create a directory by your name in drive <drive>. In this directory, create a subdirectory html_assgn. For each lab exercise create a directory as lab <lab number>.
- You may also look up the on-line help provided in the MSDN library.
- The faculty will introduce you to the editor to be used.

Learning More (Bibliography)

- HTML Source Book by Ian S. Graham
- HTML: Complete Concepts and Techniques by Gary B. Shelly
- HTML: The Definitive Guide by Chuck Musciano
- Dynamic HTML: The Definitive Reference by Danny Goodman
- HTML: The Complete Reference by Thomas A. Powell

Lab 1: HTML Basics

Goals	<ul style="list-style-type: none"> Understand the process of creating an HTML page and viewing it in a browser window. Learn to apply physical or logical character effects. Learn to manage document spacing
Time	45 minutes

1.1: Create HTML Page

Create a web page to display the text 'This is the first html page created'.

Solution:

Step 1: Click the **Start** button. On the **Programs** menu, navigate to the **Accessories** submenu. Click **Notepad**.

Step 2: Write the below HTML program in Notepad.

```
<!DOCTYPE html>
<html>
<head>
    <title>This is the first html page</title>
</head>
<body>
    This is the first html page created
</body>
</html>
```

Step 3: Save the file with extension **.html**. Save it in the **lab1** directory as **firstpage.html**.

Step 4: From Internet Explorer, on the **File** menu, click **Open**. **Open** dialog box appears. Click **Browse** to select the file you have just saved. Refer to the figure that follows.

Step 5: Once you have selected the file, click **OK** in the **Open** dialog box. Output appears as shown in the figure that follows.

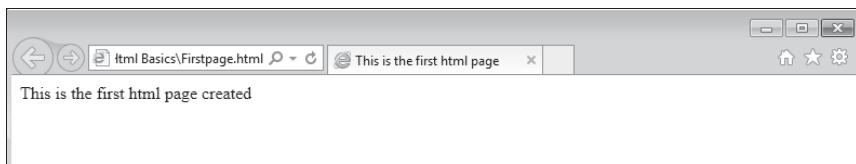


Figure 1: First.html in a browser

1.2 Example: MyFirstPage.html

```
<!DOCTYPE html>
<html>
<head>
<title>My First Page</title>
<META [http-equiv] [contents=n]>
<meta http-equiv=refresh content=60>
-will refresh the current document after every 60 seconds.

<meta http-equiv=refresh content="20;url=c:/html/html34.htm">
-will load sefcid file after 20 seconds.
<base href="c:/mydir/html/">
!-- you to use shortcuts in your URLs if you must reference several files from the same location.-->
</head>
<body> Hello World!! </body>
</html>
```

Example 1: MyFirstPage.html

Output of the above HTML code is:

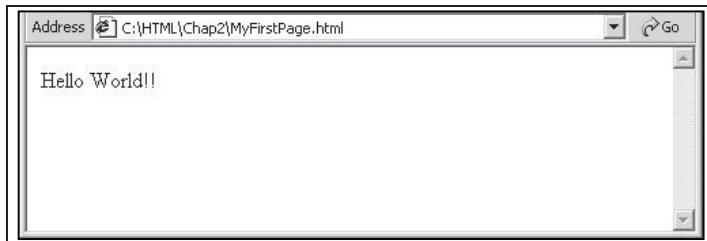


Figure 2: MyFirstPage.html Output

1.3 Example: Headers.html

```
<!DOCTYPE html>
<html>
<head><title>This is the first html page</title>
<body>This is the first html page created
<h1>This is level 1 heading</h1>
<h2>This is level 2 heading</h2>
<h3>This is level 3 heading</h3>
</body>
</head></html>
```

Example 2: Headers.html

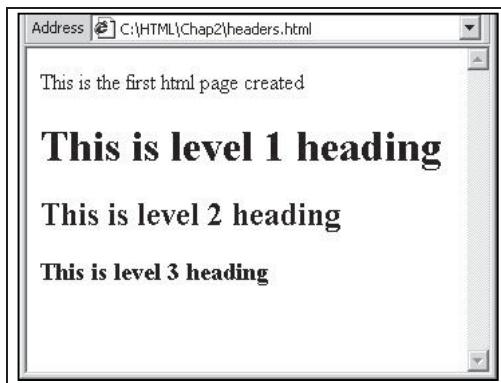


Figure 3: Headers.html Output

1.4 Example: Address.html

```
<html>
<head>
<title>Address Example</title>
</head>
<body>
<font size="2">Your address:</font><br>
Abc Xyz<br>
<address>b/102 royal palms,</address>
<address>off. s. v. road,</address>
<address>Andheri-West,</address><address>Mumbai.</address>
</body>
</html>
```

Example 3: Address.html

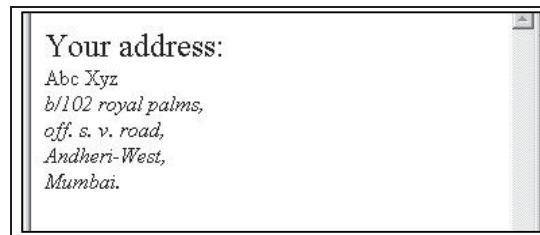


Figure 4: Address.html Output

1.5 Example: PreFormattedText.html

```
<!DOCTYPE html>
<html>
<head><title>
PREFORMATTED TEXT EXAMPLE
</title></head>
<body>
<h3>GROSS SALE WITH PREFORMATTING</h3>
<hr>
<pre>
    <b>GROSS SALES</b>
SALESMAN          SALES      RANKING
TIM                $10,000    2<BR>
TOM                $ 5,000    5<BR>
TAMMY              $20,000    1<BR>
Each line has a carriage return after it.
</pre>
<h3>GROSS SALE WITHOUT PREFORMATTING</h3>
<p><hr>
    <b>GROSS SALE</b>
SALESMAN          SALES      RANKING
TIM                $10,000    2<BR>
TOM                $ 5,000    5<BR>
TAMMY              $20,000    1<BR>
</body>
```

Example 4: PreFormattedText.html

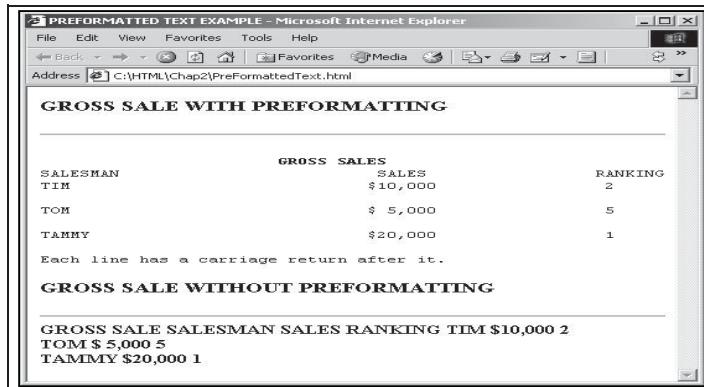


Figure 5: PreFormattedText.html Output

Problem 1: Resume Creation <<To Do>>

Problem Statement:

Create your resume page as per the format shown in the figure that follows.

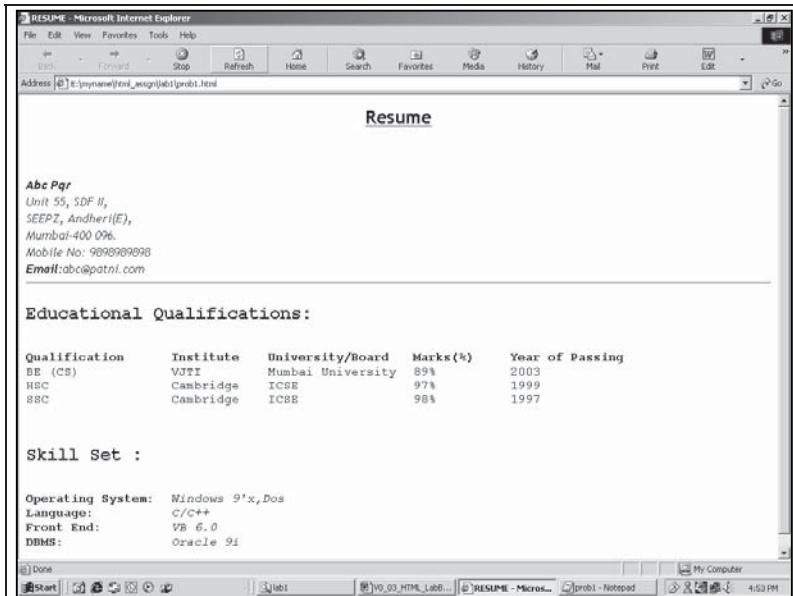


Figure 6: Resume Page

Solution:

1. Open **Editor**. Type the code and save the file.
2. Use Heading 2 for the headings “Educational Qualifications” and “Skill Set”.
3. Use font size 3 for data pertaining to educational qualifications and skill set.
4. Display details against categories under Skill Set in *italics*.
5. Start the Internet Explorer. On the **File** menu, click **Open**. **File** open dialog box appears. Click the **Browse** button and select **prob2.html file**.
6. Check if the output is as per the requirement.

Lab 2: Creating Tables

Goals	At the end of this lab session you will understand: <ul style="list-style-type: none"> • Attributes of a Table • Table Headers • Table Data • Table Formatting • Control Table Borders • Grouping of Columns
Time	90 minutes

Problem 1: Fun with Food

Problem Statement:

Create a web page, which uses a table with columns *Fruit*, *Color* and *Cost per pound* as shown in the figure that follows.

Fun with food		
Fruit	Color	Cost per pound
Grapes	Purple	1.25
Cherries	Red	154.79
Kiwi	Brown	10.00
This is the footer area		

Figure 7: Fruits Table

Solution

Step 1: Write the following code in **Notepad** and save the file.

```
<!DOCTYPE HTML>
<HTML>
<HEAD>
  <TITLE>Fruits Table</TITLE>
</HEAD>
<BODY>
<TABLE border="1" >

<CAPTION>Fun with food</CAPTION>
<COLGROUP>
  <COL>
</COLGROUP>
<COLGROUP>
```

```
<COL align="center">
<COL>
</COLGROUP>
<THEAD>
<TR>
    <TH style="background-color:yellow">Fruit</TH>
    <TH style="background-color:yellow">Color</TH>
    <TH style="background-color:yellow">Cost per pound</TH>
</TR>
</THEAD>
<TBODY>
<TR>
    <TD>Grapes</TD>
    <TD>Purple</TD>
    <TD>1.25</TD>
</TR>
<TR>
    <TD>Cherries</TD>
    <TD>Red</TD>
    <TD>154.79</TD>
</TR>
<TR>
    <TD>Kiwi</TD>
    <TD>Brown</TD>
    <TD>10.00</TD>
</TR>
</TBODY>
<TFooter>
<TR>
    <TH colspan="3">This is the footer area</TH>
</TR>
</TFooter>
</TABLE>
</BODY>
</HTML>
```

Example 5: Fruit Table

Step 2: Open the file page in the browser to check the required output.

Problem 2: Table Heading << To Do>>

Problem Statement: Create a html page. When this page is opened in a browser, it should appear as shown in the following figure

Product Table

Product	Price	Quantity	Amount
P001	1000.00	12	12000.00
P002	2000.00	10	20000.00
Total	3000.00	22	32000.00

Figure 8: Product table

Note: Table heading - Background color is : navy and font color is : white.

Solution

1. Open **Editor**. Type the code and save the file.
2. Open the page in browser
3. Check the page shown in the browser and verify that it is as per the requirement.

Problem 3: Calendar <>To Do>>

Problem Statement:

Design a web page to display a calendar for a month using html table.

January						
Sun	Mon	Tue	Wed	Thu	Fri	Sat
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

Figure 9: Calendar

Note: Background colors to be used: For all the Sundays: green, for all the Saturdays: aqua, for 1, 26 Jan: yellow

Solution

1. Open **Notepad**. Type the code and save the file.
2. Open the page in browser.
3. Verify that the output is as per requirements.

Lab 3: Working with Lists

Goals	At the end of this lab session you will be able to use following types of lists: <ul style="list-style-type: none"> • Numbered List • Bulleted List • Directory List • Glossary List
Time	30 minutes

Problem 1: Types of Lists

Problem Statement:

Design a web page as shown below

The following is a demonstration of various list styles

- An ordered list i.e. using OL and LI
- 3. Seepz
 - 10. TTC
 - 11. MIDC

Unordered list i.e. using UL and LI

- Seepz
- TTC
- MIDC

Glossary List - List of locations

Seepz	Andheri
TTC	New Bombay
MIDC	Andheri

Figure 10: List

Solution

Step 1: Write the following code in **NotePad** and save.

```
<!DOCTYPE html>
<html>
<head>
<title>Working with Lists</title>
</head>
<body>
    <p>The following is a demonstration of various list styles
    <ol start="3">
        An ordered list i.e. using OL and LI
        <li>Seepz
        <li value="10">TTC
        <li>MIDC
    </ol>
    <hr>
    <ul style="list-style-type: square">
        Unordered list i.e. using UL and LI
        <li>Seepz
        <li>TTC
        <li>MIDC
    </ul>
    <hr>
    Glossary List - List of locations
    <dl>
        <dt>Seepz
        <dd>Andheri
        <dt>TTC
        <dd>New Bombay
        <dt>MIDC
        <dd>Andheri
    </dl>
</body>
</html>
```

Example 6: Types of Lists

Step 2: Check the page shown in the browser and verify that it is as per the requirement.

Problem 2: Subjects <<To Do>>

Create a web page to display a list as shown in the figure that follows.

Display the list of Subjects using List	
1.	Language
i.	English
A.	Prose
B.	Poetry
ii.	Hindi
A.	Prose
B.	Poetry
iii.	Marathi
A.	Prose
B.	Poetry
2.	Social Study
i.	History
ii.	Geography
3.	Science
i.	Physics
A.	Part1
B.	Part2
ii.	Chemistry
A.	Organic
B.	Inorganic
iii.	Biology
A.	Botany
B.	Zoology
4.	Maths
i.	Algebra
ii.	Geometry

Figure 11: Subject list

Solution

1. Open **Notepad**. Type the code and save the file.
2. Open the page in browser.
3. Check the page shown in the browser and verify that it is as per the requirement.

Lab 4: Working with Links

Goals	At the end of this lab session you will be able to: <ul style="list-style-type: none">• Create links to web documents.• Create links to email.• Create hyperlinks for lists and table data.• Provide target for hyperlink.
Time	30 minutes

Problem 1: Welcome to Big Company

Problem Statement:

Design a simple home page for a company with a heading and 3 links – About, Products, Contact as given in the figure below.

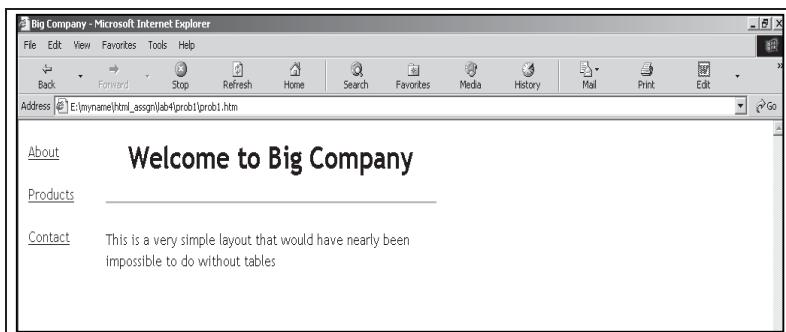


Figure 12: Big Company home page

When you click the “About” hyperlink, following page should be displayed.

The Big company was founded in 1956.

Figure 13: About

When you click the **Back** button on the browser toolbar, they should be redirected to the page prob1.html. Click the “Products” hyperlink to reach the following page:

The following are the products offered:

- Personal Health
- Beverages
- Garments
- Books

Figure 14: Products

When you click the **Back** button on the browser toolbar, they are redirected to page prob1.html. Click the “Contact” hyperlink. It opens Outlook Express and the e-mail address given in the To field, which is edude@patni.com in the following illustration, is displayed in the New message window. This email address is specified in the *mailto* attribute.

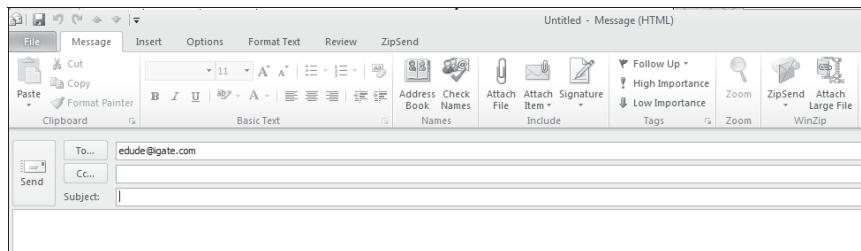


Figure 15: Contact

Solution

Step 1: Write the following code in **Notepad** and save the file.

```
<!DOCTYPE html>
<HTML>
<HEAD>
    <TITLE>Big Company</TITLE>
</HEAD>
<BODY>
    The Big company was founded in 1956.
</BODY>
</HTML>
```

Example 7: Big Company

Step 2: Write the following code in **Notepad** and save it as the file.

```
<!DOCTYPE html>
<HTML>
<HEAD>
    <TITLE>Products</TITLE>
</HEAD>
<BODY>
    The following are the products offered :

    <UL>
        <LI>Personal Health
        <LI>Beverages
        <LI>Garments
        <LI>Books
    </UL>
</BODY>
</HTML>
```

Example 8: Products

Step 3: Write the following code in **Notepad** and save the file.

```
<!DOCTYPE html>
<HTML>
<BODY>
<TABLE>
<TR>
    <TD>
        <A href="about.htm">About</A><BR><BR>
        <A href="products.htm">Products</A><BR><BR>
        <A href="mailto:edude@igate.com">Contact</A><BR><BR>
    </TD>
    <TD>
        <H1>Welcome to Big Company</H1>
        <HR>
        <P>This is a very simple layout that would have nearly been impossible to do without tables</P>
    </TD>
<TR><TD><BR></TD>
</TABLE>
</BODY>
</HTML>
```

Example 9: Welcome to Big Company

Step 4: Start the Internet Explorer. On the **File** menu, click **Open**. **Open** dialog box appears. Click the **Browse** button and open the page with links file. Verify if the links on the page are working as per the requirement.

Problem 2: Employee Details <>To Do>>**Problem Statement:**

Design a simple home page for a company to display employee details as given below.

Empcode	Emp Name	Dept Code	Experience
1001	Kiran Rao	10	8 Yrs.
1002	Aamir Khan	20	5 Yrs.
1003	Ishita Shah	30	10 Yrs.

Figure 16: EmployeeDetails

When you click the department code “10” hyperlink, page with following content should be displayed.

This is Sales department located at Mumbai...

Figure 17: Sales Department

When you click the department code “20” hyperlink, page with following content should be displayed.

This is training department located at Pune...

Figure 18: Training Department

When you click the department code “30” hyperlink, page with following content should be displayed.

This is accounts department located at Chennai...

Figure 19: Accounts Department

Lab 5: Image Handling

Goals	At the end of this lab session you will be able to: <ul style="list-style-type: none"> • Understand the use of inline images. • Attributes of an inline image. • Text and image aligning. • Use of an image as a hyperlink.
Time	30 minutes

Problem 1: Images with Clickable Areas <<To do>>

Problem Statement:

Create a web page with an image map of a picture (*bike.gif*) with four clickable rectangles. Four clickable areas for the bike image are *engine*, *seat*, *fork* and *headlight*. When you click within a rectangular area it needs to display information about selected part.

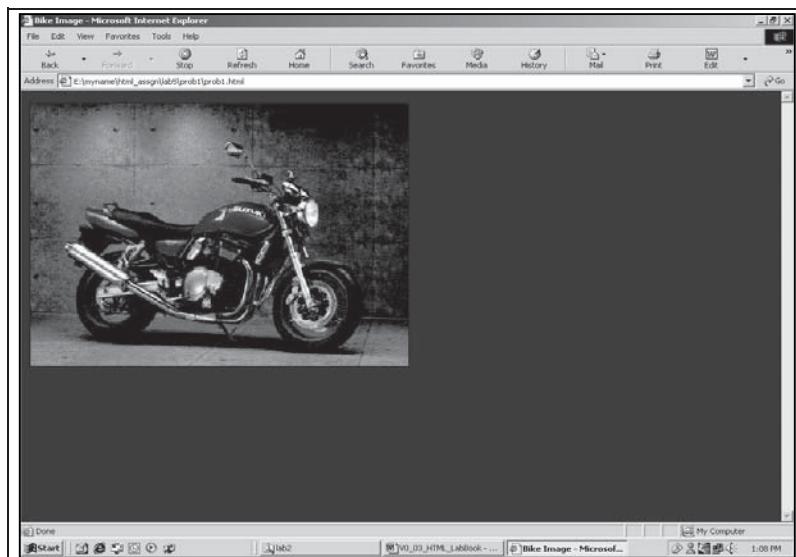


Figure 20 : Imagemap

When you click the engine portion of the image, it should produce the following output:

This is the engine of the bike. It is a twin cylinder, 16 valve, turbo charged liquid cooled specimen. Truly state of the art.

Figure 21: Engine

Solution

Step 1: Write the following code in **Notepad** and save the file.

```
<!DOCTYPE html>
<HTML>
<HEAD>
    <TITLE>Description Page</TITLE>
</HEAD>
<BODY>
    <H2>Description of the various sections of the bike</H2>
    <TABLE>
        <TR>
            <TD><A id="engine"></A>
                <B> This is the engine of the bike. It is a twin cylinder, 16 valve, turbo charged liquid cooled specimen. Truly state of the art.</B>
            </TD></TR>
        <TR>
            <TD><A id="seat"></A>
                <B> This is the seat of the bike. Designed to seat two comfortably and upholstered in calf leather.</B>
            </TD></TR>
        <TR>
            <TD><A id = "fork"></A>
                <B> This is the front wheel section of the bike. The air-suspension forks are designed to absorb even the harshest of shocks while riding. The disc-brakes on the front-wheel facilitate immediate braking even at break-neck speeds.</B>
            </TD></TR>
        <TR>
            <TD><A id = "head"></A>
                <B> This is the head section of the bike. The front panel is carefully designed to provide easy access to the various functions of the bike. The headlight is powered by 16 volt batteries to provide clear road vision even in pitch darkness.</B>
            </TD></TR>
        </TABLE>
    </BODY></HTML>
```

Example 10: Image Maps

Step 2: Write the following code in **NotePad** and save the file

```
<HTML>
<HEAD>
    <TITLE>Bike Image</TITLE>
</HEAD>
<BODY BGCOLOR = "maroon">
<MAP NAME = "bikemap">
    <AREA SHAPE = "RECT" COORDS = "198,228,246,273" HREF = "desc.htm#engine">
    <AREA SHAPE = "RECT" COORDS = "105,140,219,193" HREF = "desc.htm#seat">
    <AREA SHAPE = "RECT" COORDS = "341,221,396,284" HREF = "desc.htm#fork">
    <AREA SHAPE = "RECT" COORDS = "334,130,377,171" HREF = "desc.htm#head">
</MAP>
<IMG src = "bike.gif" ALT = "Imgmap" USEMAP = "#bikemap">
</BODY>
</HTML>
```

Example 11: Bike Image

Lab 6: Working with Frames

Goals	At the end of this lab session you will be able to: <ul style="list-style-type: none"> Understand the need for frames in web pages. Create and work with frames. Manage large content with frame.
Time	30 minutes

Problem 1: Frames

Problem Statement:

Create a web page which allows you to click on 2 hyperlinks courses, menu in the frame on the left. When you click a link, the details are displayed in the frame on the right. The file layout.html is loaded in the frame on the left.

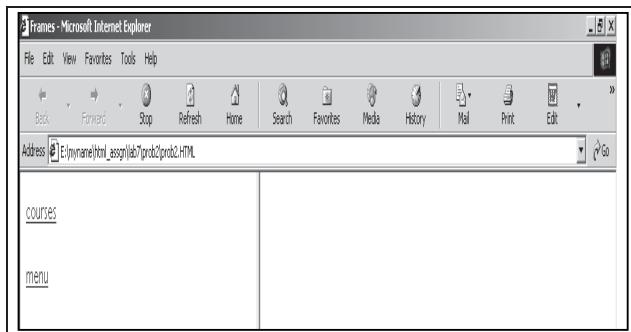


Figure 22: Frames

When you click the link “courses”, the details are displayed in the frame on the right.

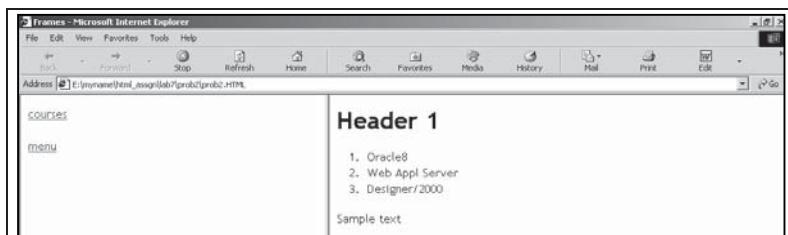


Figure 23: Courses

When you click the link “menu”, the details are displayed in the frame on the right

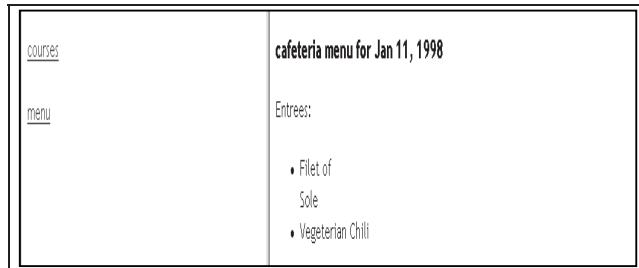


Figure 24: Menu

Solution

Step 1: Write the following code in **Notepad** and save the file.

```
<!DOCTYPE html>
<html>
<head>
<title>Frames</title>
</head>
<body>
<iframe src="content.html" width="100"></iframe>
<iframe name="side-2" width="1000"></iframe>
</body>
</html>
```

Example 12: Frames (1)

Step 2: Write the following code in **Notepad** and save it as *lab6\prob1\layout.html*.

```
<!DOCTYPE html>
<html>
<body>
<a href="courses.html" target="side-2"> courses </a><br><br>
<a href="menu.html" target="side-2"> menu </a>
</body>
</html>
```

Example 13: Frames (2)

Step 3: Write the following code in **Notepad** and save.

```
<html>
<head>
    <title>Sample Page</title>
</head>
<body>
<h1>Header 1</h1>
<ol>
    <li>Oracle8</li>
    <li>Web Appl Server</li>
    <li>Designer/2000</li>
</ol>

Sample text
</body>
</html>
```

Example 14: Courses

Step 4: Write the following code in **NotePad** and save.

```
<html>
<head>
<title>Cafeteria Menu Application</title>
</head>
<body>
<h3>cafeteria menu for Jan 11, 1998</h3>
<p>
Entrees:<br>
<ul>
<li>Fillet of <br>Sole
<li>Vegetarian Chili
</ul>
</body>
</html>
```

Example 15: Menu

Step 5: Open the file prob1\layout.html in the browser and check if the page works as per the requirement.

Lab 7: HTML Forms for User Input

Goals	At the end of this lab session you will be able to: <ul style="list-style-type: none"> • Understand the role of forms in web pages. • Understand various HTML elements used in forms. • Develop HTML forms in web pages.
Time	45 minutes

Problem 1: Form

Problem Statement:

Design a web page *prob1.html* in the directory *lab7*. When *prob1.html* is opened in the browser, the page is displayed as shown in the figure that follows.

The form contains the following elements:

- An input field labeled "Enter the password" with three placeholder dots.
- An input field labeled "Enter your surname" containing the value "Sukuru".
- A text area labeled "Address :" with the placeholder "Enter ur address".
- A section titled "Select the training programs attended :" with three checkboxes:
 - COBOL
 - IDMS
 - Java
- A section titled "Select the group you belong to :" with four radio buttons:
 -
 - Group 2
 -
 - Others
- A dropdown menu titled "Which training program would you like to attend ?" with the value "JavaScript".
- A file upload input field with a "Browse..." button.
- Three buttons at the bottom: "Exit", "Save", and "Reset".

Figure 25: Forms

Solution

Step 1: Write the following code in **Notepad** and save it as *lab6\prob1.html*.

```
<!DOCTYPE html>
<HTML>
<HEAD>
<TITLE>Form Methods</TITLE>
</HEAD>
<BODY bgcolor="skyblue">
<FORM action="mailto:edude@igate.com" name="ab" method="post"
enctype="multipart/form-data">
<P>
<LABEL>Enter the password
<INPUT type="password" name="USERNAME" size="20" value="abc" tabindex="3"></LABEL>
<INPUT type="hidden" name="coname" value="PCS"> Enter your surname:
<INPUT type="text" name="surname" SIZE="20" readonly value="Sukuru" tabindex="2"
maxlength="30"> <BR> <BR>
Address :
<TEXTAREA name="addr" Rows="5" cols="40" tabindex="0" accesskey="A">Enter ur address
</TEXTAREA>
<BR> <BR> Select the training programs attended : <BR>
<INPUT type="checkbox" name="s-cobol"> COBOL
<INPUT type="checkbox" name="s-idms" checked> IDMS
<INPUT type="checkbox" name="s-java"> Java <BR> <BR>
Select the group you belong to :
<INPUT type="radio" name="s-grp" value="grp1"> Group 1
<INPUT type="radio" name="s-grp" value="grp2" checked> Group 2
<INPUT type="radio" name="s-grp" value="grp3"> Group 3
<INPUT type="radio" name="s-grp" value="oth"> Others <BR> <BR>
Which training program would you like to attend ? <SELECT Name="pref">
<OPTION value="JS">JavaScript
<OPTION value="CORBA">CORBA
<OPTION value="VB6">Visual Basic 6
</SELECT>
</P>
<INPUT type="file" name="fnm"> <BR><BR>
<INPUT type="button" name="but" value="Exit">
<INPUT type="Submit" Value="Save" name="s-but">
<INPUT type="reset" Value="Reset">
</FORM>
</BODY>
</HTML>
```

Example 16: Forms

Step 2: Open *prob1.html* in the browser and verify if the form is displayed as per the requirement.

Problem 2: Employee Details <<To Do>>**Problem Statement:**

Design a web page *prob2.html* to accept the following employee details:

- Employee Name (Max 20 characters).
- Employee Code (Max 4 characters).
- Department (Use radio buttons).
- Date of Join (Use the format dd/mm/yyyy).
- Address.
- Training programs attended (Use check boxes).
- Training programs need to attend (Use select box).
- Send the information at empinfo@igate.com.

Name	<input type="text"/>		
Employee Code	<input type="text"/>		
Department	<input type="radio"/> Admin <input type="radio"/> HR <input type="radio"/> Technical <input type="radio"/> Accounts		
date of Joining	1 <input type="button" value="▼"/>	01 <input type="button" value="▼"/>	2014 <input type="button" value="▼"/>
Address	<div style="border: 1px solid #ccc; padding: 5px; height: 60px; width: 100%;"></div>		
Training Program attended	<input type="checkbox"/> HTML/DHTML <input type="checkbox"/> Java <input type="checkbox"/> Client/Server <input type="checkbox"/> .NET		
Training program need to attend	<input type="select" value="JavaScript"/>		
Send the information at	<input type="text" value="empinfo@igate.com"/>		

Figure 26: Employee Details

Solution

1. Open **Editor**. Type the code and save the file as lab7\prob2.html.
2. Open the page in the browser.
3. Verify if the output is as per the figure.

Lab 8: New Form Elements

Goals	At the end of this lab session, you will be able to: <ul style="list-style-type: none"> • Develop web pages using HTML5 enhanced form elements
Time	120 minutes

Problem 1: Form

Problem Statement:

Design a web page `prob1.html` in the directory `lab8`. When `prob1.html` is opened in the browser, the page is displayed as shown in the below figure

Solution

Step 1: Write the following code in **Notepad** and save it as `lab8\prob1.html`.

```
<!DOCTYPE html>
<html>
<head>
  <meta name="viewport" content="height=device-height,width=device-width,user-scalable=no" />
  <meta charset="UTF-8">
  <title>New Form Elements</title>
</head>
<body>
  <form name="Formelements" action="index2.jsp">
    <table>
      <tr>
        <td><label for="demo">Placeholder : </label></td>
        <td><input id ="demo" name="demo" placeholder="Enter Numbers Only" /></td>
      </tr>
      <tr>
        <td><label for="nameauto">Autofocus : </label></td>
        <td><input id ="nameauto" name="nameauto" type="text" autofocus/></td>
      </tr>
      <tr>
        <td><label for="range">Range : </label></td>
        <td><input id="range" name="range" type="range" min="0" max="50" value="10" /></td>
      </tr>
      <tr>
        <td><label for="search">Search : </label></td>
        <td><input id="search" name="search" type="search" placeholder="Search..." /></td>
      </tr>
    </table>
  </form>
</body>
</html>
```

```
<tr>
    <td><label for="date">Date : </label></td>
    <td>
<input id="date" name="date" type="date" min="2010-08-14" max="2014-08-14" value="" /></td>
</tr>

<tr>
    <td><label for="date">Week : </label></td>
    <td><input id="date" name="date" type="week" value="" /></td>
</tr>

<tr>
    <td><label for="date">Month : </label></td>
    <td><input id="date" name="date" type="month" value="" /></td>
</tr>

<tr>
    <td><label for="date">Time : </label></td>
    <td><input id="date" name="date" type="time" value="" /></td>
</tr>

<tr>
    <td><label for="number">Number : </label></td>
    <td>
<input id="number" name="number" type="number" step="1" min="-5" max="10" value="0" />
<td>
</tr>

<tr>
    <td><label for="required">Required : </label></td>
    <td><input id="required" name="user" type="text" required /></td>
</tr>

<tr>
    <td><label for="email">Email:</label></td>
    <td><input id="email" name="email" type="email" required /></td>
</tr>
<tr>
    <td><label for="color">Color : </label></td>
    <td><input id="color" name="color" type="color" placeholder="e.g. #bbbbbb" /></td>
</tr>
<tr>
    <td><label for="country_name">Country : </label></td>
    <td><input id="country_name" name="country_name" type="text" list="country" /></td>
```

```

<td><datalist id="country">
    <option value="Afghanistan">
    <option value="Albania">
    <option value="Algeria">
    <option value="Andorra">
    <option value="Angola">
    <option value="Caraibes">
    <option value="Cat">
    <option value="City">
    <option value="Cup">
    <option value="Clip">
</datalist></td>
</tr>

<tr><td colspan="2"><audio controls>
    <source src= "god.mp3" type="audio/mpeg" />
    <source src= "ivso_JuniorGroove.ogg" type="audio/ogg"/>
</audio></td></tr>
<!-- audio code works on Firefox and opera .ogg format only      --&gt;

&lt;tr&gt;
    &lt;td colspan="2"&gt;&lt;video
src="http://upload.wikimedia.org/wikipedia/commons/7/79/Big_Buck_Bunny_small.ogv" controls
width="300" height="250"&gt;
&lt;/video&gt;&lt;/td&gt;&lt;/tr&gt;
<!-- Video code works only on Firefox .ogg format. various ogg file extension are
.ogx, .ogv, .oga, .spx.      --&gt;
&lt;/table&gt;
&lt;table&gt;
&lt;tr&gt;
    &lt;td align="right"&gt;&lt;button type="submit" name="submit"
value="Submit"&gt;Submit&lt;/button&gt;&lt;/td&gt;
    &lt;td align="left"&gt;&lt;button type="reset" name="reset" value="reset"&gt;Reset&lt;/button&gt;&lt;/td&gt;
&lt;/tr&gt;
&lt;/table&gt;
&lt;/form&gt;
&lt;/body&gt;
&lt;/html&gt;
</pre>

```

Example 17: Code for New Form Elements

Placeholder :

Autofocus :

Range :

Search :

Date :

Week :

Month :

Time :

Number :

Required :

Email:

Color :

Country :

 0:00



 0:00

Figure 27: New Form Elements

Problem 2: Candidate Details <<To Do>>

Problem Statement:

Design a web page StudentInfoForm.html to accept the following student details:

1. Name (Accept only characters , Max 15 characters)

2. Password (Max 15 characters)
3. Phone number(Accept 10 digits)
4. Gender (Make use of radio button)
5. Date of Birth (Make use of date field and date of birth should not be greater than current date)
6. Email (Accept valid Email)
7. Highest Qualification (Make use of datalist to populate data like B.Tech, M.Tech, MBA, MCA, MSc, MA, BSC..)
8. Courses interested in (Make use of check box)
9. Comments to mention regarding Degree / External Certificates (Make use of textarea)
10. Uploading Degree / External certificates (Make use of file input type)
11. Use Placeholders to describe the type of input.
12. All fields marked (*) are mandatory

Candidate Information

Name: *	<input type="text"/>
Password: *	<input type="password"/>
Phone number: *	<input type="text"/>
Gender: *	<input type="radio"/> Male <input type="radio"/> Female
Date of Birth: *	<input type="date"/>
Email: *	<input type="text"/>
Highest Qualification: *	<input type="button" value="Select Highest Qualification"/>
Courses Interested in: *	<input type="checkbox"/> Java <input type="checkbox"/> HTML 5 <input type="checkbox"/> CSS 3 <input type="checkbox"/> Angular JS <input type="checkbox"/> JQuery
Comments: (Mention External Certifications if any)	<input type="text"/>
Upload Degree / External Certificates: *	<input type="button" value="Choose Files"/> <input type="text" value="No file chosen"/>
	<input type="button" value="Submit Information"/> <input type="button" value="Clear"/>

All fields marked (*) are mandatory

Solution

1. Open **Editor**. Type the code and save the file as lab7\prob2.html.
2. Open the page in the browser.
3. Verify if the output is as per the figure.

Appendix A: HTML Standards

Key Things To Keep In Mind:

- HTML standards help you reach the widest possible audience.
- There are many technologies that are *associated* with HTML because they are used on a Web page or in conjunction with HTML. But these technologies are not HTML:
 - CGI (Common Gateway Interface)
 - Java
 - JavaScript (JavaScript is also not Java)
 - Dynamic HTML (DHTML)
 - XML (Extensible Markup Language)
 - A variety of other emerging technologiesFor each of it, please follow the coding conventions, specified by that technology.
- Sometimes you need to break the rules and use non-standard syntax for good reasons. Try to keep this to a minimum.

How to Follow HTML Standards

Identify which version of HTML you are using in your document through the DOCTYPE line at the top of your file.

See the W3C site for more information on document types and DOCTYPE statements.

The important thing to remember is that a DOCTYPE statement is essential to assist validation software in checking your document.

- Use tools (supported by W3C) that support standards. In particular, install and use the *Tidy* program or *Tidy GUI* on your computer.
- Use W3C validation markup service to check the syntax of documents you create.
- Refer to W3C for technical and syntax information.

Some Simple HTML standards:

- The names of HTML files should always end with the ".html" extension.

Example:
Good: foo.html
Bad: foo.bar
- Always include a <HTML> tag at the very beginning and a </HTML> tag at the very end of your HTML documents.

- Always use the <HEAD> and </HEAD> tags to define a header section in your HTML documents.
- Always give your documents a title by using the <TITLE> and </TITLE> tags in the header section of your HTML documents.
- Always use the <BODY> and </BODY> tags to define the body in your HTML documents, which is everything in your document between the <HTML> and </HTML> that is not contained in your header section.
- Use the horizontal line tag <HR> to place a horizontal line beneath any prominent headers in your documents to help them stand out from the surrounding information.

Example:

```
<H1>My Document's Title</H1>
<HR>
```

- Always include a LINK with REV="MADE" in the header section of your HTML documents identifying you as the author.

Example:

```
<LINK REV="MADE" HREF="mailto:your_logonid@cs.niu.edu">
```

- Reasonable line lengths (no greater than 80 characters).
- Attributes associated with tags must be enclosed in quotes.

Example:

```

```

- Code is written in a consistent case. All command tags should be completely capitalized, in order for the tags to stand out better from the surrounding text.

Example:

Good: This text is emphasized.

Bad: This text is emphasized.

- All code should include comment tags for readability, particularly when nested tables are used.
- Images have *alt*, *height*, and *width* attributes. They must be placed in the same directory as the HTML files. These images must be referenced in the code as:

Example:

Good:

bad: .

- Links are coded correctly. All "HREF=" fields in anchor tags should always be enclosed in quotes.

Example:

Good:

Bad:

- Confirm that ©, ®, ™, and ℠ marks are coded correctly. These special characters should always be coded using their respective ASCII codes. It should also be confirmed that the superscription of these characters is done in a consistent manner.

Example :

Please code these special characters as follows:

and Ampersand: andamp;

© Copyright: and#169;

® Registration: and#174;

™ Trademark: and#153;

- Check links. There is nothing more frustrating to users than a broken link (except possibly the blink tag). If the review is of an entire site or a complete section of a site, it is helpful to use an automated link checker. Because there may be hundreds, or even thousands of links, the chance of missing one when checking them by hand is unacceptably high. Since Quality Assurance is not involved in the actual construction of a site, the producer/webmaster needs to verify that links are pointing to the correct pages that those pages still exist, etc.
- If you code a URL which does not specify a file name, always end the URL with a front slash (some browsers choke if you do not do this).

Example:

Good:

Good:

Bad:

Bad:

- Whenever possible, use logical formatting tags instead of physical —one. Let the client's browser figure out the best way to display the information.

Preferred: You should read the book <CITE>Neuromancer</CITE>

Preferred: This text should stand out

Discouraged: You should read the book <I>Neuromancer</I>

Preferred: This text should <BOLD>stand out</BOLD>

- Always "sign" any HTML documents that you create. Include a horizontal line and a link to your homepage (using the ADDRESS style) at the very bottom.

Example:

...and this is the end of my document's text.<P>

<HR>

 <ADDRESS> WWW</ADDRESS>

</BODY>

</HTML>

Appendix B: Table of Figures

Figure 1: First.html in a browser	5
Figure 2: MyFirstPage.html Output	6
Figure 3: Headers.html Output.....	7
Figure 4: Address.html Output	7
Figure 5: PreFormattedText.html Output	8
Figure 6: Resume Page	9
Figure 7: Fruits Table.....	10
Figure 8: Product table.....	12
Figure 9: Calendar.....	12
Figure 10: List	13
Figure 11: Subject list	15
Figure 12: Big Company home page.....	16
Figure 13: About	16
Figure 14: Products	17
Figure 15: Contact	17
Figure 16: EmployeeDetails.....	19
Figure 17: Sales Department	19
Figure 18: Training Department.....	19
Figure 19: Accounts Department	19
Figure 20 : Imagemap	20
Figure 21: Engine	21
Figure 22: Frames.....	23
Figure 23: Courses.....	23
Figure 24: Menu	24
Figure 25: Forms	26
Figure 26: Employee Details	28
Figure 27: New Form Elements.....	32

Appendix C: Table of Examples

Example 1: MyFirstPage.html.....	6
Example 2: Headers.html.....	6
Example 3: Address.html	7
Example 4: PreFormattedText.html	8
Example 5: Fruit Table.....	11
Example 6: Types of Lists.....	14
Example 7: Big Company.....	17
Example 8: Products	18
Example 9: Welcome to Big Company.....	18
Example 10: Image Maps	21
Example 11: Bike Image	22
Example 12: Frames (1).....	24
Example 13: Frames (2)	24
Example 14: Courses	25
Example 15: Menu	25
Example 16: Forms	27
Example 17: Code for New Form Elements.....	31