# WEB TECHNOLOGY

## II Year CS

## SMCA31

By
Dr.N.N.Krishna Veni,
Assistant Professor,
Holy Cross Home Science College,
Tuticorin

# Unit I
# Introduction

**1.1 Internet**

**1.2 Internet Services and Accessibility**

**1.3 Uses of the Internet**

**1.4 Protocols**

**1.5 Web concepts**

- **1.5.1 The client/server model at the web**
- **1.5.2 Retrieving data from the web.**

# Introduction

## 1.1 WHAT IS THE INTERNET?

- Internet or internetworking refers to a wide network through which computers are interconnected globally with one another, and capable of sharing resources among themselves.

## 1.2 INTERNET SERVICES AND ACCESSIBILITY

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

# Introduction

## Different types of Internet connections

- **Dial-Up Connections:** The dial-up access to Internet is a very popular method because it is less expensive. To use this access, the computer should have a modem to connect to the phone system and a software that uses the modem. This software instructs the modem to place a telephone call to the number provided by the ISP. At the ISP, another modem answers the call by giving the connection.
- **High Speed Connections**: High Speed Connections allow information to travel quickly. With this connection, a lot of Internet services work better at higher speeds. There are four basic types of high speed connections:
  - **Digital Subscriber Lines (DSL):** With DSL, the computer is always connected to the Internet. This technology sends digital data through the existing phone lines to carry Internet services.
  - **Cable:** Although the same wires are used, there is no interference to/from the telephone calls. The Internet signal is carried on the TV cable network.
  - **Satellite:** This connection needs a phone line to send information. It receives data at high speed from the satellite.
  - **Integrated Services Digital Network (ISDN):** ISDN is an international communications standard for sending voice, video and data over digital telephone lines or over normal telephone lines.

# Introduction

## 1.3 USES OF THE INTERNET

- Telecommuting (working from home or anywhere) and online conferencing
- Business, advertising and online shopping (e.G. Dell.Com for computers, amazon.Com for books)
- News, jobs, softwares
- Online courses, virtual classrooms, coachings
- Government services, politics and national defense
- Electronic publishing (magazines, newspapers and news services)
- Entertainment (television, radio, videos, audio mp3s, etc.)
- Teaching and learning (course websites, conferencing, simulation, visuali-zation, etc.)
- Scholarly research (searchable databases of journal articles; individual web publishing, etc.)
- General information about a subject, especially which is not easily available from other sources
- Correspondence (email, chatting, etc.)

# Introduction

## 1.4 PROTOCOLS

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

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

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

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

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

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

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

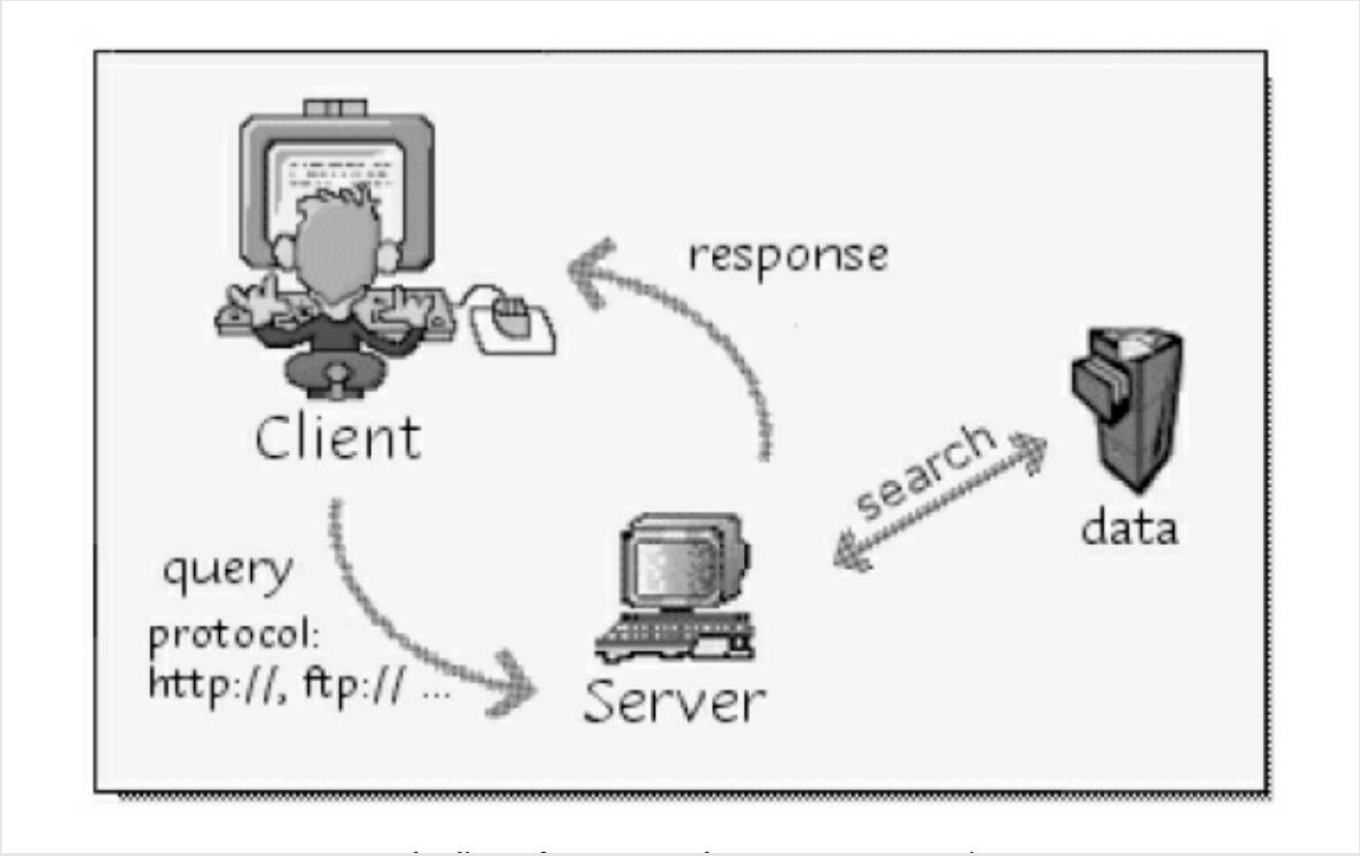
# Introduction

## 1.5 WEB CONCEPTS

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

# Introduction

## 1.5.1 The Client/Server Model of the Web



# Introduction

## 1.5.2 Retrieving Data from the Web

We can retrieve data by using the address of the file. This address is called a Uniform Resource Locator (URL).

- The **Address box** of the browser shows the URL of the currently displayed document. We can type a URL into the address box and can get the file we wanted.
- **Structure of URLs:** In the same way as each host computer has a unique IP address, every data file or document on the Internet also has a unique address called a URL (Uniform Resource Locator). The URL consists of three parts: the protocol, the domain name and the path.
- **The protocol**, is the set of rules which the computer follows in order to communicate with other computers. It lets the computer know how to process the information it receives. If the protocol is http://, for example, the computer knows it will be processing a World Wide Web document.
- **The domain name** is the Internet address of the computer (server) that is hosting the site and storing the documents. This domain name may be expressed as an IP address.
- **The path** is the directory and file specification; it lets the computer know which directory and file to access after connecting to the server. The path is not a required element, but if you know the path, it will take you directly to the desired file or document. The path is also part of the URL which changes frequently.

# Introduction

### 1.5.3 How the Web Works

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

# WEB TECHNOLOGY

## II Year CS

## SMCA31

By
Dr.N.N.Krishna Veni,
Assistant Professor,
Holy Cross Home Science College,
Tuticorin

# Unit I
# Introduction

## 1.6 Internet Protocols

# Internet Protocols

The Internet is a collection of end-systems that communicate using the TCP/IP protocol suite. TCP/IP stands for Transmission Control Protocol/Internet Protocol. A protocol suite is a set of protocols defining services at a number of layers. The TCP/IP protocol suite includes services at the network and transport layers.

### 1.6.1 INTERNET PROTOCOL (IP)

The network layer of TCP/IP is called IP (Internet Protocol). Since this is a network layer, it is responsible for sending data between end-systems on different networks. IP networks have the following properties.

- Each end-system has a unique address (and IP address). Some systems have multiple network interfaces and therefore multiple IP addresses.
- The IP layer in each end-system uses a data link layer to transmit and receive IP packets.

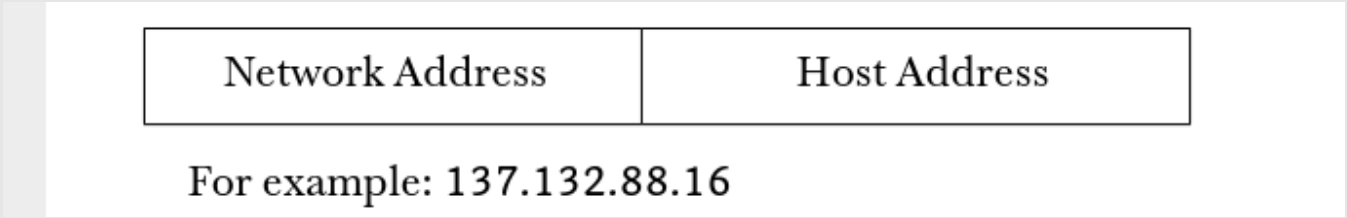
<table>
<tr><td>FTP</td><td>NFS</td><td>5–7</td><td rowspan="4">Application Layer<br>Presentation Layer<br>Session Layer</td></tr>
<tr><td>Telnet</td><td>XDR</td><td></td></tr>
<tr><td>SMTP</td><td>RPC</td><td></td></tr>
<tr><td>SNMP</td><td></td><td></td></tr>
<tr><td colspan="2">TCP/UDP</td><td>4</td><td>Transport Layer</td></tr>
<tr><td colspan="2">Routing Protocols, IP, ICMP</td><td>3</td><td>Network Layer</td></tr>
<tr><td colspan="2">Data Link Layer</td><td>2</td><td>Data Link Layer</td></tr>
<tr><td colspan="2">Physical Layer</td><td>1</td><td>Physical Layer</td></tr>
</table>

# Internet Protocols

This is accomplished using network addresses (identifying networks) and host addresses, addressing individual hosts (end-systems) on a network.

The combination of network and host address makes up the IP address. IP contains addressing information and control information to send packets. The main features of IP are to provide connectionless delivery of datagrams and provide fragmentation and reassembly of datagrams.

**IP addresses:** Each IP address is a pair of network address (netid) and host address (hostid).




**Example:**128.213.1.1

The first part of each IP address specifies the network address, for example, the network address 128.213 is the address for the Computer Science (CS) department.

The last 2 digits specify the host (end-system) within the CS network.

Therefore, netid 128.213 and hostid 1.1 form the IP address 128.213.1.1.

# Internet Protocols

When routing data packets, the routers look at the network address. Each router has a table called a routing table that indicates where it should send a packet given the destination network address.

**Routing Table:**

| *Destination Network Address* | *Forward to* |
|---|---|
| 177.132 | Router at 128.213.7.3 |
| 128.113 | Router at 128.213.1.4 |
| 144.13.7 | Router at 128.113.7.2 |
| 128.213.7 | Network A (128.213.7) |
| 128.113.7 | Network B (128.113.7) |
| 128.213.1 | Network C (128.213.1) |
| Anything else | Router at 128.213.1.4 |

# Internet Protocols

**Types of Internet addresses:** Internet addresses are divided into 5 classes A, B, C, D, and E

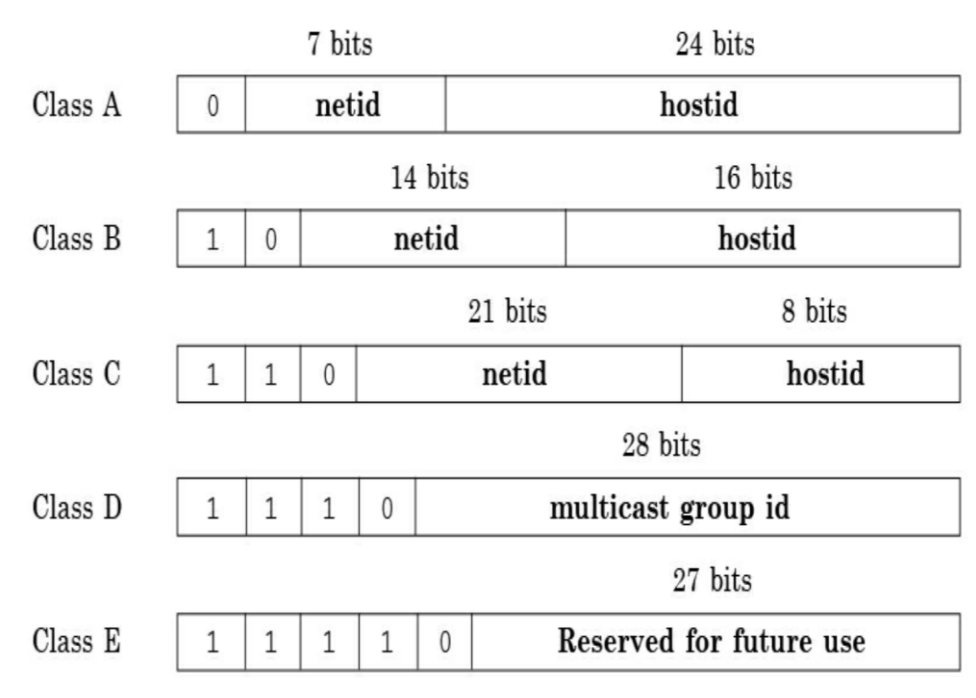



FIGURE 2.1 Class of IP addresses.

| Class | Address Range |
|---|---|
| A | 0.0.0.0 to 127.255.255.255 |
| B | 128.0.0.0 to 191.255.255.255 |
| C | 192.0.0.0 to 223.255.255.255 |
| D | 224.0.0.0 to 239.255.255.255 |
| E | 240.0.0.0 to 247.255.255.255 |

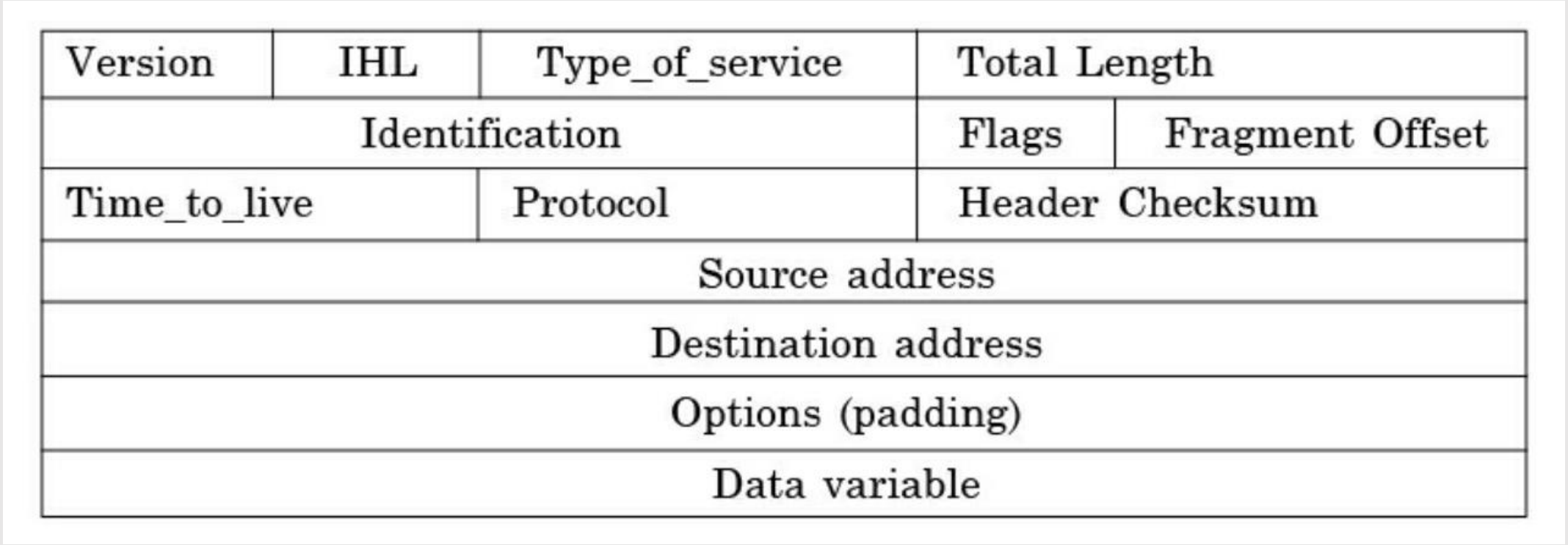
| Class | Lowest Network Identifier Address | Highest Network Identifier Address |
|---|---|---|
| A | 1.0.0.0 | 126.0.0.0 |
| B | 128.0.0.0 | 191.255.0.0 |
| C | 192.0.0.0 | 223.255.255.0 |
| D | 224.0.0.0 | 239.255.255.255 |
| E | 240.0.0.0 | 247.255.255.255 |

FIGURE 2.2 Address ranges in each class.

- Certain addresses have special meanings, 0, 127 and 255 are usually reserved for special use.
- 255 indicates a broadcast address
- The network address with 127 as the first byte is the loopback network, which is fictitious. The address 127.0.0.1 is called localhost and means the current host machine. 224.0.0.1 is the multicast address for all systems in the sub-network.

# Internet Protocols

**Packet format:**



| Version | IHL | Type_of_service | Total Length | |
|---|---|---|---|---|
| Identification | | | Flags | Fragment Offset |
| Time_to_live | | Protocol | Header Checksum | |
| Source address | | | | |
| Destination address | | | | |
| Options (padding) | | | | |
| Data variable | | | | |

# Internet Protocols

## Packet format:

The details of each field are given below:

| *Version* | *Specifies the version of IP* |
|---|---|
| IHL | IP header length which is of 32-bit words |
| TOS | Define how to handle datagrams |
| Total length | Specifies the total length of the IP packet in bytes |
| ID | Identification number used to identify the datagram fragments |
| Flags | This field consists of 3 bits: Lower order bit—specifies whether datagram is fragmented, Middle bit—specifies whether the datagram fragment is the last one, Higher order bit—not used |
| Fragmentation offset | Specifies the relative position of the fragment from the beginning of fragment data |
| TTL | A counter that is decremented every time a router receives the datagram and the datagram is discarded if it becomes zero |
| Protocol | Indicates which upper layer protocol is present |
| Header checksum | Used for ensuring the correctness of datagram packet |
| Source address | IP address of source |
| Destination address | IP address of destination |
| Options | Specifies various options, like security, to be set |

# WEB TECHNOLOGY

## II Year CS

## SMCA31

By
Dr.N.N.Krishna Veni,
Assistant Professor,
Holy Cross Home Science College,
Tuticorin

# Unit I
# Introduction

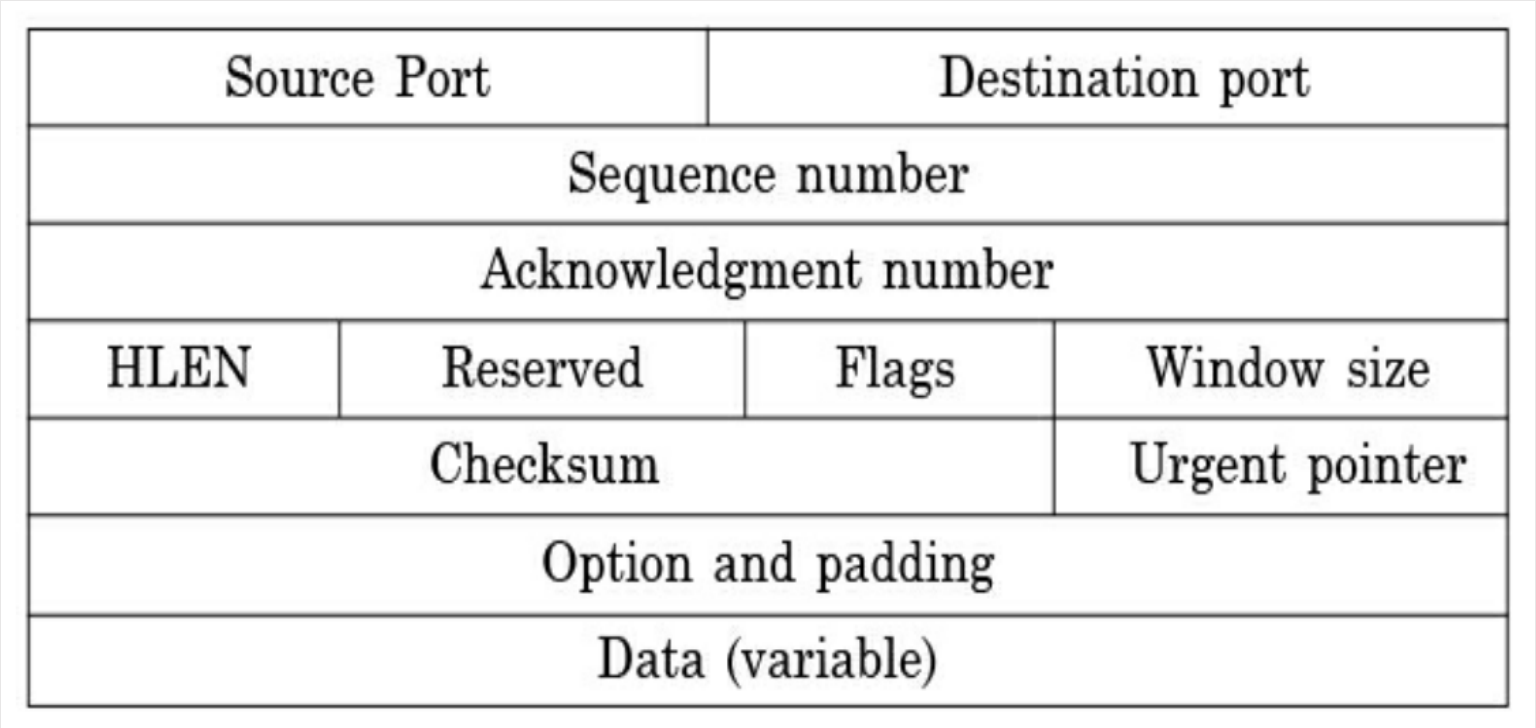
## 1.6 Internet Protocols(Cont..)

## 1.7 Host Names

# Internet Protocols

## 1.6.2 TRANSMISSION CONTROL PROTOCOL (TCP)

- The TCP provides reliable transmission through connection-oriented, end-to-end reliable packet delivery.
- It receives data from application layer and hand it over to IP layer.
- Timeout mechanism is used to detect lost packets and requests for retransmission.



# Internet Protocols

| | |
|---|---|
| Source and destination port | Specifies the port numbers at source host and destination host |
| Sequence number | Number assigned to first byte of data stream if it has been split |
| Acknowledgment number | Specifies acknowledgment from the other end |
| Reserved | Reserved for future use |
| HLEN | Length of the header in bytes |
| Flag | Defines 6 control bits for enabling flow contol, connection establishment, and termination, and the mode of transfer |
| Window size | Defines the size of the window |
| Checksum | Provided for error detection and correction |
| Urgent pointer | Used if the segment contains urgent data |
| Options | To include some optional information in the header field |

# Internet Protocols

## 1.6.3 User Datagram Protocol (UDP)

- User Datagram Protocol is a connectionless transport layer protocol.
- UDP adds no reliability, flow control or error recovery.
- UDP is used in situations, where reliability is not necessary and it is very helpful, in multimedia and multicasting applications.
- Most applications on the Internet are based on TCP.
- Both TCP and UDP must provide communication between processes identified by protocol ports. So the destination address of any TCP or UDP data includes an IP address and a protocol port number.
- Many network services operate on a prescribed port number.

### Packet format:

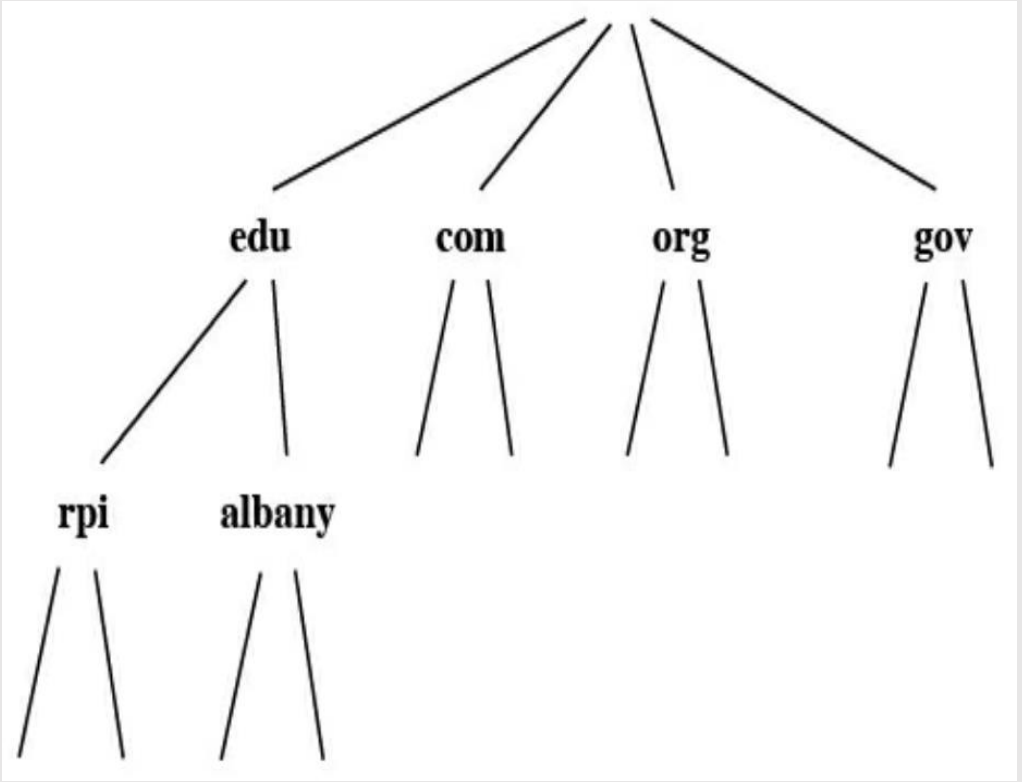
| | |
|---|---|
| Source port | Destination port |
| Length | Checksum |

| | |
|---|---|
| Source port | Port number of source host |
| Destination port | Port number of destination host |
| Length | Total length of the user datagram |
| Checksum | Used to detect errors for the entire datagram |

# Host Names

Users typically do not deal with IP addresses, but deal with the host names. These names are arranged in a hierarchical structure.

- At the top level, there are domains corresponding to educational institutions (.edu), commercial entities (.com), public organizations (.org), government entities (.gov), etc. There are also top level domains for countries.
- Each individual organization gets a name at the second level in the name hierarchy. For example, Microsoft has microsoft.com. It is possible to add additional naming levels within organizations; and this is typically done to identify sub-organizations or departments within the larger organization.
- Finally, each individual host (end-system) has a unique name within its department or sub-organization.



# WEB TECHNOLOGY

## II Year CS

## SMCA31

By
Dr.N.N.Krishna Veni,
Assistant Professor,
Holy Cross Home Science College,
Tuticorin

# Unit II
# HTML

## 2.1 Introduction

## 2.2 SGML

# HTML

## 2.1 INTRODUCTION

- HTML (the Hypertext Markup Language) is a ***predominant*** markup language.
- It ***describes the structure*** of text-based information in a document by identifying some texts as heading, paragraph, lists, etc.
- It can also ***describe the appearance and semantics*** of a text and makes it easy for any browser to render that page into the computer screen.
- In HTML, text is marked up with elements ***delineated by tags*** that are keywords contained in pairs of ***angled brackets.***
- HTML tags are ***not case sensitive***.
- The ***absence of a closing tag*** for elements that require them is considered as a syntax error and can grossly affect the formatting and look of a page.
- HTML documents ***can be created with any text editor or Front Page Express***.
- All HTML files have either ***the .htm or the .html*** file name extensions.

# HTML

## 2.2 SGML

- Standard Generalized Markup Language (SGML) is an international standard for the description of marked-up electronic text.
- SGML was made international standard by ISO in October 1986.
- SGML is a meta language which in itself does not define a markup language, but provides a framework to develop various kinds of markup languages.
- It is a language to represent document structure but is not a document structure itself.
- Every markup language defined by SGML is called as SGML application.
- An application is characterized by
  - ***Declaration:*** *specifies character and delimiters that appear in the application.*
  - ***Document Type Definition (DTD):*** *defines the format of document constructs.*
  - ***Specification*** *that tells about the semantics of the defined markup.*
  - ***Document instances*** *that contain markup and data.*

# HTML

## 2.2.1 DTD

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

<document>
<header>
<title> Introduction to SGML
<authors> Ramesh and Puri
<abstract> A short description on SGML and its constructs
<body>
<block> Standard Generalized Markup Language is popularly called as SGML
<block> The link can be found at <link url="http://www.foo.com"></link>
<ack> The following is acknowledged by peer authors
</document>

# HTML

## 2.2.2 DTD ELEMENTS

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

- name of the element
- two characters that specify the minimization rules a
- content model

**O**- Letter O indicates that it may be omitted.

**-** - The hyphen indicates that the tag must be present

**+** - The plus sign indicates that the element may occur for one or more times.

**?** - The question mark means that the elements may atmost one and possibly no occurrence

* - The asterisk indicates that the element may either be absent or appear one or more times.

, - The components it connects must appear in the order.

& - The ampersand is similar to comma specification but in any order.

| - The vertical bar indicates that only one of the components it connects will appear.

# HTML

## 2.2.3 ATTRIBUTES

Attributes are used to describe information for a particular element.

For example, <link url="http://www.foo.com"></link>

The declaration begins with ATTLIST keyword followed by name of the element to which the attributes should be added.

The declaration then contains the following components:

- name of the attribute
- type of attribute value or set of possible values.
- how a parser should interpret the absence of attribute concerned.

This is emphasized by the following keywords:

**#REQUIRED** - A value must be specified

**#IMPLIED** - Value need not be given.

**#FIXED** - Fixed value must be given.

# WEB TECHNOLOGY

## II Year CS

## SMCA31

By
Dr.N.N.Krishna Veni,
Assistant Professor,
Holy Cross Home Science College,
Tuticorin

# Unit II
# HTML

**2.3 Outline of an html document**

**2.4 Head Section**

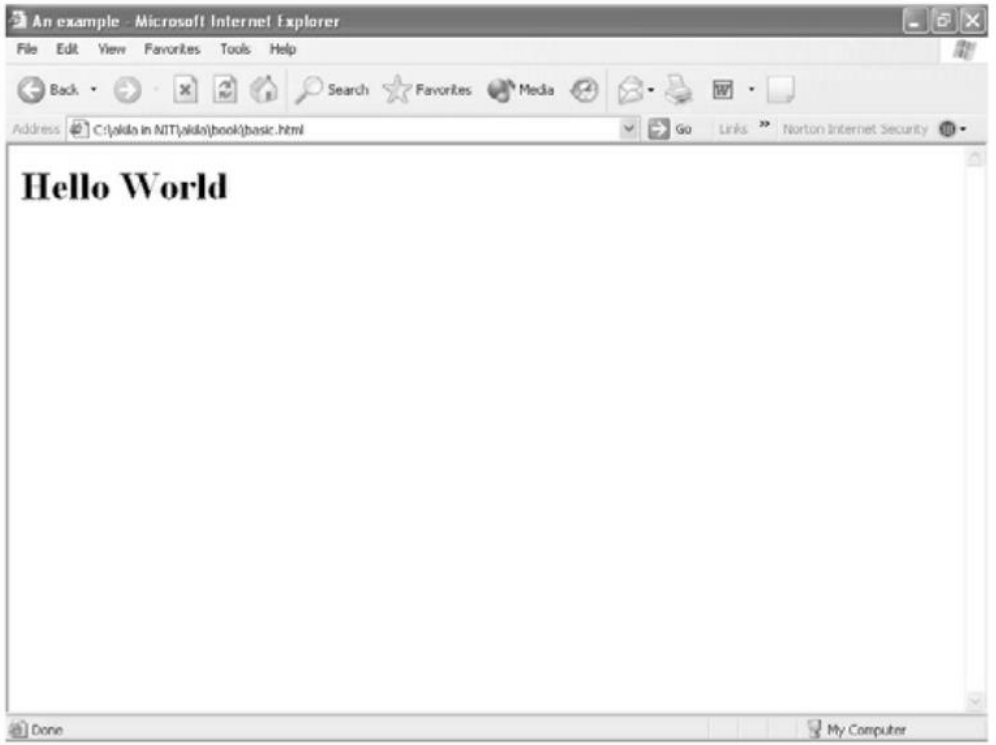
**2.5 Body Section**

# HTML

## 2.3 Outline of an html document

- The HTML tag is a keyword used to specify the appearance of the Web page.
- Every tag should be enclosed with angled brackets and should have closing tag like <html>...</html>.
- There are various tags under the body tag. The comments are enclosed in the special tags <!— comment-->.

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



# HTML

## 2.4 HEAD SECTION

- **Prologue:** Its only a comment which tells about the HTML version.

  <! Doctype HTML 4.0>

- **Link:** The link tag is used to inform the browser's previous and next document.

  <Link rel = previous href = "prev.html">

- **Base:** This tag declares global reference values for href and target attribute.

  <base href="base_url">

- **Meta:** The meta tag defines the document's meta information such as keywords, expiry date, author, page generation software used, etc.

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

- **Script:** The script tag contains the code referenced in the body of the document.

  <script language="javascript">

  ----- script code

  </script>

- **Style:** This tag specifies the style information for the document.

  <style type "mime_type">

**For example,** <style type = "text/css"> ----- style statements </style>

# HTML

## 2.5 Body Section

The tag consists of various attributes. They are listed below.

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

## Headers

- These are formatting tags to vary the text size.
- There are six header tags H1 through H6.

```
<h1> Hello World </h1>
<h2> Hello World </h2>
<h3> Hello World </h3>
<h4> Hello World </h4>
<h5> Hello World </h5>
<h6> Hello World </h6>
```

# HTML

## Paragraphs

- The tag which is used to start a paragraph is ... .
- This tag also consists of align attribute with the same values.
- To insert a line in between, the <br> tag is used.

## Text Formatting

- The page content in the body of the Web page can be presented in a neat way using different tags.
- The  tag is used to make the text in italics.
- The  tag is used to make text as bold.
- The <font> tag is also used for text styling. This tag contains various attributes like face, color and size.

## Linking

- The links are created using anchor <a>...</a> element.
- This element requires an attribute to mark the location of the object to get linked.
- The address of the object is specified using href attribute.

<a href=”http://www.yahoo.com”>

# HTML

## Internal Linking(Ex:1st lab program)

- To create name, the following tag is used. <a name="name"> </a>
- To refer to the location use <a href = "#name"> text </a>

## Embedding Images

- To add images as a background for the whole Web page, the following format is used.
  <body background="bg.gif">
- To include the image as a normal figure in the Web page the <img> tag is used. The image tag has the following attributes.
  - Src
  - Height
  - Width
  - Border
  - Align
  - Alt
- The other related tag is <map> which is used to create hot spots. The syntax is,

  <map name="name">

  -----

  </map>

# WEB TECHNOLOGY

## II Year CS

## SMCA31

By
Dr.N.N.Krishna Veni,
Assistant Professor,
Holy Cross Home Science College,
Tuticorin

# Unit II
# HTML

## 2.5 Body Section

- List
- Tables

# HTML

**Lists:** Lists are a way of creating several kinds of outlines, menus, etc. There are three different kinds of lists to choose from: ordered, unordered, and definition.

**Unordered:** For including unordered list of items start with <ul>, and end with </ul>.

- The items inside the <ul> tag should be included within <li>.
- The <li> tag has an attribute type with values **like disk, circle, or square**.

**Ordered:** For including ordered list of items start with <ol> and end with </ol>.

- It has attributes like **compact, start, type, etc**.
  - ✓ Compact - specifies the display in compact form,
  - ✓ Start - indicates number to start a
  - ✓ Type - specifies the kind of numbering to use like

**Definition lists:** Specifically used for lists in which each element is labelled with a word rather than a bullet or number.

- It should start with <dl> and end with </dl>.
- The items are included within pairs: <dt> and <dd>
- where <dt> to include definition term and <dd> to include actual definition.

# HTML

## Tables

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

The syntax is

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

# WEB TECHNOLOGY

## II Year CS

## SMCA31

By
Dr.N.N.Krishna Veni,
Assistant Professor,
Holy Cross Home Science College,
Tuticorin

# Unit II
# HTML

## 2.5 Body

- **Frames**
- **Other Special Tags**

# HTML

## Frames

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

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

**<frame**

**src="URL"**

**name="name"**

**frameborder="0 | 1"**

**scrolling="yes | no | auto">**

# HTML

## Other Special Tags and Characters

- &amp is used to insert an ampersand.
- The tag <del> is used to strike through text with a horizontal line.
- The tags <sub> and <sup> are used to turn the text into subscript and superscript respectively.
- To draw a horizontal line in the Web page <hr> tag is used.
  - *Width—to specify the width of the line.*
  - *Size—to specify the height of the line.*
  - *Noshade—eliminates the default shading effect*

# WEB TECHNOLOGY

## II Year CS

## SMCA31

By
Dr.N.N.Krishna Veni,
Assistant Professor,
Holy Cross Home Science College,
Tuticorin

# Unit II
# HTML

## 2.5 Body

- **HTML Forms**

# HTML

**HTML Forms:** Special kind of tags called form tags which are used to create interactive content.
The syntax is

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

**INPUT:**
This tag places the following form controls:

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

# HTML

**The syntax is**

<input type="text | passwd" name="name" value="default_val" size="field size"
maxlength="mas_in_length" DISABLED READONLY>

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

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

**TEXT AREA:** This tag sets up a multiple line input text window.

**The syntax is**

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

# WEB TECHNOLOGY

## II Year CS

## SMCA31

By
Dr.N.N.Krishna Veni,
Assistant Professor,
Holy Cross Home Science College,
Tuticorin

# Unit III
# Java Script

**3.1 Introduction**

**3.1.1 Need Of A Scripting Language**

**3.1.2 Language Elements**

**3.1.2.1 Identifiers:**

**3.1.2.2 Expressions:**

**3.1.2.3 Javascript Keywords**

**3.2 Operators**

# JAVA SCRIPT

## INTRODUCTION

- JavaScript was designed to provide a quicker and simpler language for enhancing Web pages and servers.
- The difference between Java and JavaScript is that Java is a general purpose object oriented language. On the other hand, JavaScript enables to embed small program codes in an HTML page.

## NEED OF A SCRIPTING LANGUAGE

- HTML is static & HTML documents have limited interactions with the users through HTML forms. JavaScript introduces increased functionality for Web browsers in the form of Java applets, plugins, and DHTML elements.
- JavaScript provides a fairly complete set of built-in functions and commands.
- Code to perform these actions can be embedded in a page and executed when the page is loaded.
- The JavaScript code is embedded in the HTML document with in <script> and </script> tags.
- HTML also contains <noscript> tag which can be used in the HTML document to display an alternative text if the browser does not support script languages.

# JAVA SCRIPT

## LANGUAGE ELEMENTS

### Identifiers:

- An identifier is a unique term that JavaScript uses to identify a variable, method, or an object.
- The identifiers can take two forms—literals and variables.
- Literals are identifiers having fixed values.
- Variables have different values during execution. Some of them are:
  - *Integers*
  - *Floating point numbers*
  - *Strings*
  - *Booleans*

### Expressions:

- An expression is a statement that is evaluated to a value. The result can be of any type.
- Expression contains variables, literals, operators and even another expression.
- The following are some of the examples.
  - *Y=67;*
  - *Str="Good Morning";*
  - *Val= x + y * b;*

# JAVA SCRIPT

## JavaScript Keywords

The language specifies a set of keywords which are reserved for a specific purpose only. These words cannot be used as variables or constants. The following are the keywords:

| | | | | | |
|---|---|---|---|---|---|
| abstract | Boolean | break | byte | case | catch |
| char | class | const | continue | debugger | default |
| delete | do | double | else | enum | export |
| extends | false | final | finally | float | for |
| function | goto | if | implements | import | in |
| Instanceof | int | interface | long | native | new |
| null | package | private | protected | public | return |
| short | static | super | switch | synchronized | this |
| throw | throws | transient | true | try | typeof |
| var | void | volatile | while | with | |

# JAVA SCRIPT

## OPERATORS

- Operators are commands, which perform operations on variables and/or literals and produce a result.
- There are two types of operators—**unary and binary**.
- Unary operators operate on only one operand and binary operators operate on two operands.

**Assignment operators:** This operator takes the result of an expression and assigns it to a variable.

For example,

x=a+b;

The other assignment operators are:

| Operators | Usage | Equivalent expression |
|---|---|---|
| += | a + = b | a = a + b |
| -= | a – = b | a = a – b |
| * = | a * = b | a = a * b |
| / = | a/ = b | a = a / b |
| %= | a% = b | a = a % b |

# JAVA SCRIPT

## Math operators

- These operators include:

  + addition

  – subtraction

  * Multiplication

  / division

- Apart from the above operators, JavaScript supports unary increment (++) and decrement (–) operators and unary minus.
- The following examples show how the increment and decrement operators are used.

  X = ++a; // increments a and assigns to X.

  X = a++; // assigns a to X and then increments.

  X = --a; // decrements a and assigns to X.

  X = a--; // assigns a to X and then decrements.

# JAVA SCRIPT

## Comparison operators:

The operators are

< less than
>greater than
<= less than and equal
>= greater than and equal
== equal
!= not equal

## Logical operators:

There are three logical operators.

&& returns true only if both the operands or expressions are true
|| returns true even if one is true
! used to negate the expression

## String operators:

The comparison operators can be used with the strings.

For concatenating two strings ‘+’ is used.

Str = “good” + “day”;

# WEB TECHNOLOGY

## II Year CS

## SMCA31

By
Dr.N.N.Krishna Veni,
Assistant Professor,
Holy Cross Home Science College,
Tuticorin

# Unit III
# Java Script

## Statements

# JAVA SCRIPT

## 5.2.5 Statements

**FOR LOOP**

The for loop consists of three expressions enclosed within brackets and separated by semicolon.

- *Initial expression—initializes the loop variables. It is evaluated before the loop starts.*
- *Condition—evaluated in each iteration and checks whether evaluated to true or false.*
- *Update expression—changes the value of the loop variables.*

**Syntax:** for ([initial];[condition];[update])

```
{
   statements
 }
```

# JAVA SCRIPT

**2. FOR...IN LOOP**

This is a special form of loop which iterates the variable over all the properties in an object.

**Syntax:**

```
for (var in obj)
{
   Statements
}
```

**3. WHILE AND DO...WHILE LOOP**

The while statement contains a condition and block of statements. If the condition is evaluated to true, the block of statements are executed.

**Syntax:**

```
while (condition)
{
  statements;
 }
```

```
do
{
  statements;
}while (condition);
```

# JAVA SCRIPT

**BREAK STATEMENT:** This statement is used to terminate the current iteration in the while or for loop**The syntax is:** break;

**CONTINUE STATEMENT:** This statement is used to continue the loop for the next iteration without executing the statements following the continue statement.

**The syntax is:** continue;

**IF...ELSE STATEMENT :**

```
If (condition)
{
   block1;
}
[else
{
    block2;
}]
```

# JAVA SCRIPT

**VAR STATEMENT:**

This statement is used to declare the variables. Initialization can be done in this statement itself.

**The syntax is**

var name1 [=value], var name2 [=value],...

var x=0, y=1;

# WEB TECHNOLOGY

## II Year CS

## SMCA31

By
Dr.N.N.Krishna Veni,
Assistant Professor,
Holy Cross Home Science College,
Tuticorin

# Unit III
# Java Script

**3.1 Introduction**

**3.1.1 Need Of A Scripting Language**

**3.1.2 Language Elements**

**3.1.2.1 Identifiers:**

**3.1.2.2 Expressions:**

**3.1.2.3 Javascript Keywords**

**3.2 Operators**

# JAVA SCRIPT

## 5.2.5 Statements

**FOR LOOP**

The for loop consists of three expressions enclosed within brackets and separated by semicolon.

- *Initial expression—initializes the loop variables. It is evaluated before the loop starts.*
- *Condition—evaluated in each iteration and checks whether evaluated to true or false.*
- *Update expression—changes the value of the loop variables.*

**Syntax:** for ([initial];[condition];[update])

```
{
   statements
 }
```

# JAVA SCRIPT

**2. FOR...IN LOOP**

This is a special form of loop which iterates the variable over all the properties in an object.

**Syntax:**

```
for (var in obj)
{
   Statements
}
```

**3. WHILE AND DO...WHILE LOOP**

The while statement contains a condition and block of statements. If the condition is evaluated to true, the block of statements are executed.

**Syntax:**

```
while (condition)
{
  statements;
 }
```

```
do
{
  statements;
}while (condition);
```

# JAVA SCRIPT

**BREAK STATEMENT:** This statement is used to terminate the current iteration in the while or for loop**The syntax is:** break;

**CONTINUE STATEMENT:** This statement is used to continue the loop for the next iteration without executing the statements following the continue statement.

**The syntax is:** continue;

**IF...ELSE STATEMENT :**

```
If (condition)
{
   block1;
}
[else
{
    block2;
}]
```

# JAVA SCRIPT

**VAR STATEMENT:**

This statement is used to declare the variables. Initialization can be done in this statement itself.

**The syntax is**

var name1 [=value], var name2 [=value],...

var x=0, y=1;

**FUNCTIONS:**

In JavaScript, functions are created in the following way:

```
function name( )
{
    statement;
    statement;
    statement
}
```

# JAVA SCRIPT

Functions can be called from within other functions.

**For example:**

```
function sayGoodbye()
{
   alert(“Goodbye!”)
}
function sayHello()
{
   alert(“Hi”);
  sayGoodbye()
}
```

# WEB TECHNOLOGY

## II Year CS

## SMCA31

By
Dr.N.N.Krishna Veni,
Assistant Professor,
Holy Cross Home Science College,
Tuticorin

# Unit IV
# Objects of Java Script

The Window Object

# OBJECTS OF JAVA SCRIPT

## The Window Object

Window is the fundamental object in the browser. It represents the browser window in which the document appears. The properties include

- **Status** The contents of the status bar (at the bottom of the browser window).

  For example, window.status = “Hi!”; will display the string “Hi!” on the status bar.

- **Location** The location and URL of the document currently loaded into the window (as displayed in the location bar).

  For example, alert(window.location); will display an alert containing the location

- **Length** The number of frames (if any) into which the current window is divided.

  For example, alert(window.length); will display the number of frames in the current window.

- **Parent** The parent window, if the current window is a sub-window in a frameset.

  For example, var parentWindow = window.parent;

  alert(parentWindow.length);

- **Top** The top-level window, of which all other windows are sub-windows.

  For example, Var topWindow = window.top;

  alert(topWindow.length);

# OBJECTS OF JAVA SCRIPT

**The methods include:**

- **alert()** Displays an ‘alert’ dialog box, containing text entered by the page designer, and an ‘OK’ button.

  **For example,** alert(“Hi!”);

- **confirm()** Displays a ‘confirm’ dialog box, containing text entered by the user, an ‘OK’ button, and a ‘Cancel’ button. The method returns true or false.

  **For example,** var response = confirm(“surely want to Delete File?”);

  alert(response);

- **prompt()** Displays a message, a box into which the user can type text, an ‘OK’ button, and a ‘Cancel’ button. The method returns a text string.

  **The syntax is:** prompt(message_string, default_response_string)

  **For example,** var fileName = prompt(“Select File”, “file.txt”);

  alert(fileName);

# OBJECTS OF JAVA SCRIPT

**The methods include:**

- **open()** Opens a new browser window and loads either an existing page or a new document into it.

  **The syntax is** open(URL_string, name_string, parameter_string)

  **For example,** var parameters = “height=100,width=200”;

  newWindow = open(“NewDoc.html”, “newDocument”, parameters);

- **close()** Closes a window. If no window is specified, closes the current window.

  **The syntax is** window_name.close()

  **For example,** newWindow.close()

**Window events include:**

- **onLoad()** Message sent each time a document is loaded into a window.

  **for example,** <body onLoad=“displayWelcome()”>
- **onUnload()** Message sent each time a document is closed or replaced with another document.

  **for example,** <body onUnload=“displayclose()”>

# OBJECTS OF JAVA SCRIPT

## 5.3.2 The Document Object

The Document object represents the HTML document displayed in a browser window. Document properties include:

- **bgColor** The colour of the background. **For example:** document.bgColor = “green”;
- **fgColor** The colour of the text. **For example:** document.fgColor = “blue”;
- **linkColor** The colour used for un-visited links. **For example:** document.linkColor = “brown.
- **alinkColor** The colour used for an active. **For example:** document.alinkColor = “lightred”;
- **vlinkColor** The colour used for visited links. **For example:** document.vlinkColor = “darkred”;
- **title** The title of the document, as displayed at the top of the browser window.

  **For example:** document.title = “This title is displayed now”;
- **forms** An array containing all the forms (if any) in the document. It accepts an index number in the following way: **forms[index-number]**

# OBJECTS OF JAVA SCRIPT

**Document methods include:**

- **write()** Allows a string of text to be written to the document. It can be used to generate new HTML code in response to user actions.

  **For example:** document.write("<h1>Hello</h1>");

### 5.3.3 Forms Object

Form properties include:

- **name** The name of the form, as defined in the HTML <form>, for example, <form name="myForm">
- **method** The method used to submit the information in the form, for example: <form method="POST">
- **action** The action to be taken when the form is submitted, for example, <form action="mto@bco.com">
- **length** The number of elements (text-boxes, buttons, etc.) in the form. For example: alert(document.forms[2].length);
- **elements** An array of all the elements in the form. Individual elements are referenced by index-number.

# WEB TECHNOLOGY

## II Year CS

## SMCA31

By
Dr.N.N.Krishna Veni,
Assistant Professor,
Holy Cross Home Science College,
Tuticorin

# Unit IV
# Objects of Java Script

The Window Object

# OBJECTS OF JAVA SCRIPT

## Text-boxes and Text-areas

**Properties include:**

- *name*

**Events:**

- *onFocus*
- *onBlur*

## Buttons, Radio-buttons and Checkboxes

**Properties include:**

- *name*
- *Value*

**Methods include:**

- *focus()*
- *blur()*
- *click()*

**Events include:**

- *onclick*
- *onfocus*
- *onblur*

# OBJECTS OF JAVA SCRIPT

**The methods include:**

- **alert()** Displays an ‘alert’ dialog box, containing text entered by the page designer, and an ‘OK’ button.

  **For example,** alert(“Hi!”);

- **confirm()** Displays a ‘confirm’ dialog box, containing text entered by the user, an ‘OK’ button, and a ‘Cancel’ button. The method returns true or false.

  **For example,** var response = confirm(“surely want to Delete File?”);

  alert(response);

- **prompt()** Displays a message, a box into which the user can type text, an ‘OK’ button, and a ‘Cancel’ button. The method returns a text string.

  **The syntax is:** prompt(message_string, default_response_string)

  **For example,** var fileName = prompt(“Select File”, “file.txt”);

  alert(fileName);

# OBJECTS OF JAVA SCRIPT

**The methods include:**

- **open()** Opens a new browser window and loads either an existing page or a new document into it.

  **The syntax is** open(URL_string, name_string, parameter_string)

  **For example,** var parameters = “height=100,width=200”;

  newWindow = open(“NewDoc.html”, “newDocument”, parameters);

- **close()** Closes a window. If no window is specified, closes the current window.

  **The syntax is** window_name.close()

  **For example,** newWindow.close()

**Window events include:**

- **onLoad()** Message sent each time a document is loaded into a window.

  **for example,** <body onLoad=“displayWelcome()”>
- **onUnload()** Message sent each time a document is closed or replaced with another document.

  **for example,** <body onUnload=“displayclose()”>

# OBJECTS OF JAVA SCRIPT

## 5.3.2 The Document Object

The Document object represents the HTML document displayed in a browser window. Document properties include:

- **bgColor** The colour of the background. **For example:** document.bgColor = “green”;
- **fgColor** The colour of the text. **For example:** document.fgColor = “blue”;
- **linkColor** The colour used for un-visited links. **For example:** document.linkColor = “brown.
- **alinkColor** The colour used for an active. **For example:** document.alinkColor = “lightred”;
- **vlinkColor** The colour used for visited links. **For example:** document.vlinkColor = “darkred”;
- **title** The title of the document, as displayed at the top of the browser window.

  **For example:** document.title = “This title is displayed now”;
- **forms** An array containing all the forms (if any) in the document. It accepts an index number in the following way: **forms[index-number]**

# OBJECTS OF JAVA SCRIPT

**Document methods include:**

- **write()** Allows a string of text to be written to the document. It can be used to generate new HTML code in response to user actions.

  **For example:** document.write("<h1>Hello</h1>");

### 5.3.3 Forms Object

Form properties include:

- **name** The name of the form, as defined in the HTML <form>, for example, <form name="myForm">
- **method** The method used to submit the information in the form, for example: <form method="POST">
- **action** The action to be taken when the form is submitted, for example, <form action="mto@bco.com">
- **length** The number of elements (text-boxes, buttons, etc.) in the form. For example: alert(document.forms[2].length);
- **elements** An array of all the elements in the form. Individual elements are referenced by index-number.

# Math object

By

A.dhanusha

# Math Object Methods:

| Method | Description |
|---|---|
| abs(x) | Returns the absolute value of x |
| acos(x) | Returns the arccosine of x, in radians |
| acosh(x) | Returns the hyperbolic arccosine of x |
| asin(x) | Returns the arcsine of x, in radians |
| asinh(x) | Returns the hyperbolic arcsine of x |
| atan(x) | Returns the arctangent of x as a numeric value between -PI/2 and PI/2 radians |
| atan2(y, x) | Returns the arctangent of the quotient of its arguments |
| atanh(x) | Returns the hyperbolic arctangent of x |
| cbrt(x) | Returns the cubic root of x |
| ceil(x) | Returns x, rounded upwards to the nearest integer |
| cos(x) | Returns the cosine of x (x is in radians) |
| cosh(x) | Returns the hyperbolic cosine of x |
| exp(x) | Returns the value of $E^x$ |

| | |
|---|---|
| expm1(x) | Returns the value of $E^x$ minus 1 |
| floor(x) | Returns x, rounded downwards to the nearest integer |
| fround(x) | Returns the nearest (32-bit single precision) float representation of a number |
| log(x) | Returns the natural logarithmof x |
| log10(x) | Returns the base-10 logarithm of x |
| log1p(x) | Returns the natural logarithm of 1 + x |
| log2(x) | Returns the base-2 logarithm of x |
| max(x, y, z, ..., n) | Returns the number with the highest value |
| min(x, y, z, ..., n) | Returns the number with the lowest value |
| pow(x, y) | Returns the value of x to the power of y |
| random() | Returns a random number between 0 and 1 |
| round(x) | Rounds x to the nearest integer |
| sign(x) | Returns the sign of a number (checks whether it is positive, negative or zero) |
| sin(x) | Returns the sine of x (x is in radians) |

| | |
|---|---|
| sinh(x) | Returns the hyperbolic sine of x |
| sqrt(x) | Returns the square root of x |
| tan(x) | Returns the tangent of an angle |
| tanh(x) | Returns the hyperbolic tangent of a number |
| trunc(x) | Returns the integer part of a number (x) |

## Usage : sqrt(x) method

Math.sqrt(x)

**Parameters : x is a number (Required).**

Top-level, predefined JavaScript object.

**sqrt** : The sqrt() method is used to get the square root of a number.

Since **sqrt** is a static method of **Math**, it is always used as **Math.sqrt(x)**

w3r

The math object allows us to perform various mathematical operations that are not provided by basic operators we have already looked at its methods include the following as shown in the table before table.

The math object also has methods to perform trigonometrical operations such as sin(),cos(),tan(),etc.., and a set of properties that includes values for pi()and other constants.


Thank You
WisdomTimes.com

# WEB TECHNOLOGY

## II Year CS

## SMCA31

By
Dr.N.N.Krishna Veni,
Assistant Professor,
Holy Cross Home Science College,
Tuticorin

# PHP &SQL

SUBMITTED BY:

JACQULIN SHRAVYA ROSE.J

# WHAT ARE HTML & CSS?

- *HTML*, HYPERTEXT MARKUP LANGUAGE, GIVES CONTENT STRUCTURE AND MEANING BY DEFINING THAT CONTENT AS, FOR EXAMPLE, HEADINGS, PARAGRAPHS, OR IMAGES. *CSS*, OR CASCADING STYLE SHEETS, IS A PRESENTATION LANGUAGE CREATED TO STYLE THE APPEARANCE OF CONTENT—USING, FOR EXAMPLE, FONTS OR COLORS.
- THE TWO LANGUAGES—HTML AND CSS—ARE INDEPENDENT OF ONE ANOTHER AND SHOULD REMAIN THAT WAY. CSS SHOULD NOT BE WRITTEN INSIDE OF AN HTML DOCUMENT AND VICE VERSA. AS A RULE, HTML WILL ALWAYS REPRESENT CONTENT, AND CSS WILL ALWAYS REPRESENT THE APPEARANCE OF THAT CONTENT.

# UNDERSTANDING COMMON CSS TERMS

- IN ADDITION TO HTML TERMS, THERE ARE A FEW COMMON CSS TERMS YOU WILL WANT TO FAMILIARIZE YOURSELF WITH. THESE TERMS INCLUDE *SELECTORS*, *PROPERTIES*, AND *VALUES*. AS WITH THE HTML TERMINOLOGY, THE MORE YOU WORK WITH CSS, THE MORE THESE TERMS WILL BECOME SECOND NATURE.

# SELECTORS

- AS ELEMENTS ARE ADDED TO A WEB PAGE, THEY MAY BE STYLED USING CSS. A *SELECTOR* DESIGNATES EXACTLY WHICH ELEMENT OR ELEMENTS WITHIN OUR HTML TO TARGET AND APPLY STYLES (SUCH AS COLOR, SIZE, AND POSITION) TO. SELECTORS MAY INCLUDE A COMBINATION OF DIFFERENT QUALIFIERS TO SELECT UNIQUE ELEMENTS, ALL DEPENDING ON HOW SPECIFIC WE WISH TO BE. FOR EXAMPLE, WE MAY WANT TO SELECT EVERY PARAGRAPH ON A PAGE, OR WE MAY WANT TO SELECT ONLY ONE SPECIFIC PARAGRAPH ON A PAGE.
- SELECTORS GENERALLY TARGET AN ATTRIBUTE VALUE, SUCH AS AN ID OR CLASS VALUE, OR TARGET THE TYPE OF ELEMENT, SUCH AS <H1> OR <P>.
- WITHIN CSS, SELECTORS ARE FOLLOWED WITH CURLY BRACKETS, {}, WHICH ENCOMPASS THE STYLES TO BE APPLIED TO THE SELECTED ELEMENT. THE SELECTOR HERE IS TARGETING ALL <P> ELEMENTS.

# PROPERTIES

- ONCE AN ELEMENT IS SELECTED, A PROPERTY DETERMINES THE STYLES THAT WILL BE APPLIED TO THAT ELEMENT. PROPERTY NAMES FALL AFTER A SELECTOR, WITHIN THE CURLY BRACKETS, {}, AND IMMEDIATELY PRECEDING A COLON, :. THERE ARE NUMEROUS PROPERTIES WE CAN USE, SUCH AS BACKGROUND, COLOR, FONT-SIZE, HEIGHT, AND WIDTH, AND NEW PROPERTIES ARE OFTEN ADDED. IN THE FOLLOWING CODE, WE ARE DEFINING THE COLOR AND FONT-SIZE PROPERTIES TO BE APPLIED TO ALL <P>ELEMENTS.
- P
- {
- COLOR: ...;
- FONT-SIZE: ...;
- }

# VALUES

- SO FAR WE'VE SELECTED AN ELEMENT WITH A SELECTOR AND DETERMINED WHAT STYLE WE'D LIKE TO APPLY WITH A PROPERTY. NOW WE CAN DETERMINE THE BEHAVIOR OF THAT PROPERTY WITH A VALUE. VALUES CAN BE IDENTIFIED AS THE TEXT BETWEEN THE COLON, :, AND SEMICOLON, ;. HERE WE ARE SELECTING ALL <P> ELEMENTS AND SETTING THE VALUE OF THE COLOR PROPERTY TO BE ORANGE AND THE VALUE OF THE FONT-SIZE PROPERTY TO BE 16 PIXELS.
- P
- {
- COLOR: ORANGE;
- FONT-SIZE: 16PX;
- }

- TO REVIEW, IN CSS OUR RULE SET BEGINS WITH THE SELECTOR, WHICH IS IMMEDIATELY FOLLOWED BY CURLY BRACKETS. WITHIN THESE CURLY BRACKETS ARE DECLARATIONS CONSISTING OF PROPERTY AND VALUE PAIRS. EACH DECLARATION BEGINS WITH A PROPERTY, WHICH IS FOLLOWED BY A COLON, THE PROPERTY VALUE, AND FINALLY A SEMICOLON.
- IT IS A COMMON PRACTICE TO INDENT PROPERTY AND VALUE PAIRS WITHIN THE CURLY BRACKETS. AS WITH HTML, THESE INDENTATIONS HELP KEEP OUR CODE ORGANIZED AND LEGIBLE.

# WORKING WITH SELECTORS

- SELECTORS, AS PREVIOUSLY MENTIONED, INDICATE WHICH HTML ELEMENTS ARE BEING STYLED. IT IS IMPORTANT TO FULLY UNDERSTAND HOW TO USE SELECTORS AND HOW THEY CAN BE LEVERAGED. THE FIRST STEP IS TO BECOME FAMILIAR WITH THE DIFFERENT TYPES OF SELECTORS. WE'LL START WITH THE MOST COMMON SELECTORS: *TYPE*, *CLASS*, AND *ID* SELECTORS.

# TYPE SELECTORS

- *TYPE* SELECTORS TARGET ELEMENTS BY THEIR ELEMENT TYPE. FOR EXAMPLE, SHOULD WE WISH TO TARGET ALL DIVISION ELEMENTS, <DIV>, WE WOULD USE A TYPE SELECTOR OF DIV. THE FOLLOWING CODE SHOWS A TYPE SELECTOR FOR DIVISION ELEMENTS AS WELL AS THE CORRESPONDING HTML IT SELECTS.
- CSS
- **DIV** { … }
- HTML
- **<DIV>**…**</DIV>**
- **<DIV>**…**</DIV>**

# CLASS SELECTORS

- *CLASS* SELECTORS ALLOW US TO SELECT AN ELEMENT BASED ON THE ELEMENT'S CLASS ATTRIBUTE VALUE. CLASS SELECTORS ARE A LITTLE MORE SPECIFIC THAN TYPE SELECTORS, AS THEY SELECT A PARTICULAR GROUP OF ELEMENTS RATHER THAN ALL ELEMENTS OF ONE TYPE.
- CLASS SELECTORS ALLOW US TO APPLY THE SAME STYLES TO DIFFERENT ELEMENTS AT ONCE BY USING THE SAME CLASSATTRIBUTE VALUE ACROSS MULTIPLE ELEMENTS.
- WITHIN CSS, CLASSES ARE DENOTED BY A LEADING PERIOD, ., FOLLOWED BY THE CLASS ATTRIBUTE VALUE. HERE THE CLASS SELECTOR WILL SELECT ANY ELEMENT CONTAINING THE CLASS ATTRIBUTE VALUE OF AWESOME, INCLUDING BOTH DIVISION AND PARAGRAPH ELEMENTS.
- CSS
- .AWESOME { … }
- HTML
- <DIV CLASS="AWESOME">…</DIV>
- <P CLASS="AWESOME">…</P>

# ID SELECTORS

- *ID* SELECTORS ARE EVEN MORE PRECISE THAN CLASS SELECTORS, AS THEY TARGET ONLY ONE UNIQUE ELEMENT AT A TIME. JUST AS CLASS SELECTORS USE AN ELEMENT'S CLASS ATTRIBUTE VALUE AS THE SELECTOR, ID SELECTORS USE AN ELEMENT'S ID ATTRIBUTE VALUE AS A SELECTOR.
- REGARDLESS OF WHICH TYPE OF ELEMENT THEY APPEAR ON, ID ATTRIBUTE VALUES CAN ONLY BE USED ONCE PER PAGE. IF USED THEY SHOULD BE RESERVED FOR SIGNIFICANT ELEMENTS.
- WITHIN CSS, ID SELECTORS ARE DENOTED BY A LEADING HASH SIGN, #, FOLLOWED BY THE ID ATTRIBUTE VALUE. HERE THE ID SELECTOR WILL ONLY SELECT THE ELEMENT CONTAINING THE ID ATTRIBUTE VALUE OF SHAYHOWE.
- CSS
- #SHAYHOWE { … }
- HTML
- <DIV ID="SHAYHOWE">…</DIV>

# ADDITIONAL SELECTORS

- SELECTORS ARE EXTREMELY POWERFUL, AND THE SELECTORS OUTLINED HERE ARE THE MOST COMMON SELECTORS WE'LL COME ACROSS. THESE SELECTORS ARE ALSO ONLY THE BEGINNING. MANY MORE ADVANCED SELECTORS EXIST AND ARE READILY AVAILABLE. WHEN YOU FEEL COMFORTABLE WITH THESE SELECTORS, DON'T BE AFRAID TO LOOK INTO SOME OF THE MORE ADVANCED SELECTORS.
- ALL RIGHT, EVERYTHING IS STARTING TO COME TOGETHER. WE ADD ELEMENTS TO A PAGE INSIDE OUR HTML, AND WE CAN THEN SELECT THOSE ELEMENTS AND APPLY STYLES TO THEM USING CSS. NOW LET'S CONNECT THE DOTS BETWEEN OUR HTML AND CSS, AND GET THESE TWO LANGUAGES WORKING TOGETHER.

# REFERENCING CSS

- IN ORDER TO GET OUR CSS TALKING TO OUR HTML, WE NEED TO REFERENCE OUR CSS FILE WITHIN OUR HTML. THE BEST PRACTICE FOR REFERENCING OUR CSS IS TO INCLUDE ALL OF OUR STYLES IN A SINGLE EXTERNAL STYLE SHEET, WHICH IS REFERENCED FROM WITHIN THE <HEAD>ELEMENT OF OUR HTML DOCUMENT. USING A SINGLE EXTERNAL STYLE SHEET ALLOWS US TO USE THE SAME STYLES ACROSS AN ENTIRE WEBSITE AND QUICKLY MAKE CHANGES SITEWIDE.
- **OTHER OPTIONS FOR ADDING CSS**
- OTHER OPTIONS FOR REFERENCING CSS INCLUDE USING INTERNAL AND INLINE STYLES. YOU MAY COME ACROSS THESE OPTIONS IN THE WILD, BUT THEY ARE GENERALLY FROWNED UPON, AS THEY MAKE UPDATING WEBSITES CUMBERSOME AND UNWIELDY.

- TO CREATE OUR EXTERNAL CSS STYLE SHEET, WE'LL WANT TO USE OUR TEXT EDITOR OF CHOICE AGAIN TO CREATE A NEW PLAIN TEXT FILE WITH A .CSS FILE EXTENSION. OUR CSS FILE SHOULD BE SAVED WITHIN THE SAME FOLDER, OR A SUBFOLDER, WHERE OUR HTML FILE IS LOCATED.
- WITHIN THE <HEAD> ELEMENT OF THE HTML DOCUMENT, THE <LINK> ELEMENT IS USED TO DEFINE THE RELATIONSHIP BETWEEN THE HTML FILE AND THE CSS FILE. BECAUSE WE ARE LINKING TO CSS, WE USE THE RELATTRIBUTE WITH A VALUE OF STYLESHEET TO SPECIFY THEIR RELATIONSHIP. FURTHERMORE, THE HREF (OR HYPERLINK REFERENCE) ATTRIBUTE IS USED TO IDENTIFY THE LOCATION, OR PATH, OF THE CSS FILE.

- CONSIDER THE FOLLOWING EXAMPLE OF AN HTML DOCUMENT **<HEAD>** ELEMENT THAT REFERENCES A SINGLE EXTERNAL STYLE SHEET.
- <HEAD>
- <LINK REL="STYLESHEET" HREF="MAIN.CSS">
- </HEAD>

- IN ORDER FOR THE CSS TO RENDER CORRECTLY, THE PATH OF THE HREF ATTRIBUTE VALUE MUST DIRECTLY CORRELATE TO WHERE OUR CSS FILE IS SAVED. IN THE PRECEDING EXAMPLE, THE MAIN.CSS FILE IS STORED WITHIN THE SAME LOCATION AS THE HTML FILE, ALSO KNOWN AS THE ROOT DIRECTORY.
- IF OUR CSS FILE IS WITHIN A SUBDIRECTORY OR SUBFOLDER, THE HREF ATTRIBUTE VALUE NEEDS TO CORRELATE TO THIS PATH ACCORDINGLY. FOR EXAMPLE, IF OUR MAIN.CSS FILE WERE STORED WITHIN A SUBDIRECTORY NAMED STYLESHEETS, THE HREF ATTRIBUTE VALUE WOULD BE STYLESHEETS/MAIN.CSS, USING A FORWARD SLASH TO INDICATE MOVING INTO A SUBDIRECTORY.

# PROPERTIES OF TAGS

- IN ADDITION TO TAGS AND CONTENT, AN ELEMENT CAN HAVE ATTRIBUTES AND EVENTS. WHILE ATTRIBUTES DEFINE VALUES OR PROPERTIES TO BE USED BY BROWSERS IN THE PROCESSING OF THE DOCUMENT, EVENTS CAN BE EMPLOYED TO SPECIFY BEHAVIORS OR ACTIONS TO BE PERFORMED WHEN CERTAIN CONDITIONS ARE MET, LIKE FOR EXAMPLE, WHEN THE USERS CLICKS THE ELEMENT.
- ATTRIBUTES AND EVENTS SHARE A COMMON SYNTAX: THEY MUST BE INSERTED AS A LIST OF SPACE-SEPARATED ITEMS INSIDE THE STAR TAG, AFTER THE ELEMENT'S NAME AND PRECEDED BY A SPACE. EACH ONE OF THESE ITEMS IS COMPOSED BY A NAME (FOR THE ATTRIBUTE OR EVENT), THE EQUAL SIGN ("=") AND THE VALUE OR FUNCTION (SOMETIMES OPTIONALLY) ENCLOSED BY QUOTES. THE FOLLOWING EXAMPLE SHOWS A B ELEMENT WITH ONE ATTRIBUTE (STYLE) AND ONE EVENT (ONCLICK).
- <B STYLE="COLOR: RED“
- ONCLICK="CHANGECOLOR(THIS.PARENTNODE)">BOLD TEXT</B>

- THE CONTENT OF AN ELEMENT IS, GENERALLY SPEAKING, WHATEVER FALLS INBETWEEN ITS OPENING AND CLOSING TAGS. DEPENDING ON THE ELEMENT, THIS CAN GO FROM ABSOLUTELY NOTHING TO A PIECE OF HTML DOCUMENT. THIS CONTENT IS WHAT WILL BE AFFECTED BY THE ELEMENT'S FUNCTIONALITY OR MEANING. FOR EXAMPLE, THE EM ELEMENT GRANTS ITS CONTENT EMPHASIS, AND BROWSERS USUALLY DISPLAY ITS TEXT WITH A PARTICULAR FONT STYLE TO MAKE IT STAND OUT FROM REGULAR TEXT.
- SOME ELEMENTS, KNOWN AS EMPTY ELEMENTS, AREN'T ALLOWED TO HAVE CONTENT AND THEIR DECLARATION CONSIST ONLY OF THE OPENING TAG WITH ANY NUMBER OF ATTRIBUTES AND EVENTS.
- IN THE FOLLOWING EXAMPLE ARE THREE ELEMENTS EACH CONTAINING A DIFFERENT TYPE OF CONTENT: THE PARAGRAPH (P ELEMENT), COINTAINING OTHER ELEMENTS; A WORD WITH EMPHASIS (EM ELEMENT), CONTAINING PLAIN TEXT; AND A BUTTON (INPUT (TYPE=BUTTON) ELEMENT), WHICH IS AN EMPTY ELEMENT.
- <P>IS THIS A <EM>RAINY</EM> DAY?
- <INPUT TYPE="BUTTON" VALUE="YES"></P>

# GENERAL SYNTAX OF PROPERTY VALUE

- PROPERTY NAME: VAL 1 VAL 2 ..... VAL N
- A COLON SEPARATES THE NAME FROM THE VALUE AND A SPACE MUST SEPARATE EACH OF THE ALTERNATIVE VALUES AND THE END IS MARKED WITH A SEMICOLON.
- EG:
- <H2 STYLE="COLOR:PINK;"> EXAMPLE</H2>