

UNIT II

WEBSITES BASICS, HTML 5, CSS 3, WEB 2.0

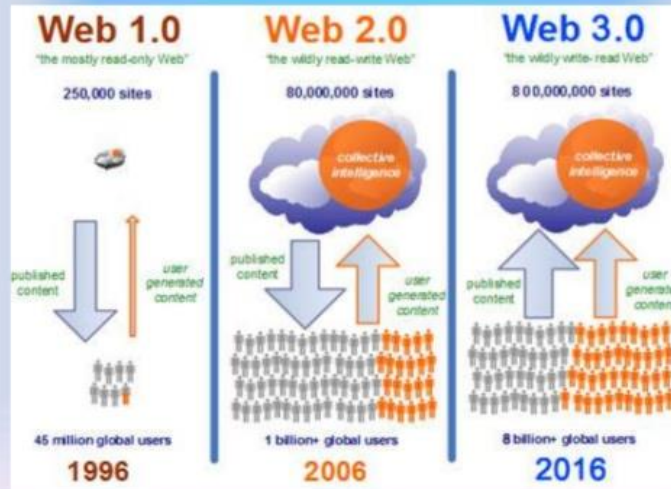
Web 2.0: Basics-RIA Rich Internet Applications – Collaborations tools – Understanding websites and web servers: Understanding Internet – Difference between websites and web server- Internet technologies Overview –Understanding the difference between internet and intranet; HTML and CSS: HTML 5.0 , XHTML, CSS 3.

2.1 Web 2.0:

Changing the trends in the use of WWW Tech and Web Design that aim to enhance the Creativity , Secure Information Sharing ,Increase Collaboration improve the functionality of the web as we know it web.10 .Web 2.0 Services are replace Desktop Computing

- Web 1.0 focused on a relatively small number of companies and advertisers producing content = “brochure web”
- Web 2.0 *involves* the user. Web 1.0 is as a *lecture* and Web 2.0 is a *conversation*.
- Many Web 2.0 companies are built almost entirely on user-generated content and harnessing collective intelligence.
 - Google, MySpace, Flickr, YouTube and Wikipedia, users create the content, while the sites provide the platforms.
 - The user is not only contributing content and developing open source software, but directing how media is delivered, and deciding which news and information outlets you trust.
 - Social bookmarking sites such as del.icio.us and ma.gnolia
 - Social media sites such as Digg or Reddit
 - Social networking, MySpace, Facebook, Bebo, LinkedIn, and Second Life
- Web 2.0 businesses leverage the Long Tail
 - Long Tail = economic model in which the market for non-hits could be significant and sometimes even greater than the market for big hits
- Web development technologies = Ajax, XHTML, Cascading Style Sheets, JavaScript, the Document Object Model, XML and the XMLHttpRequest object and popular Ajax toolkits—Dojo and Script.aculo.us
- Rich Internet Applications (RIAs)—web applications that offer the responsiveness and rich GUI features of desktop applications
- Key tools for building RIAs = Adobe’s Flex, Microsoft’s Silverlight, ASP.NET Ajax and Sun’s JavaServer Faces
- Other Web development tools and technologies = Adobe’s Dreamweaver, JSON, the web servers IIS and Apache, MySQL, PHP and ASP.NET
- Web services allow you to incorporate functionality from existing applications into your own applications quickly and easily.
- Amazon Web Services ,Google Maps web services ,eBay web services

Web 1.0 Vs. Web 2.0 Vs. Web 3.0



14

Ola Sayed Ahmed

2.1.1 Web 2.0 Features:

- **Search**- Finding information through keyword search
- **Links** - Connects information sources together using the model of the Web.
- **Authoring** - The ability to create and update content leads to the collaborative work of many authors. Wiki users may extend, undo, redo and edit each other's work. Comment systems allow readers to contribute their viewpoints.
- **Tags** - Categorization of content by users adding "tags" — short, usually one-word or two word descriptions — to facilitate searching. For example, a user can tag a metal song as "death metal". Collections of tags created by many users within a single system may be referred to as "folksonomies" (i.e., folk taxonomies).
- **Extension** - Software that makes the Web an application platform as well as a document server. Examples include Adobe Reader, Adobe Flash, Microsoft Silverlight, ActiveX, Oracle Java, QuickTime, and Windows Media.
- **Signals** - The use of syndication technology, such as RSS feeds to notify users of content changes.



Web 1.0 → Web 2.0

Read only
Web as reading platform
Developer authorship
Individual intelligence
Software applications
Commercial/proprietary
Static
Impersonal
Restricted collaboration
Short tail
Official releases
Text-based
HD as storage platform
Lecture

Read/Write/Collaborate
Web as publishing platform
Public authorship
Collective intelligence
Web as software platform
Open source/shared
Dynamic
It knows you & your needs
Collaborative
Long tail
Constantly versioning
Multimodal
Web as storage platform
Conversation

from <http://inductionworkshop.wikispaces.com> joycevalenza

2.1.2 Search

- “Content is King” Search engines are the primary tools people use to find information on the web Traffic to the major search engines is growing rapidly – Americans conducted 8 billion search queries in June 2007, up 26% from the previous year.
- Attention economy = constant flow of information in today’s world causes attention to continually be diverted
 - Search engines have gained popularity by helping users quickly find and filter the information
- *Google Search*
- Google is the leading search and online advertising company
 - founded by Larry Page and Sergey Brin
 - Google’s success in search is largely based on its PageRank algorithm and its unique infrastructure of servers
 - Google offers specialty search engines for images, news, videos, blogs and more.
 - Google web services → build Google Maps and other Google services into your applications
 - AdWords, Google’s pay-per-click (PPC) contextual advertising program
 - AdSense is Google’s advertising program for publishers
- Yahoo!
- MSN ,Location-Based Search

2.1.3 Content Networks

- Content networks = websites or collections of websites that provide information in various forms
- articles, wikis, blogs, etc
- filters the vast amounts of information on the Internet

2.1.4 Blogging

- *History of Blogging*
 - Blogs are websites consisting of entries listed in reverse chronological order
 - Grown exponentially in recent years because of easy-to-use blogging software and increasingly economical Internet access
 - Blogs can also now incorporate media, such as music or videos
 - Xanga or LiveJournal
- *Blog Components*
 - Reader comments
 - Trackbacks
 - Blogroll
- *Blogging and Journalism*
 - Encouraged citizen journalism
 - Significant news resource
 - Many bloggers are recognized as members of the media

2.1.5 Social Networking

- Social networking sites
 - Allow users to keep track of their existing interpersonal relationships and form new ones
- *Network Effects*
 - Increased value of a network as its number of users grows
 - Example = eBay—the more buyers and sellers that use the site, the more valuable the site becomes to its users
 - Set the user preferences to default to share content so users will automatically contribute to the value of the network
- *Face book*
 - Hitwise named Facebook the “preferred network among college students
 - Facebook held an 85% market share of four-year U.S. universities and had over 31 million users
- *LinkedIn*
 - Business-oriented social networking site
 - stay in touch with professional contacts
 - network with new contacts
 - check references
 - find a job or a potential employee
 - privacy concerns are more
 - *Xing* -Xing is a professional networking site based out of Germany and popular in Europe

2.2 Rich Internet Applications (RIAs)

- Rich Internet Applications (RIAs)
 - Web applications that offer the responsiveness, “rich” features and functionality approaching that of desktop applications
- *Ajax*
 - Asynchronous JavaScript and XML
 - Allows partial page
 - Creates a more responsive GUI, allowing users to continue interacting with the page as the server processes requests
 - Technologies that make up Ajax—XHTML, CSS, JavaScript, the DOM, XML, and the XMLHttpRequest object
- *Dojo*
 - Dojo is an open source JavaScript
- *Flex*
 - RIA framework that allows you to build scalable, cross-platform, multimedia-rich applications that can be delivered over the Internet
- *Silverlight*
 - Microsoft app formerly known as Windows Presentation Foundation Everywhere (WPF/E)
 - Competitor to Flex and Flash
 - Uses a compact version of the .NET framework
 - User interfaces built in Extensible Application Markup Language (XAML)—Microsoft’s XML-based format for describing user interfaces
- *JavaFX*
 - Sun Microsystems’ counterpart to Flex and Silverlight
 - Consists of the JavaFX Script and JavaFX Mobile (for mobile devices)
- *Ruby on Rails*
 - Open source framework based on the Ruby scripting language that allows you to build database-intensive applications quickly, easily, and with less code
- *Script.aculo.us*
 - Library for creating “eye candy” effects
 - Built on the Prototype JavaScript framework
 - Encapsulates the DOM and provides cross-browser processing capabilities
 - Core effects include opacity, scale, morph, move, highlight and parallel
- *JavaServer Faces*
 - Java-based web application framework
 - Separates design elements from business logic and provides a set of user-interface components (JSF components) that make developing RIAs simple
- *ASP.NET Ajax*
 - Extension of the .NET framework for creating Ajax-enabled applications
- *Adobe Integrated Runtime and Google Gears*
 - AIR allows users to run Flex web applications on their desktops even when they are *not* connected to the Internet
 - Google Gears allows use of web applications while offline

2.3 Collaborations Tools Web Services, Mashups, Widgets and Gadgets

- Incorporating web services into new programs allows people to develop new applications quickly
- *APIs*
- Provide applications with access to external services and databases
 - Examples: Sun's Java API and Web Services APIs
- *Mashups*
 - Combine content or functionality from existing web services, websites and RSS feeds to serve a new purpose
 - Housingmaps.com
 - Yahoo! Pipes
- *Widgets and Gadgets*
 - Mini applications designed to run either as stand-alone applications or as add-on features in web pages
 - Personalize your Internet experience by displaying real-time weather conditions, aggregating RSS feeds, viewing maps, receiving event reminders, providing easy access to search engines and more.
- *Amazon Web Services*
 - Amazon is a leading provider of web services
- *REST (Representational State Transfer)-Based Web Services*
 - Architectural style for implementing web services
 - Identified by a unique URL
 - RESTful web services are alternatives to those implemented with SOAP
- *Location-Based Services (LBS)*
 - Applications that take your geographic location (city, state, location of your mobile device, etc.) into consideration
- *Global Positioning System (GPS)*
 - Local search
- *Global Positioning System (GPS)*
 - Uses numerous satellites that send signals to a GPS receiver to determine its exact location.
- *Mapping Services*
 - Google Maps is one of the most popular mapping applications available online.
 - Google Earth provides satellite images of virtually any location on the planet
 - MapQuest provides similar mapping services
 - Additional mapping services include Yahoo! Local Maps and MSN Live Search
 - Companies such as NAVTEQ and Tele Atlas provide digital map data for in-vehicle and portable navigation devices, websites, location-based services and more
- *GeoRSS and Geotagging*
 - Set of standards for representing geographical information in an RSS feed (GeoRSS)
 - Geotagging can be used to add location information (longitude, latitude, etc.) to websites, images, RSS feeds, videos and more

- *XML*
 - Extensible Markup Language that is a markup language that allows you to label data based on its meaning
 - Describes data in a way that is meaningful to both humans and computers
 - Document Type Definition (DTD) or a schema, which defines the structure for the document
 - XML Vocabularies
 - XHTML for web content
 - CML for chemistry
 - MathML for mathematical content and formulas
 - XBRL for financial data
- *RSS and Atom*
 - Sites that offer RSS and Atom feeds can maintain an “open connection” with their readers
 - Most major web browsers support RSS and Atom feeds
- *JSON*
 - JavaScript Object Notation (JSON)
 - Text-based data interchange format used to represent JavaScript objects as strings and transmit them over a network
 - Commonly used in Ajax applications
- *VoIP*
 - Voice over Internet Protocol (VoIP) is the technology used to make free or inexpensive phone calls over the Internet.

Understanding websites and web servers

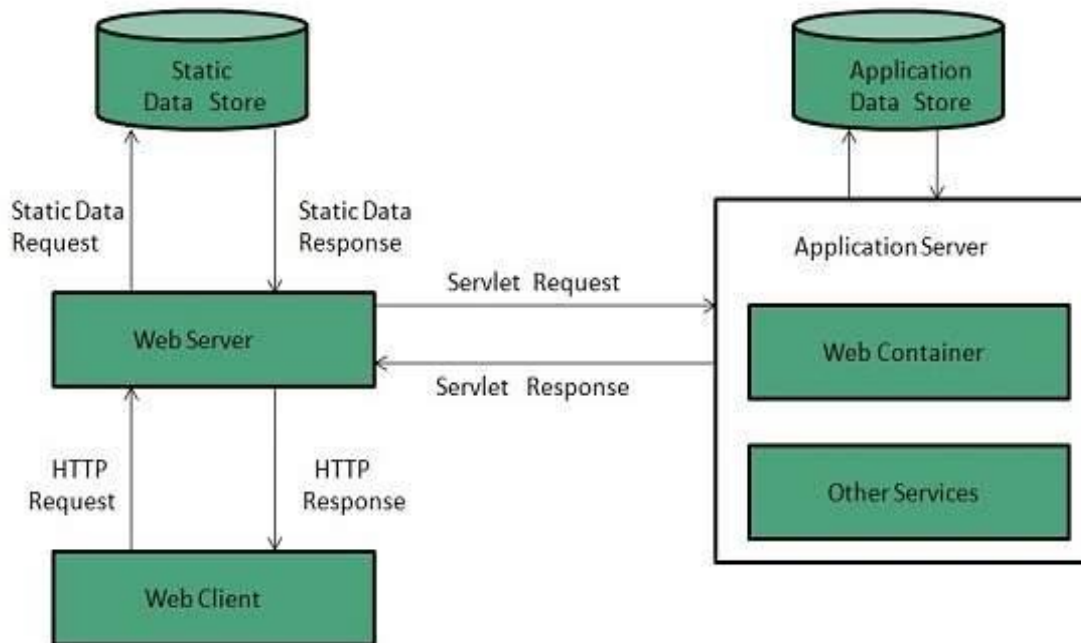
- Program that understands the HTTP protocol and generates appropriate responses
 - Clients “connect” to the machine
 - Clients send a “request”
 - Server reads request, generates “response”
 - Client interprets response appropriately

2.4 Web Server Working

Web server is a computer where the web content is stored. Basically web server is used to host the web sites but there exists other web servers also such as gaming, storage, FTP, email etc.

Web server respond to the client request in either of the following two ways:

- Sending the file to the client associated with the requested URL.
- Generating response by invoking a script and communicating with database



- When client sends request for a web page, the web server search for the requested page if requested page is found then it will send it to client with an HTTP response.
- If the requested web page is not found, web server will the send an **HTTP response:Error 404 Not found.**
- If client has requested for some other resources then the web server will contact to the application server and data store to construct the HTTP response.

Example

1. Apache HTTP Server
2. Internet Information Services (IIS)
3. Sun Java System Web Server
4. Jigsaw Server

2.5 Web Browser (Client)

Web Browser is application software that allows us to view and explore information on the web. User can request for any web page by just entering a URL into address bar.

Web browser can show text, audio, video, animation and more. It is the responsibility of a web browser to interpret text and commands contained in the web page.

Earlier the web browsers were text-based while now days graphical-based or voice-based web browsers are also available. Following are the most common web browser available today:

Browser	Vendor
Internet Explorer	Microsoft
Google Chrome	Google
Mozilla Firefox	Mozilla
Netscape Navigator	Netscape Communications Corp.
Opera	Opera Software
Safari	Apple
Sea Monkey	Mozilla Foundation
K-meleon	K-meleon

Architecture

There is a lot of web browser available in the market. All of them interpret and display information on the screen however their capabilities and structure varies depending upon implementation. But the most basic component that all web browser must exhibit are listed below:

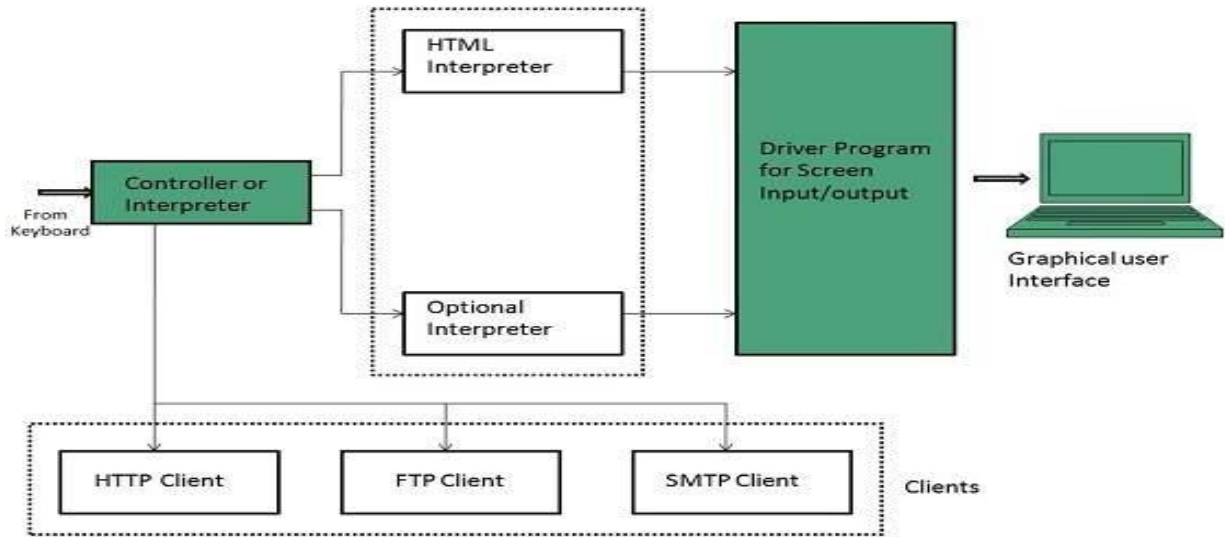
- Controller/Dispatcher
- Interpreter
- Client Programs

Controller works as a control unit in CPU. It takes input from the keyboard or mouse, interpret it and make other services to work on the basis of input it receives.

Interpreter receives the information from the controller and executes the instruction line by line. Some interpreters are mandatory while some are optional For example, HTML interpreter program is mandatory and java interpreter is optional.

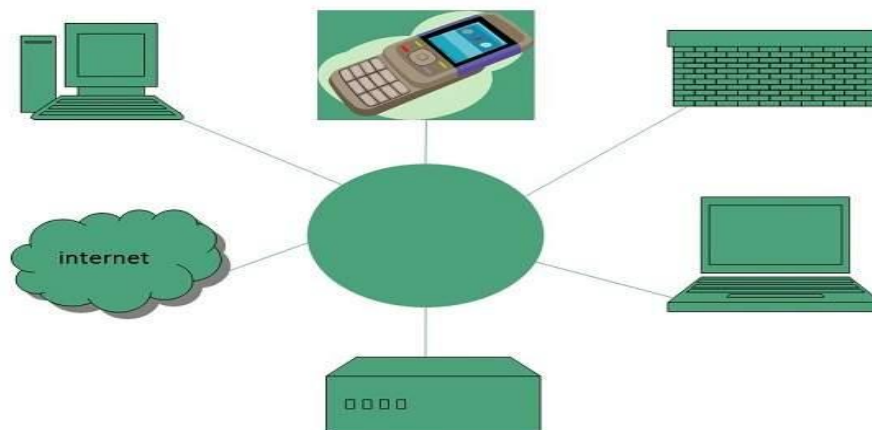
Client Program describes the specific protocol that will be used to access a particular service. Following are the client programs tat are commonly used:

- HTTP,SMTP,FTP,NNTP,POP



Internet

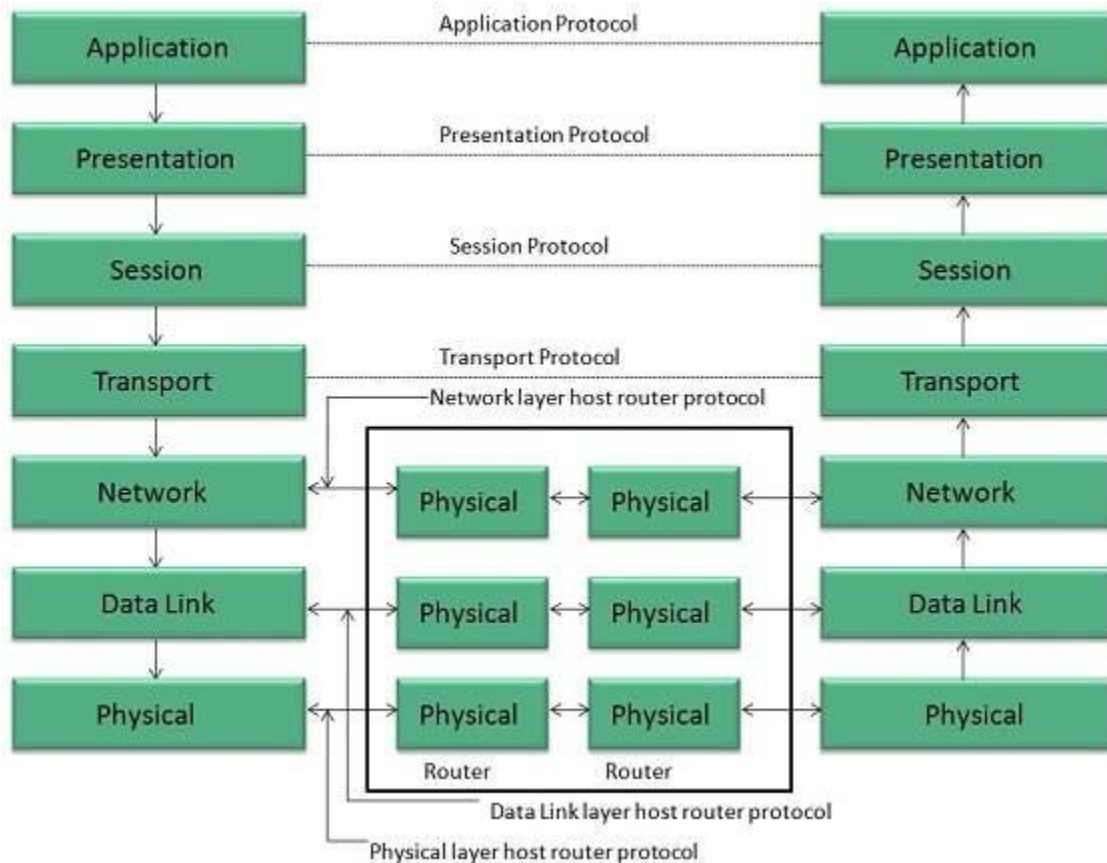
- Internet is a world-wide global system of interconnected computer networks.
- Internet uses the standard Internet Protocol (TCP/IP).
- Every computer in internet is identified by a unique IP address.
- IP Address is a unique set of numbers (such as 110.22.33.114) which identifies a computer location.
- A special computer DNS (Domain Name Server) is used to give name to the IP Address so that user can locate a computer by a name.
- For example, a DNS server will resolve a name **http://www.tutorialspoint.com** to a particular IP address to uniquely identify the computer on which this website is hosted.
- Internet is accessible to every user all over the world.



Internet Reference Models-OSI Reference Model

OSI is acronym of **Open System Interface**. This model is developed by the **International organization of Standardization (ISO)** and therefore also referred as **ISO-OSI Model**.

The OSI model consists of seven layers as shown in the following diagram. Each layer has a specific function, however each layer provide services to the layer above.



PHYSICAL LAYER

The Physical layer is responsible for the following activities:

- Activating, maintaining and deactivating the physical connection.
- Defining voltages and data rates needed for transmission.
- Converting digital bits into electrical signal.
- Deciding whether the connection is simplex, half duplex or full duplex.

DATA LINK LAYER

The data link layer performs the following functions:

- Performs synchronization and error control for the information which is to be transmitted over the physical link.
- Enables error detection, and adds error detection bits to the data which are to be transmitted.

NETWORK LAYER

Following are the functions of Network Layer:

- To route the signals through various channels to the other end.
- To act as the network controller by deciding which route data should take.
- To divide the outgoing messages into packets and to assemble incoming packets into messages for higher levels.

TRANSPORT LAYER

The Transport layer performs the following functions:

- It decides if the data transmission should take place on parallel paths or single path.
- It performs multiplexing, splitting on the data.
- It breaks the data groups into smaller units so that they are handled more efficiently by the network layer.

The Transport Layer guarantees transmission of data from one end to other end.

SESSION LAYER

The Session layer performs the following functions:

- Manages the messages and synchronizes conversations between two different applications.
- It controls logging on and off, user identification, billing and session management.

PRESENTATION LAYER

The Presentation layer performs the following functions:

- This layer makes it sure that the information is delivered in such a form that the receiving system will understand and use it.

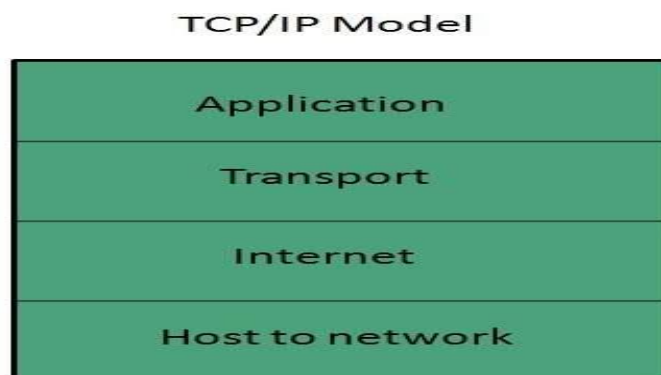
APPLICATION LAYER

The Application layer performs the following functions:

- It provides different services such as manipulation of information in several ways, retransferring the files of information, distributing the results etc.
- The functions such as LOGIN or password checking are also performed by the application layer.

TCP/IP Reference Model

TCP/IP model is practical model and is used in the Internet. **TCP/IP** is acronym of Transmission Control Protocol and Internet Protocol. The **TCP/IP** model combines the two layers (Physical and Data link layer) into one layer i.e. **Host-to-Network** layer. The following diagram shows the various layers of TCP/IP model:



APPLICATION LAYER

This layer is same as that of the OSI model and performs the following functions:

- It provides different services such as manipulation of information in several ways, retransferring the files of information, distributing the results etc.
- The functions such as LOGIN or password checking are also performed by the application layer.

Protocols used: TELNET, FTP, SMTP, DN, HTTP, NNTP are the protocols employed in this layer.

TRANSPORT LAYER

It does the same functions as that of transport layer in OSI model. Here are the key points regarding transport layer:

- It uses **TCP** and **UDP** protocol for end to end transmission.
- TCP is reliable and **connection oriented protocol**.
- TCP also handles flow control.
- The UDP is not reliable and a **connection less protocol** also does not perform flow control.

Protocols used: TCP/IP and UDP protocols are employed in this layer.

INTERNET LAYER

The function of this layer is to allow the host to insert packets into network and then make them travel independently to the destination. However, the order of receiving the packet can be different from the sequence they were sent.

Protocols used: Internet Protocol (IP) is employed in Internet layer.

HOST-TO-NETWORK LAYER

This is the lowest layer in TCP/IP model. The host has to connect to network using some protocol, so that it can send IP packets over it. This protocol varies from host to host and network to network.

Protocols used: ARPANET, SATNET, LAN, packet radio are the protocols which are used in this layer.

Domain Name System Architecture

The Domain name system comprises of **Domain Names**, **Domain Name Space**, **Name Server** that have been described below:

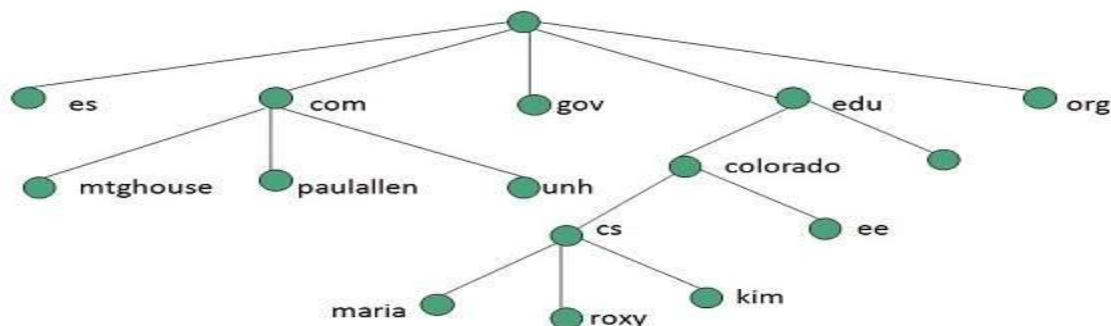
Domain Names

Domain Name is a symbolic string associated with an IP address. There are several domain names available; some of them are generic such as **com**, **edu**, **gov**, **net** etc, while some country level domain names such as **au**, **in**, **za**, **us** etc. The following table shows the **Generic Top-Level** Domain names:

Domain Name	Meaning
Com	Commercial business
Edu	Education
Gov	U.S. government agency
Int	International entity
Mil	U.S. military
Net	Networking organization
Org	Non profit organization

Domain Name Space

The domain name space refers a hierarchy in the internet naming structure. This hierarchy has multiple levels (from 0 to 127), with a root at the top. The following diagram shows the domain name space hierarchy:



DNS Working

DNS translates the domain name into IP address automatically. Following steps will take you through the steps included in domain resolution process:

1. When we type **www.Google.com** into the browser, it asks the local DNS Server for its IP address.
Here the local DNS is at ISP end.
2. When the local DNS does not find the IP address of requested domain name, it forwards the request to the root DNS server and again enquires about IP address of it.
3. The root DNS server replies with delegation that **I do not know the IP address of www.tutorialspoint.com but know the IP address of DNS Server.**
4. The local DNS server then asks the com DNS Server the same question.
5. The **com** DNS Server replies the same that it does not know the IP address of www.tutorialspoint.com but knows the address of tutorialspoint.com.
6. Then the local DNS asks the tutorialspoint.com DNS server the same question.
7. Then tutorialspoint.com DNS server replies with IP address of www.tutorialspoint.com.
8. Now, the local DNS sends the IP address of www.tutorialspoint.com to the computer that sends the request.

Internet Protocols

Transmission Control Protocol (TCP)

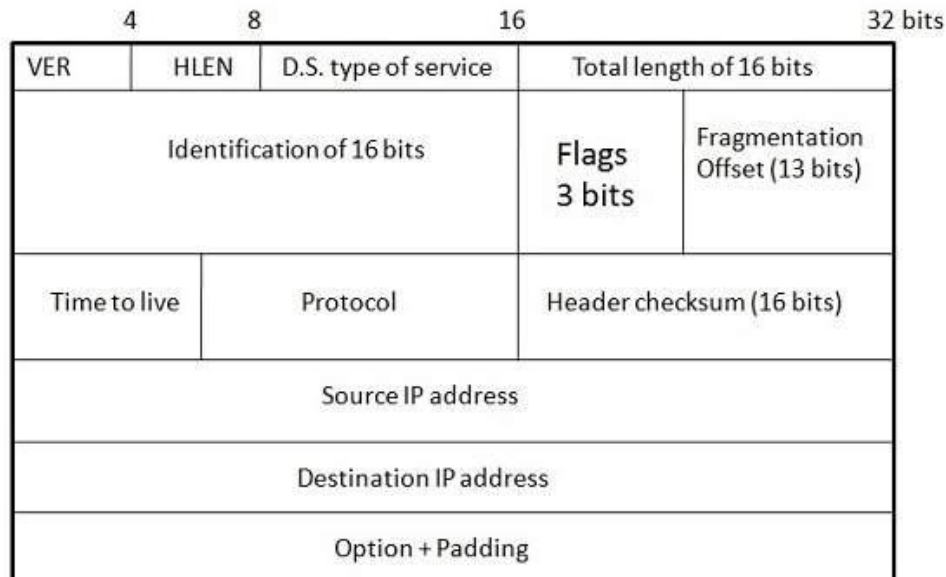
- Transmission Control Protocol (TCP) corresponds to the Transport Layer of OSI Model.
- TCP is a reliable and connection oriented protocol.
- TCP offers:
 - Stream Data Transfer.
 - Reliability.
 - Efficient Flow Control
 - Full-duplex operation.
 - Multiplexing.
- TCP offers connection oriented end-to-end packet delivery.
- TCP ensures reliability by sequencing bytes with a forwarding acknowledgement number that indicates to the destination the next byte the source expect to receive.
- It retransmits the bytes not acknowledged within specified time period.

Internet Protocol (IP)

Internet Protocol is **connectionless** and **unreliable** protocol. It ensures no guarantee of successfully transmission of data.

In order to make it reliable, it must be paired with reliable protocol such as TCP at the transport layer.

Internet protocol transmits the data in form of a datagram as shown in the following diagram:



User Datagram Protocol (UDP)

Like IP, UDP is connectionless and unreliable protocol. It doesn't require making a connection with the host to exchange data. Since UDP is unreliable protocol, there is no mechanism for ensuring that data sent is received.

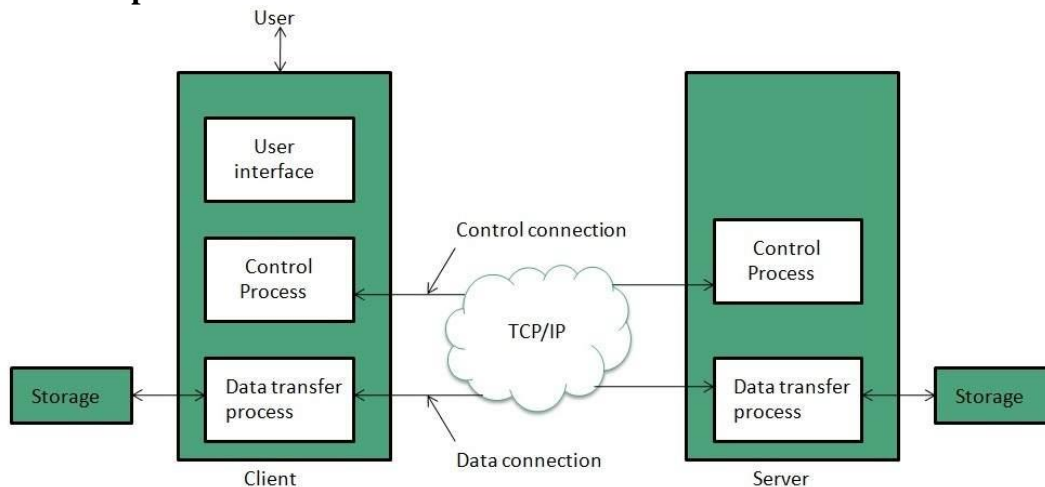
UDP transmits the data in form of a datagram. The UDP datagram consists of five parts as shown in the following diagram:



File Transfer Protocol (FTP)

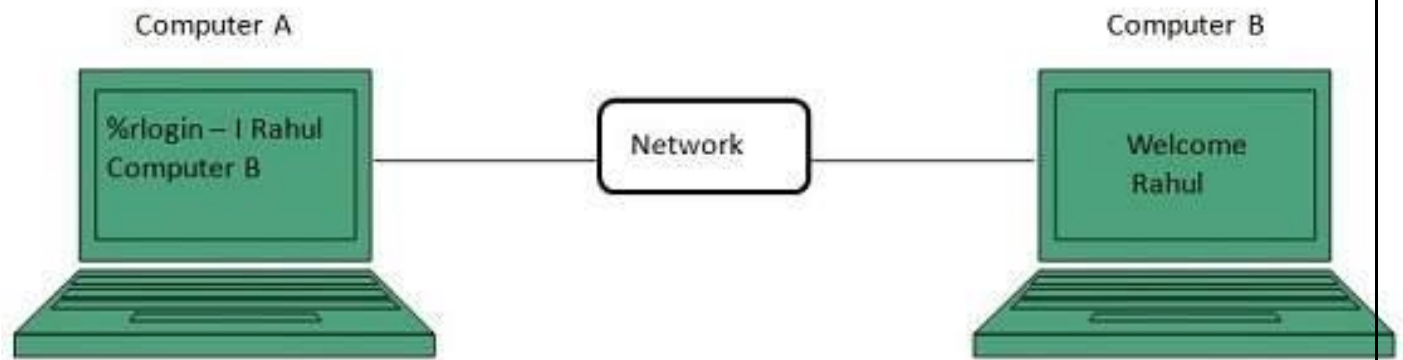
FTP is used to copy files from one host to another. FTP offers the mechanism for the same in following manner:

- FTP creates two processes such as Control Process and Data Transfer Process at both ends i.e. at client as well as at server.
- FTP establishes two different connections: one is for data transfer and other is for control information.
- **Control connection** is made between **control processes** while **Data Connection** is made between
- FTP uses **port 21** for the control connection and **Port 20** for the data connection.



Telnet

Telnet is a protocol used to log in to remote computer on the internet. There are a number of Telnet clients having user friendly user interface. The following diagram shows a person is logged in to computer A, and from there, he remote logged into computer B.



Hyper Text Transfer Protocol (HTTP)

HTTP is a communication protocol. It defines mechanism for communication between browser and the web server. It is also called request and response protocol because the communication between browser and server takes place in request and response pairs.

HTTP REQUEST

HTTP request comprises of lines which contains:

- Request line
- Header Fields
- Message body

HTTP RESPONSE

Like HTTP request, HTTP response also has certain structure. HTTP response contains:

- Status line
- Headers
- Message body

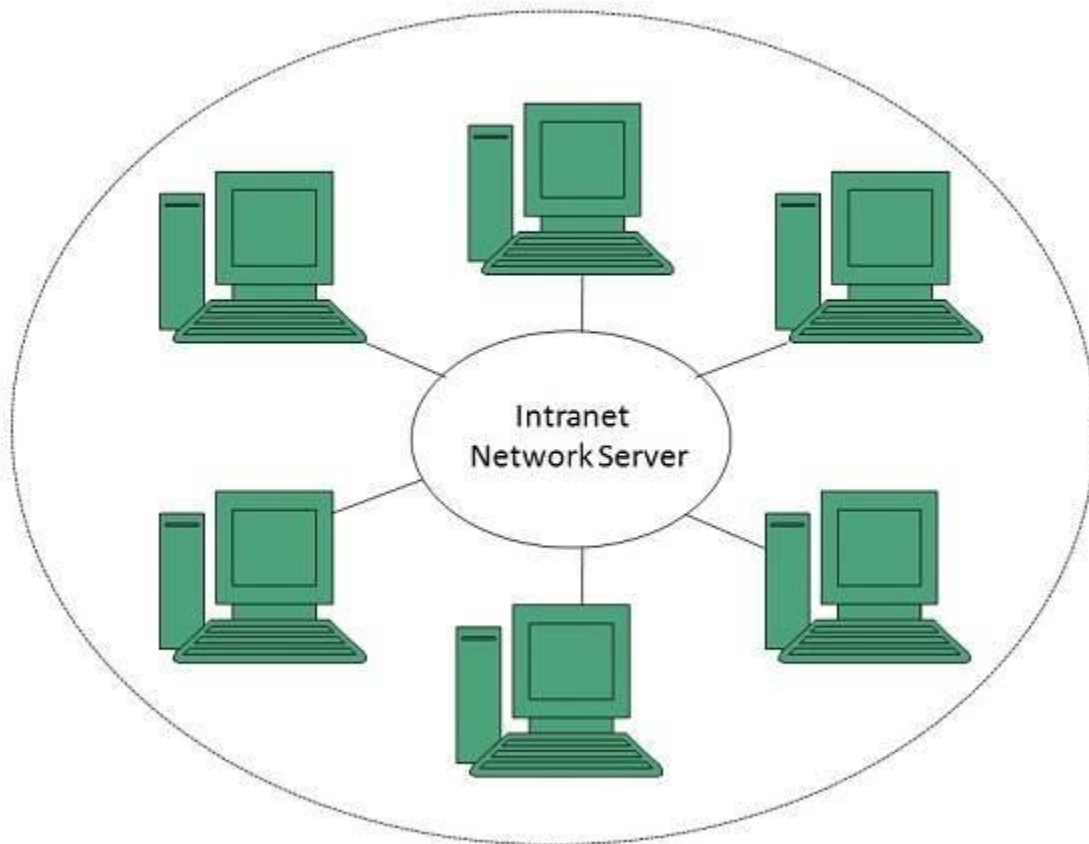
E-mail Protocols

- **SMTP** -SMTP **stands for** Simple Mail Transfer Protocol
- **IMAP**-IMAP **stands for** Internet Mail Access Protocol.
- POP stands for Post Office Protocol. It is generally used to support a single client

Intranet

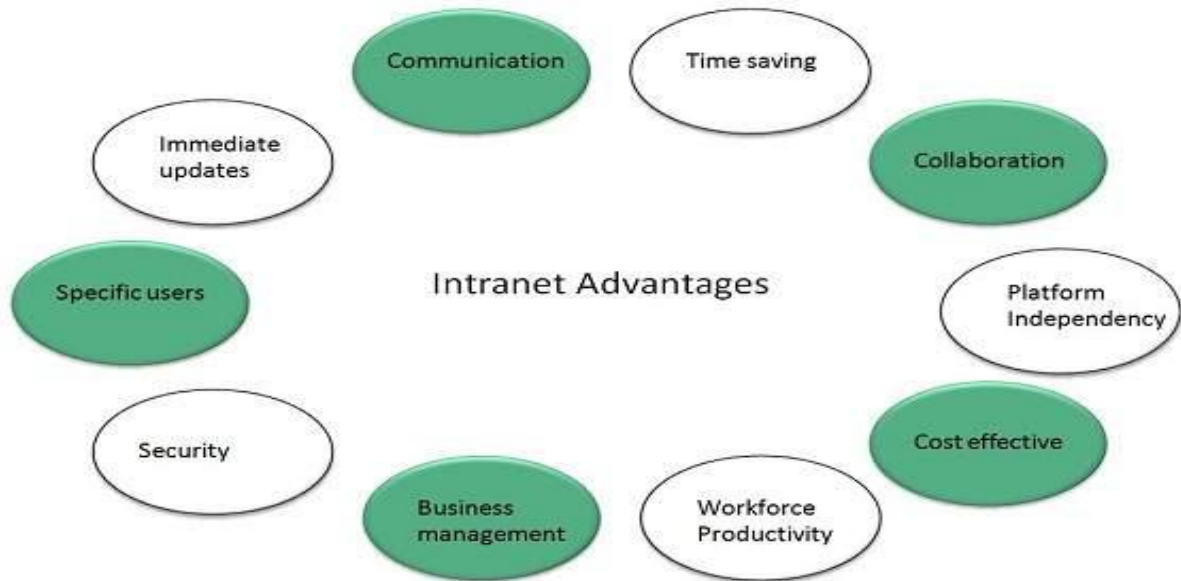
Intranet is defined as private network of computers within an organization with its own server and firewall. Moreover we can define Intranet as:

- Intranet is system in which multiple PCs are networked to be connected to each other. PCs in intranet are not available to the world outside of the intranet.
- Usually each company or organization has their own Intranet network and members/employees of that company can access the computers in their intranet.
- Every computer in internet is identified by a unique IP address.
- Each computer in Intranet is also identified by a IP Address, which is unique among the computers in that Intranet.



Benefits

Intranet is very efficient and reliable network system for any organization. It is beneficial in every aspect such as collaboration, cost-effectiveness, security, productivity and much more.



Communication

Intranet offers easy and cheap communication within an organization. Employees can communicate using chat, e-mail or blogs.

Time Saving

Information on Intranet is shared in real time.

Collaboration

Information is distributed among the employees as according to requirement and it can be accessed by the authorized users, resulting in enhanced teamwork.

Platform Independency

Intranet can connect computers and other devices with different architecture.

Cost Effective

Employees can see the data and other documents using browser rather than printing them and distributing duplicate copies among the employees, which certainly decreases the cost.

Workforce Productivity

Data is available at every time and can be accessed using company workstation. This helps the employees work faster.

Business Management

It is also possible to deploy applications that support business operations.

Security

Since information shared on intranet can only be accessed within an organization, therefore there is almost no chance of being theft.

Specific Users

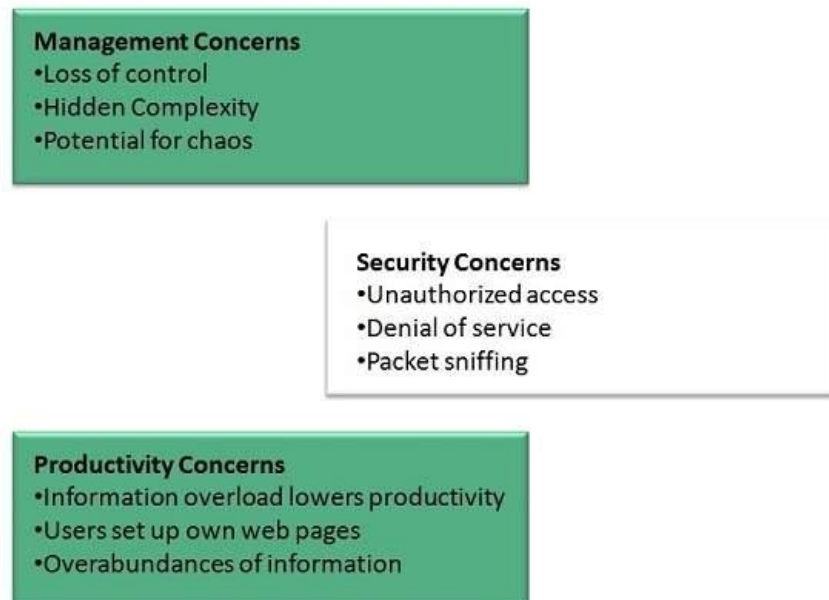
Intranet targets only specific users within an organization therefore, once can exactly know whom he is interacting.

Immediate Updates

Any changes made to information are reflected immediately to all the users.

Issues

Apart from several benefits of Intranet, there also exist some issues.. These issues are shown in the following diagram:



Applications

Intranet applications are same as that of Internet applications. Intranet applications are also accessed through a web browser. The only difference is that, Intranet applications reside on local server while Internet applications reside on remote server. Here, we've discussed some of these applications:

Document publication applications

Document publication applications allow publishing documents such as manuals, software guide, employee profits etc without use of paper.

Electronic resources applications

It offers electronic resources such as software applications, templates and tools, to be shared across the network.

Interactive Communication applications

Like on internet, we have e-mail and chat like applications for Intranet, hence offering an interactive communication among employees.

Support for Internet Applications

Intranet offers an environment to deploy and test applications before placing them on Internet.

Internet vs. Intranet

Apart from similarities there are some differences between the two. Following are the

Differences between Internet and Intranet:

Intranet	Internet
Localized Network.	Worldwide Network
Doesn't have access to Intranet	Have access to Internet.
More Expensive	Less Expensive
More Safe	Less Safe
More Reliability	Less Reliability

Extranet vs. Intranet

Extranet	Intranet
Internal network that can be accessed externally.	Internal network that cannot be accessed externally.
Extranet is extension of company's Intranet.	Only limited users of a company.
For limited external communication between customers, suppliers and business partners.	Only for communication within a company.

HTML, XHTML, HTML 5.0

Basic Understanding

What is SGML?

This is **Standard Generalized Markup Language** (SGML) application conforming to International Standard ISO 8879. HTML is widely regarded as the standard publishing language of the World Wide Web.

This is a language for describing markup languages, particularly those used in electronic document exchange, document management, and document publishing. HTML is an example of a language defined in SGML.

HTML is a programming language that allows you to tell the browser what you want it to display and how you want it to be displayed, in simple terms, it is a Webpage.

HTML there are certain markers, like commands, that tell the Browser what to do, these are called tags. By using tags you can have tables, fonts, colors, pictures, links, and almost anything you can think up, and experimentation with tags can lead to some pretty cool WebPages.

What is XML?

XML stands for **EXtensible Markup Language**. XML is a markup language much like HTML and it was designed to describe data. XML tags are not predefined. You must define your own tags according to your needs.

XHTML stands for **Extensible Hypertext Markup Language**. It is the next step in the evolution of the internet. The XHTML 1.0 is the first document type in the XHTML family.

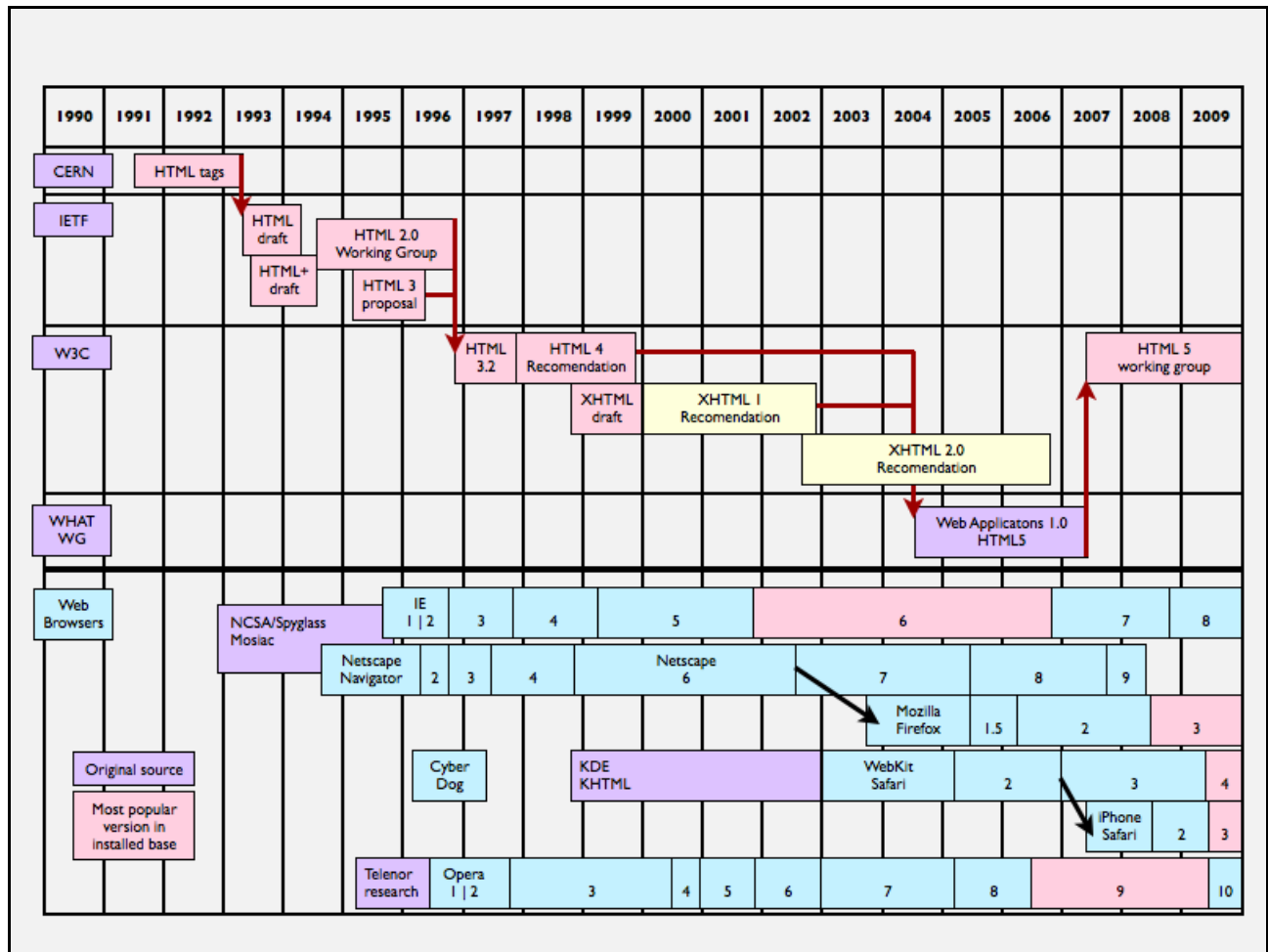
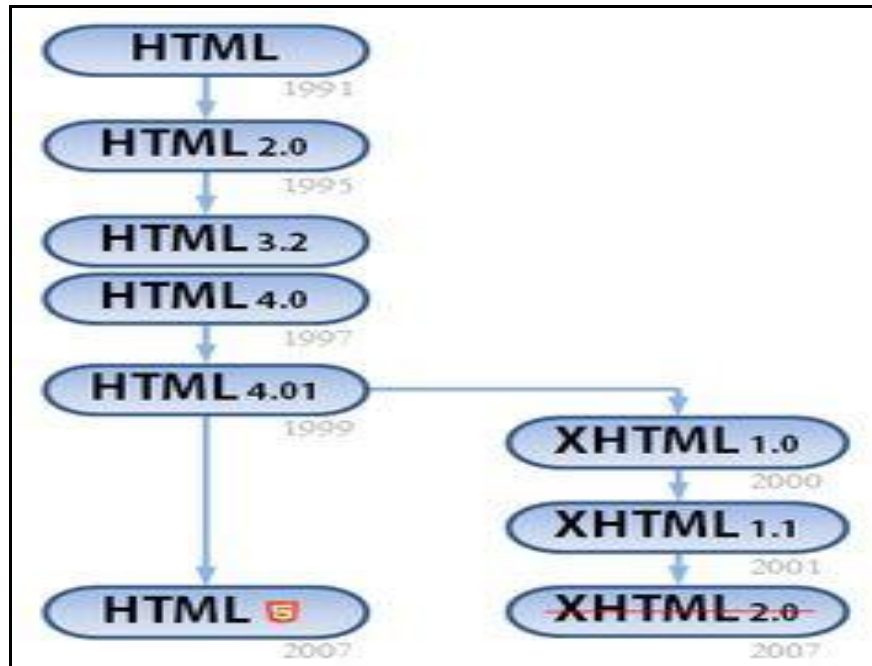
XHTML is almost identical to HTML 4.01 with only few differences. This is a cleaner and stricter version of HTML 4.01. If you already know HTML, then you need to give little attention to learn this latest version of HTML.

XHTML was developed by World Wide Web Consortium (W3C) to help web developers make the transition from HTML to XML. By migrating to XHTML today, web developers can enter the XML world with all of its benefits, while still remaining confident in the backward and future compatibility of the content.

Why Use XHTML?

Developers who migrate their content to XHTML 1.0 get the following benefits –

- XHTML documents are XML conforming as they are readily viewed, edited, and validated with standard XML tools.
- XHTML documents can be written to operate better than they did before in existing browsers as well as in new browsers.
- XHTML documents can utilize applications such as scripts and applets that rely upon either the HTML Document Object Model or the XML Document Object Model.
- XHTML gives you a more consistent, well-structured format so that your webpages can be easily parsed and processed by present and future web browsers.
- You can easily maintain, edit, convert and format your document in the long run.
- Since XHTML is an official standard of the W3C, your website becomes more compatible with many browsers and it is rendered more accurately.
- XHTML combines strength of HTML and XML. Also, XHTML pages can be rendered by all XML enabled browsers.
- XHTML defines quality standard for your webpages and if you follow that, then your web pages are counted as quality web pages. The W3C certifies those pages with their quality stamp.



XHTML Document Structure

A typical HTML document will have the following structure

```
<html>
```

```
  <head>
```

```
    <title>This is document title</title>
```

Document header related tags

```
  </head>
```

```
  <body>
```

Document body related tags

```
  </body>
```

```
</html>
```

Heading Tags

Any document starts with a heading. You can use different sizes for your headings. HTML also has six levels of headings, which use the elements <h1>, <h2>, <h3>, <h4>, <h5>, and <h6>. While displaying any heading, browser adds one line before and one line after that heading.

Example

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title>Heading Example</title>
```

```
  </head>
```

```
  <body>
```

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

```
    <h2>This is heading 2</h2>
```

```
    <h3>This is heading 3</h3>
```

```
    <h4>This is heading 4</h4>
```

```
    <h5>This is heading 5</h5>
```

```
    <h6>This is heading 6</h6>
```

```
  </body>
```

```
</html>
```

Paragraph Tag

The <p> tag offers a way to structure your text into different paragraphs. Each paragraph of text should go in between an opening <p> and a closing </p> tag as shown below in the example –

Horizontal Lines

Horizontal lines are used to visually break-up sections of a document. The <hr> tag creates a line from the current position in the document to the right margin and breaks the line accordingly.

Example

```
<html>

  <head>

    <title>Paragraph Example</title>

  </head>

  <body>

    <p>Here is a first paragraph of text.</p><hr/>

    <p>Here is a second paragraph of text.</p><hr/>

    <p>Here is a third paragraph of text.</p>

  </body>

</html>
```

Line Break Tag

Whenever you use the
 element, anything following it starts from the next line. This tag is an example of an empty element, where you do not need opening and closing tags, as there is nothing to go in between them.

Hyperlink <a>

The essence of an HTML document lies in the first two words: (H)yper(T)ext. In other words, it is the ability to link to other documents that makes HTML unique. How do HTML documents link to other documents? It does so via the <a> tag. The attributes for the <a> tag are href and name. Below we show an example for each:

Example 1: Link to an external document.

HTML:

`This link` takes you to a document called "sample.html."

[This link](#) takes you to a document called "sample.html."

Example 2: Link to an anchor within the same document.

`This link` takes you to a pre-determined location on the same page.

[This link](#) takes to a you to a pre-determined location on the same page.

Image Tags:

The `` tag is used to embed an image on the HTML document. The attributes are as follows:

- **src:** The file path to the image file.
- **width:** The width of the image, in pixels.
- **height:** The height of the image, in pixels.
- **alt:** The text the browser will display when visitors mouse over the image. Search engines often places more weight on the text in the alt attribute.

HTML Table: This section lists the tags often used with HTML tables: `<table>`, `<tr>`, and `<td>`.

`<table>`

The `<table>` tag specifies the presence of a table. This is very often used in conjunction with the `<tr>` and the `<td>` tags. The following attributes are commonly used to define the properties of this table:

- **width:** This specifies the width of the table. Can be specified in pixels or in relative terms (for example, 100%).
- **border:** This specifies whether the table will have a border. The number indicates the thickness of the border.
- **cellspacing:** The amount spacing between the cell wall and the cell border. The area enclosed by the cell walls are the maximum amount of area that text can be displayed in a cell.
- **cellpadding:** The amount padding between cells and the each cell wall in a table.
- **bgcolor:** This specifies the background color for this table. The color value may be specified as the color name or the six-character color code.

<tr>

The <tr> tag specifies the presence of a row. The following attributes are commonly used to define the properties of this row:

- bgcolor: This specifies the background color for this row. The color value may be specified as the color name or the six-character color code.
- height: This specifies the height of the row.
- rowspan: This specifies the number of rows this particular row occupies.

<td>

The <td> tag specifies the presence of a column. Columns are specified within each row. The following attributes are commonly used to define the properties of this column:

- valign: This specifies how the text is vertically aligned inside the column. Common values include "top", "middle", and "bottom."
- width: This specifies the width of the column. Can be specified in pixels or in relative terms (for example, 50%).
- bgcolor: This specifies the background color for this column. The color value may be specified as the color name or the six-character color code.
- colspan: This specifies the number of columns this particular column occupies.

Example 1

HTML:

```
<table border=1>
<tr><td>Element 1</td><td>Element 2</td><td>Element 3</td></tr>
<tr><td>Element 4</td><td>Element 5</td><td>Element 6</td></tr>
</table>
```

Display:

Element 1	Element 2	Element 3
Element 4	Element 5	Element 6

Example 2

HTML:

```
<table border=1 width=500>
<tr><td width=200>Element 1</td><td width=150>Element 2</td><td width=150>Element
3</td></tr>
<tr><td>Element 4</td><td>Element 5</td><td>Element 6</td></tr>
</table>
```

Display:

Element 1	Element 2	Element 3
Element 4	Element 5	Element 6

Example 3

HTML:

```
<table border=1 width=500>
<tr bgcolor=red><td width=200><b>Element 1</b></td><td width=150>Element 2</td><td
width=150>Element 3</td></tr>
<tr><td bgcolor=55ff55>Element 4</td><td>Element 5</td><td><i>Element 6</i></td></tr>
</table>
```

Display:

Element 1	Element 2	Element 3
Element 4	Element 5	<i>Element 6</i>

Example 4

HTML:

```
<table border=1>
<tr><td>Element 1</td><td>Element 2</td><td>Element 3</td></tr>
<tr><td colspan=2>Element 4</td><td>Element 5</td></tr>
</table>
```

Display:

Element 1	Element 2	Element 3
Element 4	Element 5	

Example 5: cellspacing and cellpadding attributes.

HTML:

```
<table border=1 cellspacing=10 cellpadding=0>
<tr><td>Element 1</td><td>Element 2</td><td>Element 3</td></tr>
```

```
<tr><td>Element 4</td><td>Element 5</td><td>Element 6</td></tr>
</table>
```

Display:

Element 1	Element 2	Element 3
Element 4	Element 5	Element 6

Example 6: cellspacing and cellpadding attributes.

HTML:

```
<table border=1 cellspacing=0 cellpadding=10>
<tr><td>Element 1</td><td>Element 2</td><td>Element 3</td></tr>
<tr><td>Element 4</td><td>Element 5</td><td>Element 6</td></tr>
</table>
```

Element 1	Element 2	Element 3
Element 4	Element 5	Element 6

Example 7: cellspacing and cellpadding attributes.

HTML:

```
<table border=1 cellspacing=10 cellpadding=10>
<tr><td>Element 1</td><td>Element 2</td><td>Element 3</td></tr>
<tr><td>Element 4</td><td>Element 5</td><td>Element 6</td></tr>
</table>
```

Element 1	Element 2	Element 3
Element 4	Element 5	Element 6

HTML Lists

There are 3 different types of lists. A tag starts an ordered list, for unordered lists, and <dl> for definition lists.

- - unordered list; *bullets*
- - ordered list; *numbers*
- <dl> - definition list; *dictionary*

HTML Ordered Lists

Use the tag to begin an ordered list. Place the (list item) tag between your opening and closing tags to create list items. Ordered simply means numbered, as the list below demonstrates.

HTML Code:

```
<h4 align="center">Goals</h4> <ol>
```

```
<li>Find a Job</li> <li>Get Money</li> <li>Move Out</li> </ol>
```

Numbered list:

Goals

1. Find a Job
2. Get Money
3. Move Out

Start your ordered list on any number besides 1 using the *start* attribute.

HTML Code:

```
<h4 align="center">Goals</h4> <ol start="4" >
```

```
<li>Buy Food</li> <li>Enroll in College</li> <li>Get a Degree</li> </ol>
```

Numbered List Start:

Goals

4. Buy Food
5. Enroll in College
6. Get a Degree

Nothing fancy here, start simply defines which number to begin numbering with. There are 4 other types of ordered lists. Instead of generic numbers you can replace them with Roman numerals or letters, both capital and lower-case. Use the *type* attribute to change the numbering.

HTML Code:

```
<ol type="a"> <ol type="A"> <ol type="i"> <ol type="I">
```

Ordered List Types:

Lower-Case Letters	Upper-Case Letters	Lower-Case Numerals	Upper-Case Numerals
a. Find a Job	A. Find a Job	i. Find a Job	I. Find a Job
b. Get Money	B. Get Money	ii. Get Money	II. Get Money
c. Move Out	C. Move Out	iii. Move Out	III. Move Out

HTML Unordered Lists

Create a bulleted list with the tag. The bullet itself comes in three flavors: squares, discs, and circles. The default bullet displayed by most web browsers is the traditional full disc.

HTML Code:

```
<h4 align="center">Shopping List</h4> <ul>
```

```
<li>Milk</li> <li>Toilet Paper</li> <li>Cereal</li> <li>Bread</li>
```

```
</ul>
```

Unordered Lists:

Shopping List

- Milk
- Toilet Paper
- Cereal
- Bread

Here's a look at the other flavors of unordered lists may look like.

HTML Code:

```
<ul type="square"> <ul type="disc"> <ul type="circle">
```

Unordered List Types:

type="square"	type="disc"	type="circle"
▪ Milk	• Milk	○ Milk
▪ Toilet Paper	• Toilet Paper	○ Toilet Paper
▪ Cereal	• Cereal	○ Cereal
▪ Bread	• Bread	○ Bread

HTML Definition Term Lists

Make definition lists as seen in dictionaries using the <dl> tag. These lists displace the term word just above the definition itself for a unique look. It's wise to bold the terms to displace them further.

- <dl> - defines the start of the list
- <dt> - definition term
- <dd> - defining definition

HTML Code:

```
<dl>
<dt><b>Fromage</b></dt>
<dd>French word for
cheese.</dd>
<dt><b>Voiture</b></dt>
<dd>French word for
car.</dd> </dt>
```

Frame Set:

- **frameset** - The parent tag that defines the characteristics of this frames page. Individual frames are defined inside it.
- **frameset cols="#%, *"** - Cols(columns) defines the width that each frame will have. In the above example we chose the menu (the 1st column) to be 30% of the total page and used a "*", which means the content (the 2nd column) will use the remaining width for itself.
- **frame src=""** - The location of the web page to load into the frame.

frameset rows="#%, *" - rows defines the height that each frame will have. In the above example we chose the new title (the 1st row) to be 20% of the total page height and used a "*", which means that menu and content (which are the 2nd row) will use the remaining height.

```
<html>
<head>
<title>HTML Frames - A Basic Frame Layout</title>
</head>
<frameset cols="25%,75%">
<frame src="menu.html" name="menu">
<frame src="welcome.html" name="content">
</frameset>
</html>
```

HTML Forms

The <form> tag is used to create an HTML form for user input.

A form can contain input elements like text fields, checkboxes, radio-buttons, submit buttons and more. A form can also contain select menus, textarea, fieldset, legend, and label elements. Forms are used to pass data to a server.

```
<form method="post" action=" ">
```

```
Name: <input type="text" size="10" maxlength="40" name="name"> <br /> Password: <input type="password" size="10" maxlength="10" name="password">
```

```
</form>
```

Common Form Controls

TABLE 2.5: HTML 4.01/XHTML 1.0 non-deprecated form controls.

Element	type Attribute	Control
input	text	Text input
input	password	Password input
input	checkbox	Checkbox
input	radio	Radio button
input	submit	Submit button
input	image	Graphical submit button
input	reset	Reset button (form clear)
input	button	Push button (for use with scripts)
input	hidden	Non-displayed control (stores server-supplied information)
input	file	File select
button	submit	Submit button with content (not an empty element)
button	reset	Cancel button with content (not an empty element)
button	button	Button with content but no predefined action
select	N/A	Menu
option	N/A	Menu item
optgroup	N/A	Heading in a hierarchical menu
textarea	N/A	Multi-line text input
label	N/A	Associate label with control(s)
fieldset	N/A	Groups controls
legend	N/A	Add caption to a fieldset

Simple Example for Registration Form with all form controls

```
<html> <head> <title>JavaScript Form Validation using a sample registration form</title>
</head> <body>

<h1>Registration Form</h1>

<p>Use tab keys to move from one input field to the next.</p>

<form name='registration'>

  User id:<input type="text" name="userid" size="12" />

  Password:<input type="password" name="passid" size="12" />

  Name:<input type="text" name="username" size="50" />

  Address:<input type="text" name="address" size="50" />

  Country:<select name="country">

    <option selected="" value="Default">(Please select a country)</option>

    <option value="AF">Australia</option> <option value="AL">Canada</option>

    <option value="DZ">India</option> <option value="AS">Russia</option>

    <option value="AD">USA</option> </select>

  ZIP Code:<input type="text" name="zip" />

  Email:<input type="text" name="email" size="50" />

  Sex:<input type="radio" name="sex" value="Male" /><span>Male</span>

    <input type="radio" name="sex" value="Female" /><span>Female</span>

  Language:<input type="checkbox" name="en" value="en" checked /><span>English</span>

    <input type="checkbox" name="nonen" value="noen" /><span>Non English</span>

  About:<textarea name="desc" id="desc"></textarea>

  <input type="submit" name="submit" value="Submit" />

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

HTML5

HTML5 Introduction

- Successor of HTML 4.01 and XHTML 1.1
- It comes with new tags, features and APIs
- Below is a non exhaustive list of features that tend to be labelled as "HTML5" in the medias:
 - New structural elements (<header>, <footer>, <nav> and more)
 - Forms 2.0 and client-side validation
 - Native browser support for audio and video (<video>, <audio>)
 - Canvas API and SVG
 - Web storage
 - Offline applications
 - Geolocation
 - Drag & Drop
 - Web Workers
 - New communications API (Server Sent Events, Web Sockets, ...)

Future of HTML5

- 84% of Developers Plan to Adopt Key HTML5 Features
- The key to understanding HTML5 is that it is not one, but a group of technologies. Within HTML5, developers have a tremendous amount of choice regarding what they use and what they don't use
- The power of HTML5 being ready for prime-time can be seen in Microsoft's choice to utilize it in Windows 8

New and Updated HTML5 Elements

HTML5 introduces 28 new elements:

<section>, <article>, <aside>, <hgroup>, <header>, <footer>, <nav>, <figure>, <figcaption>, <video>, <audio>, <source>, <embed>, <mark>, <progress>, <meter>, <time>, <ruby>, <rt>, <rp>, <wbr>, <canvas>, <command>, <details>, <summary>, <datalist>, <keygen> and <output>

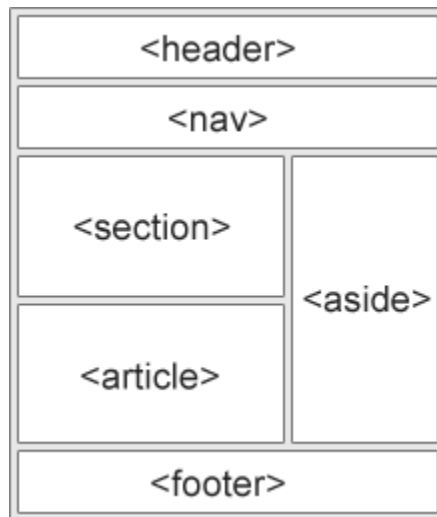
Semantic Elements

Many web sites contain HTML code like: `<div id="nav">` `<div class="header">` `<div id="footer">`

to indicate navigation, header, and footer.

HTML5 offers new semantic elements to define different parts of a web page:

- `<article>`
- `<aside>`
- `<details>`
- `<figcaption>`
- `<figure>`
- `<footer>`
- `<header>`
- `<main>`
- `<mark>`
- `<nav>`
- `<section>`
- `<summary>`
- `<time>`



HTML5 offers new elements for better document structure:

New Input Types

New Input Types	New Input Attributes
<ul style="list-style-type: none">• color• date• datetime• datetime-local• email• month• number• range• search• tel• time• url• week	<ul style="list-style-type: none">• autocomplete• autofocus• form• formaction• formenctype• formmethod• formnovalidate• formtarget• height and width• list• min and max• multiple• pattern (regexp)• placeholder• required• step

HTML5 - New Attribute Syntax

HTML5 allows four different syntaxes for attributes.

Type	Example
Empty	<code><input type="text" value="John" disabled></code>

Unquoted	<code><input type="text" value=John></code>
Double-quoted	<code><input type="text" value="John Doe"></code>
Single-quoted	<code><input type="text" value='John Doe'></code>

HTML5 Graphics

Tag	Description
<code><canvas></code>	Defines graphic drawing using JavaScript
<code><svg></code>	Defines graphic drawing using SVG

New Media Elements

Tag	Description
<code><audio></code>	Defines sound or music content
<code><embed></code>	Defines containers for external applications (like plug-ins)
<code><source></code>	Defines sources for <code><video></code> and <code><audio></code>
<code><track></code>	Defines tracks for <code><video></code> and <code><audio></code>
<code><video></code>	Defines video or movie content

HTML5 CANVAS

What is HTML Canvas?

The HTML <canvas> element is used to draw graphics, on the fly, via scripting (usually JavaScript).

The <canvas> element is only a container for graphics. You must use a script to actually draw the graphics.

Canvas has several methods for drawing paths, boxes, circles, text, and adding images.

With HTML5's Canvas API, we can draw anything and not only the rectangles, all through JavaScript. This can improve the performance of websites by avoiding the need to download images off the network. With canvas, we can draw shapes and lines, arcs and text, gradients and patterns. In addition, canvas gives us the power to manipulate pixels in images and even video.

The Canvas 2D Context spec is supported in:

Safari 2.0+, Chrome 3.0+, Firefox 3.0+, Internet Explorer 9.0+, Opera 10.0+, iOS (Mobile Safari) 1.0+, Android 1.0+

Canvas Examples

A canvas is a rectangular area on an HTML page. By default, a canvas has no border and no content.

The markup looks like this:

```
<canvas id="myCanvas" width="200" height="100"></canvas>
```

Basic Canvas Example

```
<canvas id="myCanvas" width="200" height="100" style="border:1px solid #000000;">
</canvas>
```

Drawing with JavaScript

```
var c = document.getElementById("myCanvas");  
var ctx = c.getContext("2d");  
ctx.fillStyle = "#FF0000";  
ctx.fillRect(0,0,150,75);
```

Draw a Line

```
var c = document.getElementById("myCanvas");  
var ctx = c.getContext("2d");  
ctx.moveTo(0,0);  
ctx.lineTo(200,100);  
ctx.stroke();
```

Draw a Circle

```
var c = document.getElementById("myCanvas");  
var ctx = c.getContext("2d");  
ctx.beginPath();  
ctx.arc(95,50,40,0,2*Math.PI);  
ctx.stroke();
```

Draw a Text

```
var c = document.getElementById("myCanvas");  
var ctx = c.getContext("2d");  
ctx.font = "30px Arial";  
ctx.fillText("Hello World",10,50);
```

Stroke Text

```
var c = document.getElementById("myCanvas");  
var ctx = c.getContext("2d");  
ctx.font = "30px Arial";  
ctx.strokeText("Hello World",10,50);
```

Draw Linear Gradient

```
var c = document.getElementById("myCanvas");  
var ctx = c.getContext("2d");  
  
// Create gradient  
var grd = ctx.createLinearGradient(0,0,200,0);  
grd.addColorStop(0,"red");  
grd.addColorStop(1,"white");  
  
// Fill with gradient  
ctx.fillStyle = grd;  
ctx.fillRect(10,10,150,80);
```

Draw Image

```
var c = document.getElementById("myCanvas");  
var ctx = c.getContext("2d");  
var img = document.getElementById("scream");  
ctx.drawImage(img,10,10);
```

What is SVG?

- SVG stands for Scalable Vector Graphics
- SVG is used to define graphics for the Web
- SVG is a W3C recommendation

The HTML <svg> Element

The HTML <svg> element (introduced in HTML5) is a container for SVG graphics. SVG has several methods for drawing paths, boxes, circles, text, and graphic images.

SVG Circle

Example

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

<svg width="100" height="100">
  <circle cx="50" cy="50" r="40" stroke="green" stroke-width="4" fill="yellow" />
</svg>

</body>
</html>
```

SVG Rectangle

Example

```
<svg width="400" height="100">
  <rect width="400" height="100" style="fill:rgb(0,0,255);stroke-width:10;stroke:rgb(0,0,0)" />
</svg>
```

SVG Rounded Rectangle

Example

```
<svg width="400" height="180">
  <rect x="50" y="20" rx="20" ry="20" width="150" height="150"
  style="fill:red;stroke:black;stroke-width:5;opacity:0.5" />
</svg>
```

SVG Star

Example

```
<svg width="300" height="200">  
  <polygon points="100,10 40,198 190,78 10,78 160,198"  
  style="fill:lime;stroke:purple;stroke-width:5;fill-rule:evenodd;" />  
</svg>
```

Comparison of Canvas and SVG

Canvas	SVG
<ul style="list-style-type: none">• Resolution dependent• No support for event handlers• Poor text rendering capabilities• You can save the resulting image as .png or .jpg• Well suited for graphic-intensive games	<ul style="list-style-type: none">• Resolution independent• Support for event handlers• Best suited for applications with large rendering areas (Google Maps)• Slow rendering if complex (anything that uses the DOM a lot will be slow)• Not suited for game applications

CSS

HTML mark-up can be used to represent

- Semantics: h1 means that an element is a top-level heading
- Presentation: h1 elements look a certain way
- ◆ It's advisable to separate semantics from presentation because:
 - It's easier to present documents on multiple platforms (browser, cell phone, spoken, ...)
 - It's easier to generate documents with consistent look
 - Semantic and presentation changes can be made independently of one another (division of labor)
 - User control of presentation is facilitated

- ◆ Cascading Style Sheets (CSS)
 - Applies to (X)HTML as well as XML documents in general

A styled HTML document



```
body { background-color:lime }
```

```
p { font-size:x-large; background-color:yellow }
```

CSS Selectors

CSS selectors allow you to select and manipulate HTML elements.

CSS selectors are used to "find" (or select) HTML elements based on their id, class, type, attribute, and more.

The element Selector

The element selector selects elements based on the element name.

You can select all <p> elements on a page like this: (all <p> elements will be center-aligned, with a red text color)

Example

```
p {  
    text-align: center;  
    color: red;  
}
```

The id Selector

The id selector uses the id attribute of an HTML element to select a specific element.

An id should be unique within a page, so the id selector is used if you want to select a single, unique element.

To select an element with a specific id, write a hash character, followed by the id of the element.

The style rule below will be applied to the HTML element with id="para1":

Example

```
#para1 {  
    text-align: center;  
    color: red;  
}  
<p id="para1">xxxxx</p>
```

The class Selector

The class selector selects elements with a specific class attribute.

To select elements with a specific class, write a period character, followed by the name of the class:

In the example below, all HTML elements with class="center" will be center-aligned:

Example

```
.center {  
    text-align: center;  
    color: red;  
}  
<p class="center">xxxx</p>
```

You can also specify that only specific HTML elements should be affected by a class. In the example below, all <p> elements with class="center" will be center-aligned:

```
p.center {  
    text-align: center;  color: red;  
}
```

Grouping Selectors

If you have elements with the same style definitions, like this:

```
h1 {  
    text-align: center;  
    color: red;  
}
```

```
h2 {  
    text-align: center;  
    color: red;  
}
```

```
p {  
    text-align: center;  
    color: red;  
}
```

you can group the selectors, to minimize the code.

To group selectors, separate each selector with a comma.

In the example below we have grouped the selectors from the code above:

```
h1, h2, p {  
    text-align: center;  
    color: red;  
}
```


Three Ways to Insert CSS

There are three ways of inserting a style sheet:

- External style sheet
- Internal style sheet
- Inline style

External Style Sheet

With an external style sheet, you can change the look of an entire website by changing just one file!

Each page must include a reference to the external style sheet file inside the <link> element. The <link> element goes inside the head section:

```
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
```

An external style sheet can be written in any text editor. The file should not contain any html tags. The style sheet file must be saved with a .css extension. An example of a style sheet file called "myStyle.css", is shown below:

```
body {
    background-color: lightblue;
}

h1 {
    color: navy;
    margin-left: 20px;
}
```

Internal Style Sheet

An internal style sheet may be used if one single page has a unique style.

Internal styles are defined within the <style> element, inside the head section of an HTML page:

Example

```
<head>
<style>
body { background-color: linen;}

h1 {
    color: maroon;
    margin-left: 40px;
}
</style>
</head>
```

Inline Styles

An inline style may be used to apply a unique style for a single element.

An inline style loses many of the advantages of a style sheet (by mixing content with presentation). Use this method sparingly!

To use inline styles, add the style attribute to the relevant tag. The style attribute can contain any CSS property. The example shows how to change the color and the left margin of a <h1> element:

Example

```
<h1 style="color:blue;margin-left:30px;">This is a heading.</h1>
```

Multiple Styles Will Cascade into One

- in an external CSS file
- inside the <head> section of an HTML page
- inside an HTML element

Cascading order

What style will be used when there is more than one style specified for an HTML element?

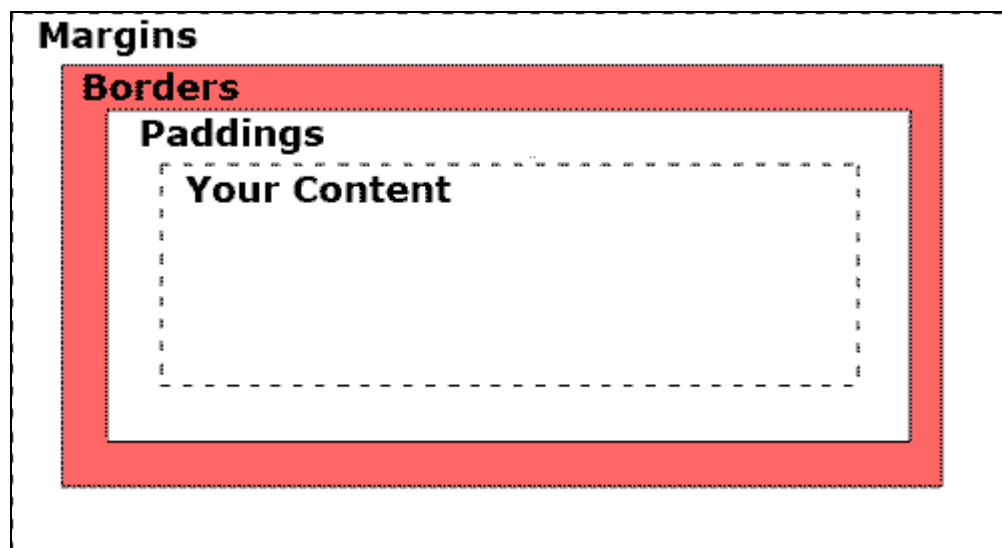
Generally speaking we can say that all the styles will "cascade" into a new "virtual" style sheet by the following rules, where number three has the highest priority:

1. Browser default
2. External and internal style sheets (in the head section)
3. Inline style (inside an HTML element)

So, an inline style (inside an HTML element) has the highest priority, which means that it will override a style defined inside the <head> tag, or in an external style sheet, or in a browser (a default value).

Box Model

- shows the areas that CSS styles are applied too
- Your text and or graphic images content is placed in the box which can be surrounded by padding, border and margin as shown in the Box Model Diagram



The box model allows us to add a border around elements, and to define space between elements.

Content - The content of the box, where text and images appear

Padding - Clears an area around the content. The padding is transparent; We can set specific padding sizes by using these four properties:

- padding-top:
- padding-right:
- padding-bottom:
- padding-left

Border - A border that goes around the padding and content

- border-width: 10px; *absolute (px, in, cm, or 'thin', 'medium', 'thick'), or relative (ems)*
- border-style: *dotted, dashed, solid, double, groove, ridge, inset, outset, hidden, etc*
- border-color: *r: 'blue', 'red', #FF9900, rgb(0,0,255)etc*

Margin - Clears an area outside the border. The margin is transparent ,We can set the specific margin sizes by using these four properties:

- margin-top:
- margin-right:
- margin-bottom:
- margin-left:

In practice, few web designers use these properties, preferring a **shorthand** method that condenses the declaration to a single line:

margin: 10px; →	Sets all four margins to be 10px.
margin: 10px 5px; →	Sets the top and bottom margins as 10px and the left and right margins as 5px.
margin: 20px 10px 5px 15px; →	Sets the top margin as 20px, the right margin as 10px, the bottom margin as 5px, and the left margin as 15px.

padding: 5px;



Sets padding as 5px on all four sides.

padding: 25px 5px;



Sets top and bottom padding as 25px and left and right padding as 5px.

padding: 50px 10px 15px 5px; →

Sets top padding as 50px, right padding as 10px, bottom padding as 15px, and the left padding as 5px.

border-width: 10px;
border-style: solid dashed;
border-color: blue red orange gray;

border: 5px solid blue;

Font and Text Property

- The **font-family** property is used to change the face of a font.
- The **font-style** property is used to make a font italic or oblique.
- The **font-variant** property is used to create a small-caps effect.
- The **font-weight** property is used to increase or decrease how bold or light a font appears.
- The **font-size** property is used to increase or decrease the size of a font.
- The **font** property is used as shorthand to specify a number of other font properties.

- The **color** property is used to set the color of a text.
- The **direction** property is used to set the text direction.
- The **letter-spacing** property is used to add or subtract space between the letters that make up a word.
- The **word-spacing** property is used to add or subtract space between the words of a sentence.
- The **text-indent** property is used to indent the text of a paragraph.
- The **text-align** property is used to align the text of a document.
- The **text-decoration** property is used to underline, overline, and strikethrough text.
- The **text-transform** property is used to capitalize text or convert text to uppercase or lowercase letters.
- The **white-space** property is used to control the flow and formatting of text.

CSS3 Introduction

CSS3 is the latest standard for CSS.

CSS3 is completely backwards-compatible with earlier versions of CSS.

CSS3 has been split into "modules". It contains the "old CSS specification" (which has been split into smaller pieces). In addition, new modules are added.

Some of the most important CSS3 modules are:

- Selectors
- Box Model
- Backgrounds and Borders
- Image Values and Replaced Content
- Text Effects
- 2D/3D Transformations
- Animations
- Multiple Column Layout
- User Interface

CSS3 border-radius Property

If you specify only one value for the border-radius property, this radius will be applied to all 4 corners.

However, you can specify each corner separately if you wish. Here are the rules:

- **Four values:** first value applies to top-left, second value applies to top-right, third value applies to bottom-right, and fourth value applies to bottom-left corner
- **Three values:** first value applies to top-left, second value applies to top-right and bottom-left, and third value applies to bottom-right
- **Two values:** first value applies to top-left and bottom-right corner, and the second value applies to top-right and bottom-left corner
- **One value:** all four corners are rounded equally

Example

```
#rcorners4 {  
    border-radius: 15px 50px 30px 5px;
```

```

background: #8AC007;
padding: 20px;
width: 200px;
height: 150px;
}
#rcorners5 {
border-radius: 15px 50px 30px;
background: #8AC007;
padding: 20px;
width: 200px;
height: 150px;
}

```

CSS3 Rounded Corners Properties

Property	Description
border-radius	A shorthand property for setting all the four border-*-*-radius properties
border-top-left-radius	Defines the shape of the border of the top-left corner
border-top-right-radius	Defines the shape of the border of the top-right corner
border-bottom-right-radius	Defines the shape of the border of the bottom-right corner
border-bottom-left-radius	Defines the shape of the border of the bottom-left corner

CSS3 border-image Property

The CSS3 `border-image` property allows you to specify an image to be used instead of the normal border around an element.

The property has three parts:

1. The image to use as the border

2. Where to slice the image
3. Define whether the middle sections should be repeated or stretched

CSS3 Border Properties

Property	Description
border-image	A shorthand property for setting all the border-image-* properties
border-image-source	Specifies the path to the image to be used as a border
border-image-slice	Specifies how to slice the border image
border-image-width	Specifies the widths of the border image
border-image-outset	Specifies the amount by which the border image area extends beyond the border box
border-image-repeat	Specifies whether the border image should be repeated, rounded or stretched

CSS3 Backgrounds

CSS3 contains a few new background properties, which allow greater control of the background element.

In this chapter you will learn how to add multiple background images to one element.

You will also learn about the following new CSS3 properties:

- background-size
- background-origin
- background-clip

CSS3 Multiple Backgrounds

CSS3 allows you to add multiple background images for an element, through the background-image property.

The different background images are separated by commas, and the images are stacked on top of each other, where the first image is closest to the viewer.

The following example has two background images, the first image is a flower (aligned to the bottom and right) and the second image is a paper background (aligned to the top-left corner):

Example

```
#example1 {  
    background-image: url(img_flwr.gif), url(paper.gif);  
    background-position: right bottom, left top;  
    background-repeat: no-repeat, repeat;  
}
```

Multiple background images can be specified using either the individual background properties (as above) or the background shorthand property.

The following example uses the background shorthand property (same result as example above):

Example

```
#example1 {  
    background: url(img_flwr.gif) right bottom no-repeat, url(paper.gif) left top repeat;  
}
```

CSS3 Background Properties

Property	Description
background	A shorthand property for setting all the background properties in one declaration

background-clip	Specifies the painting area of the background
background-image	Specifies one or more background images for an element
background-origin	Specifies where the background image(s) is/are positioned
background-size	Specifies the size of the background image(s)

CSS3 background-origin Property

The CSS3 `background-origin` property specifies where the background image is positioned.

The property takes three different values:

- `border-box` - the background image starts from the upper left corner of the border
- `padding-box` - (default) the background image starts from the upper left corner of the padding edge
- `content-box` - the background image starts from the upper left corner of the content

CSS3 background-clip Property

The CSS3 `background-clip` property specifies the painting area of the background.

The property takes three different values:

- `border-box` - (default) the background is painted to the outside edge of the border
- `padding-box` - the background is painted to the outside edge of the padding
- `content-box` - the background is painted within the content box

CSS3 Colors

CSS supports color names, hexadecimal and RGB colors.

In addition, CSS3 also introduces:

- RGBA colors
- HSL colors

- HSLA colors
- opacity

RGBA Colors

RGBA color values are an extension of RGB color values with an alpha channel - which specifies the opacity for a color.

An RGBA color value is specified with: `rgba(red, green, blue, alpha)`. The alpha parameter is a number between 0.0 (fully transparent) and 1.0 (fully opaque).

Example

```
#p1 {background-color: rgba(255, 0, 0, 0.3);} /* red with opacity */
```

HSL Colors

HSL stands for Hue, Saturation and Lightness.

An HSL color value is specified with: `hsl(hue, saturation, lightness)`.

1. Hue is a degree on the color wheel (from 0 to 360):
 - 0 (or 360) is red
 - 120 is green
 - 240 is blue
2. Saturation is a percentage value: 100% is the full color.
3. Lightness is also a percentage; 0% is dark (black) and 100% is white.

Example: `#p1 {background-color: hsl(255, 0, 0, 0.3);} /* red with opacity */`

HSLA Colors

HSLA color values are an extension of HSL color values with an alpha channel - which specifies the opacity for a color.

An HSLA color value is specified with: `hsla(hue, saturation, lightness, alpha)`, where the alpha parameter defines the opacity. The alpha parameter is a number between 0.0 (fully transparent) and 1.0 (fully opaque).

Example

```
#p1 {background-color: hsla(120, 100%, 50%, 0.3);} /* green with opacity */
```

Opacity

The CSS3 opacity property sets the opacity for a specified RGB value.

The opacity property value must be a number between 0.0 (fully transparent) and 1.0 (fully opaque).

Example: #p1 {background-color:rgb(255,0,0);opacity:0.6;} /* red with opacity */

CSS3 Gradients

CSS3 gradients let you display smooth transitions between two or more specified colors.

CSS3 gradients you can reduce download time and bandwidth usage. In addition, elements with gradients look better when zoomed, because the gradient is generated by the browser.

CSS3 defines two types of gradients:

- **Linear Gradients (goes down/up/left/right/diagonally)**
- **Radial Gradients (defined by their center)**

The following table lists the CSS3 shadow properties:

Property	Description
box-shadow	Adds one or more shadows to an element
text-shadow	Adds one or more shadows to a text

CSS3 Text Shadow

The CSS3 text-shadow property applies shadow to text.

In its simplest use, you only specify the horizontal shadow (2px) and the vertical shadow (2px):

text-shadow: *h-shadow v-shadow blur-radius color|none|initial|inherit*;

CSS3 Box Shadow box-shadow: none|*h-shadow v-shadow blur spread color |inset|initial|inherit*;

Question Bank

Part-A

1. Illustrate the inline CSS to show a section of document with a font size of 20
2. Create a simple code using image map.
3. Tabulate the difference between internet and intranet.
4. How Rich Internet Application differ from traditional web application
5. Classify the need for different types of cascading style sheet
6. Compare web 2.0 and web 1.0
7. List some of the collaboration tools.
8. Discuss on how RIA differ from a traditional web application.
9. Point out the difference between websites and web server..
10. Explain URL.
11. Classify different table tag.
12. Show the structure of HTML 5
13. Predict the benefits of intranet? Discuss Extranet.
14. List any two font formatting tag.
15. Classify the need for cascading style sheet.
16. Prepare a simple code using image map.
17. Name the three ways of inserting a style sheet
18. List the features of CSS?
19. Tell about difference between HTML and XHTML.
20. What do you infer from XHTML?

Part-B

1. Discuss the basic structure of HTML5
2. Discuss About XHTML tags with example
3. Creates a web page which should display some of the egg. Colleges in tamilnadu in a tabular form with name, address and URL of the web site of the college.
4. Summarize the overview of internet technologies.
5. Describe the core syntax of CSS with the help of some suitable example
6. List and explain in detail the types of selector strings .and explain the significance of with the help of a real time application. Write necessary code snippets
7. Elaborate with Example Usage of CSS in HTML
8. Write Brief Notes on Rich Internet Applications
9. Write Brief Notes on Collaborations tools
10. Describe the CSS box model in detail? Create a webpage using background-image property where the code has to written in XHTML.
11. Discus on website and web server ,Internet and Intranet ,HTML and XHTML
12. Write Short notes on HTML 5 and CSS3
13. Compare a) i)HTML ii)XHTML iii)HTML 5 b) i)CSS ii)CSS3
14. Write Brief Notes on RIA?
15. Create an XHTML document markup of your Resumes
16. Write the CSS rule for controlling the positioning of elements in an XHTML Document ?