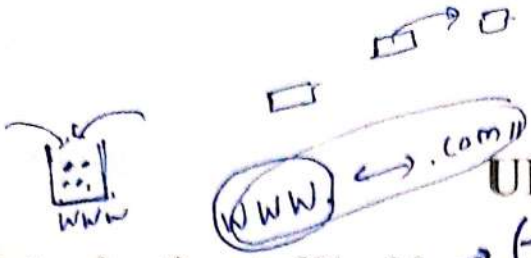


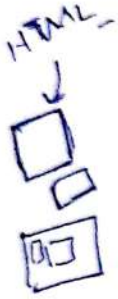
UNIT 1

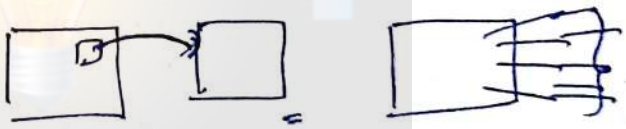
1 Introduction to World Wide Web

(It is a huge collection of information linked to each other around the globe)

www.google.com Web page

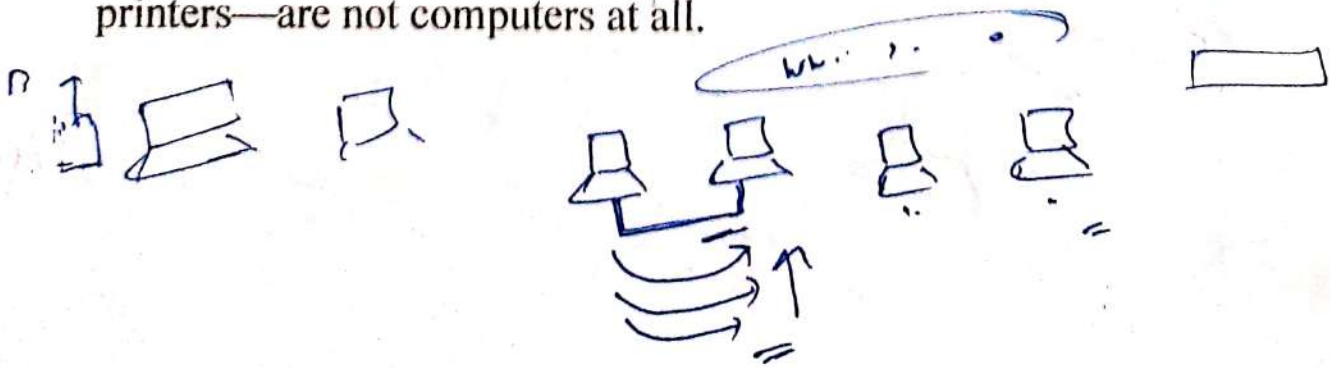
The World Wide Web (www, W3) is an information space where documents and other web resources are identified by URIs, interlinked by hypertext links, and can be accessed via the Internet. It has become known simply as *the Web.* Hypertext documents are commonly called *web pages,* which are primarily text documents formatted and annotated with the Hypertext Markup Language (HTML). Webpages may contain links to images, video, and software components that are rendered to users of a web browser application, running on the user's computer, as coherent pages of multimedia content. Embedded hyperlinks permit users to navigate between web pages. When multiple web pages are published with a common theme or within a common domain name, the collection is usually called a *web site.*





Internet

The Internet is a huge collection of computers connected in a communications network. These computers are of every imaginable size, configuration, and manufacturer. In fact, some of the devices connected to the Internet—such as plotters and printers—are not computers at all.



FTP

The innovation that allows all of these diverse devices to communicate with each other is a single, low-level protocol: The Transmission Control Protocol/Internet Protocol (TCP/IP). TCP/IP became the standard for computer network connections in 1982.

It can be used directly to allow a program on one computer to communicate with a program on another computer via the Internet. In most cases, however, a higher-level protocol runs on top of TCP/IP.

Internet Protocol Address (2) IP

IP Addr

The Internet Protocol (IP) address of a machine connected to the Internet is a unique 32-bit number. IP addresses usually are written as four 8-bit numbers, separated by periods.

192.162.008.007

The four parts are separately used by Internet-routing computers to decide where a message must go next to get to its destination. numbers, separated by periods.

198.162.168.192

WWW.900.com

The four parts are separately used by Internet-routing computers to decide where a message must go next to get to its destination. Although people nearly always type domain names into their browsers, the IP works just as well. For example, the IP for United Airlines (www.ual.com)

IPv4

32 bit-

IPv6

8b . 8b . 8b . 8bi

$2^8 = 256$

0 -> 255

Web Computer

IPadd

Domain Names

Because people have difficulty dealing with and remembering numbers, machines on the Internet also have textual names. These names begin with the name of the host machine, followed by progressively larger enclosing collections of machines, called domains.

There may be two, three, or more domain names. The first domain name, which appears immediately to the right of the host name, is the domain of which the host is a part.

The second domain name gives the domain of which the first domain is a part. The last domain name identifies the type of organization in which the host resides, which is the largest domain in the site's name

Web Browsers

A web browser is a software program that allows a user to locate, access, and display web pages. In common usage, a web browser is usually shortened to "browser." Browsers are used primarily for displaying and accessing websites on the internet, as well as other content created using languages such as Hypertext Markup Language (HTML) and Extensible Markup Language (XML).

Browsers translate web pages and websites delivered using Hypertext Transfer Protocol (HTTP) into human-readable content. They also have the ability to display other protocols and prefixes, such as secure HTTP (HTTPS), File Transfer Protocol (FTP), email handling (mailto:), and files (file:). In addition,

most browsers also support external plug-ins required to display active content, such as in-page video, audio and game content.

Most commonly used web browsers are Google Chrome, Firefox, Internet Explorer, Opera, Safari, etc.

Architecture

There are a lot of web browser available in the market. All of them interpret and display information on the screen however their capabilities and structure vary 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 based on input it receives.

Interpreter receives the information from the controller and execute the instruction line by line. Some interpreter 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 service. Following are the client programs that are commonly used: HTTP, SMTP, FTP, NNTP, POP

Web Servers

Web server is a computer where the web content is stored.

WEB SERVER

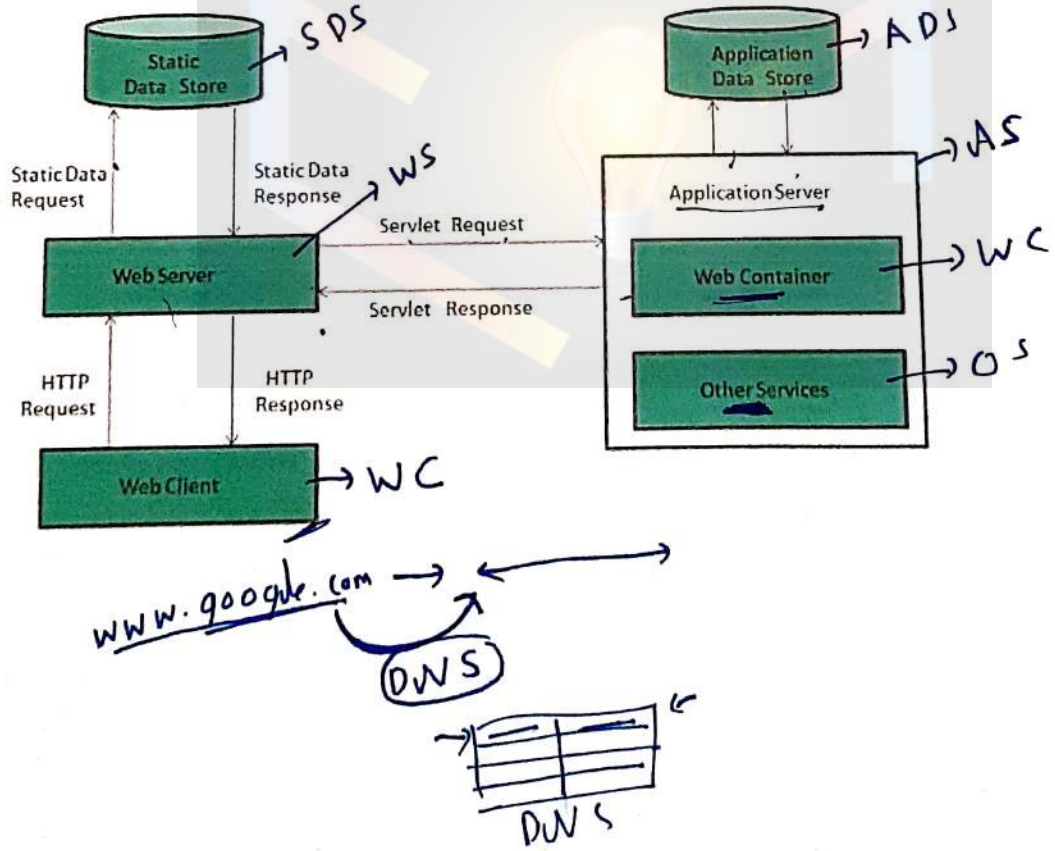
Web servers are computers that deliver (serves up) Web pages. Every Web server has an IP address and possibly a domain name. For example, if you enter the URL http://www.webopedia.com/index.html in your browser, this sends a request to the Web server whose domain name is webopedia.com. The server then fetches the page named index.html and sends it to your browser.

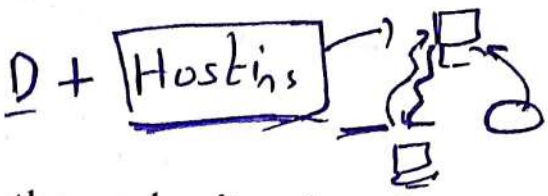
Any computer can be turned into a Web server by installing server software and connecting the machine to the Internet. There are many Web server software applications, including public domain software and commercial packages.

Web Server Working

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

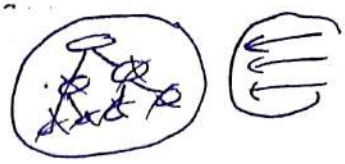
- Sending the file to the client associated with the requested URL.
- Generating response by invoking a script and communicating with database





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

The most commonly used Web servers are Apache and Microsoft's Internet Information Server (IIS).



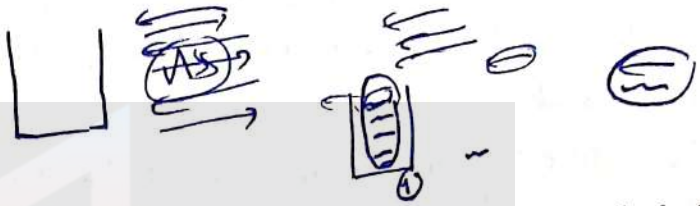
Architecture

Web Server Architecture follows the following two approaches:

- Concurrent Approach
- Single-Process-Event-Driven Approach.

Concurrent approach allows the web server to handle multiple client requests at the same time. It can be achieved by following methods:

- Multi-process
- Multi-threaded
- Hybrid method
- Multi-processing



In this a single process (parent process) initiates several single-threaded child processes and distribute incoming requests to these child processes. Each of the child processes are responsible for handling single request.

It is the responsibility of parent process to monitor the load and decide if processes should be killed or forked.

Multi-threaded: Unlike Multi-process, it creates multiple single-threaded process.

Hybrid: It is combination of above two approaches. In this approach multiple process are created and each process initiates multiple threads. Each of the threads handles one connection. Using multiple threads in single process results in less load on system resources.

Examples

Following table describes the most leading web servers available today:

1.Apache HTTP Server (OS)

This is the most popular web server in the world developed by the Apache Software Foundation. Apache web server is an open source software and can be installed on almost all operating systems including Linux, UNIX,

Windows, FreeBSD, Mac OS X and more. About 60% of the web server machines run the Apache Web Server.

2.Internet Information Services (IIS)

The Internet Information Server (IIS) is a high-performance Web Server from Microsoft. This web server runs on Windows NT/2000 and 2003 platforms (and may be on upcoming new Windows version also). IIS comes bundled with Windows NT/2000 and 2003; Because IIS is tightly integrated with the operating system, so it is relatively easy to administer it.

3.Lighttpd

The lighttpd, pronounced lighty is also a free web server that is distributed with the FreeBSD operating system. This open source web server is fast, secure and consumes much less CPU power. Lighttpd can also run on Windows, Mac OS X, Linux and Solaris operating systems.

4.Sun Java System Web Server

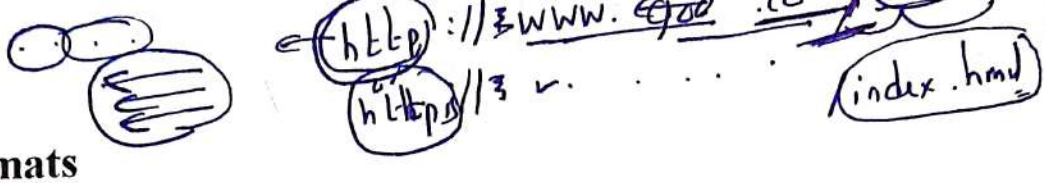
This web server from Sun Microsystems is suited for medium and large web sites. Though the server is free it is not open source. It, however, runs on Windows, Linux and UNIX platforms. The Sun Java System web server supports various languages, scripts and technologies required for Web 2.0 such as JSP, Java Servlets, PHP, Perl, Python, and Ruby on Rails, ASP and ColdFusion etc.

5.Jigsaw Server

Jigsaw (W3C's Server) comes from the World Wide Web Consortium. It is open source and free and can run on various platforms like Linux, UNIX, Windows, and Mac OS X Free BSD etc. Jigsaw has been written in Java and can run CGI scripts and PHP programs.

Uniform Resource Locator (URL)

Uniform (or universal) resource locators (URLs) are used to identify documents (resources) on the Internet. There are many different kinds of resources, identified by different forms of URLs.



URL Formats

All URLs have the same general format: scheme: object-address .The scheme is often a communications protocol. Common schemes include http, ftp, gopher, telnet, file, mailto, and news.

HTTP protocol supports the Web. This protocol is used to request and send eXtensible Hypertext Markup Language (XHTML) documents. In the case of HTTP, the form of the object address of a URL is as follows: //fully-qualified-domain-name/path-to-document .

URL Paths

The path to the document for the HTTP protocol is similar to a path to a file or directory in the file system of an operating system and is given by a sequence of directory names and a file name, all separated by whatever separator character the operating system uses. For UNIX servers, the path is specified with forward slashes; for Windows servers, it is specified with backward slashes.

The path in a URL can differ from a path to a file because a URL need not include all directories on the path. A path that

includes all directories along the way is called a complete path. In most cases, the path to the document is relative to some base path that is specified in the configuration files of the server. Such paths are called partial paths.

E.g.: http://www.gumboco.com/files/f99/storefront.html

Multipurpose Internet Mail Extensions (MIME)

Multipurpose Internet Mail Extensions (MIME) is an Internet standard that helps extend the limited capabilities of email by allowing insertion of images, sounds and text in a message. It was proposed by Bell Communications in 1991, and the specification was originally defined in June 1992 for RFCs 1341 and 1342.

MIME was designed to extend the format of email to support non-ASCII characters, attachments other than text format, and message bodies which contain multiple parts. MIME describes the message content type and the type of encoding used with the help of headers. All manually composed and automated emails are transmitted through SMTP in MIME format. The association of Internet email with SMTP and MIME standards is such that the emails are sometimes referred to as SMTP/MIME email. The MIME standard defines the content types which are of prime importance in communication protocols like HTTP for the World Wide Web. The data are transmitted in the form of email messages through HTTP even though the data are not an email.

The features offered by MIME to email services are as follows:

Support for multiple attachments

in a single message Support for

non-ASCII characters

Support for layouts, fonts and colors which are categorized as rich text.

Support for attachments which may contain executables, audio,

images and video files, etc. Support for unlimited message

length.

MIME is extensible because it defines a method to register new content types and other MIME attribute values. The format of a message body is described by MIME using special header directives. This is done so that the email can be represented correctly by the client.

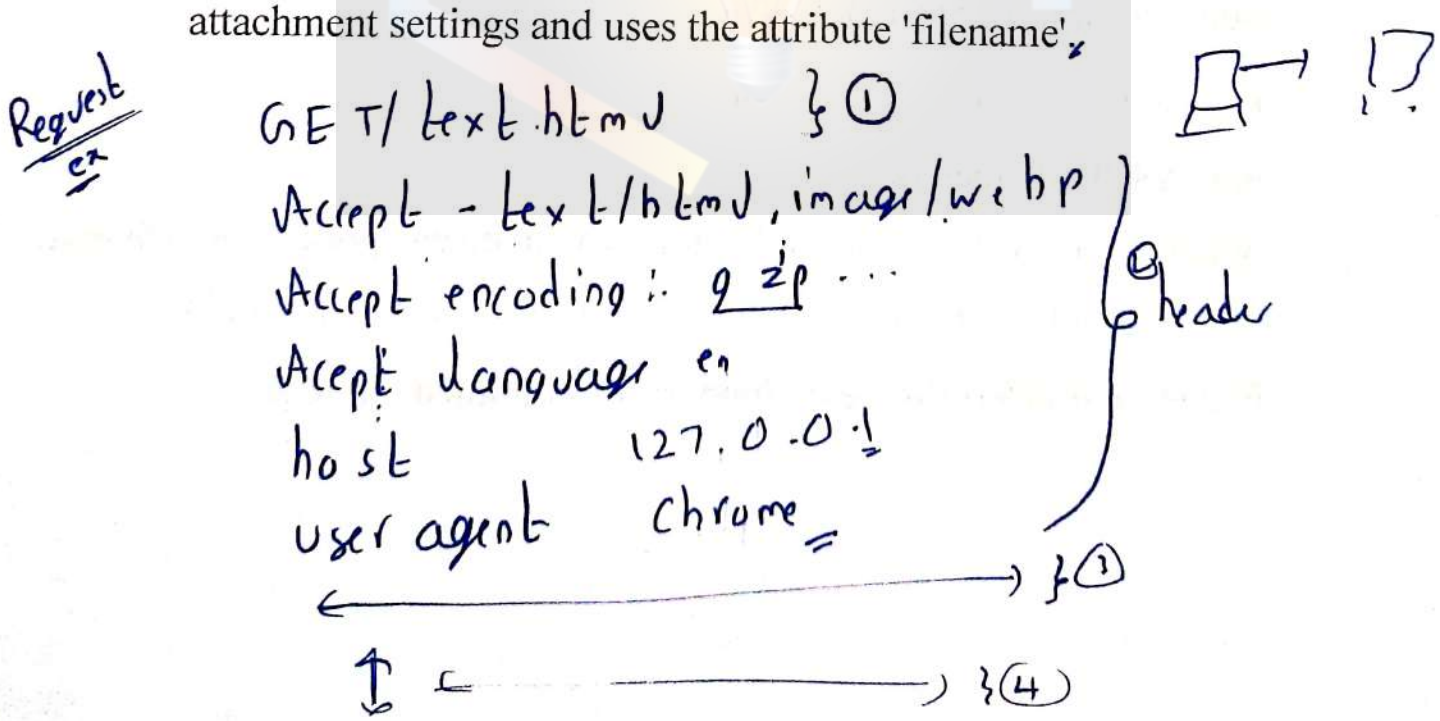
MIME Version: The presence of MIME Version generally indicates whether the message is MIME formatted. The value of the header is 1.0 and it is shown as MIME-Version: 1.0. The idea behind this was to create more advanced versions of MIME like 2.0 and so on.

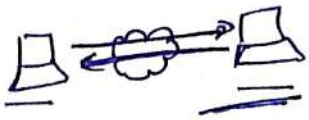
Content-Type: This describes the data's Internet media type and the subtype. It may consist of a 'charset' parameter separated by a semicolon specifying the character set to be used. For example: Content-Type: Text/Plain. UTF-8

Content-Transfer-Encoding: It specifies the encoding used in the message body.

Content-Description: Provides additional information about the content of the message.

Content-Disposition: Defines the name of the file and the attachment settings and uses the attribute 'filename'.



Hypertext Transfer Protocol (HTTP) ①

All Web communications transactions use the same protocol: The Hypertext Transfer Protocol (HTTP). HTTP consists of two phases: the request and the response.

Each HTTP communication (request or response) between a browser and a Web server consists of two parts: a header and a body. The header contains information about the communication; the body contains the data of the communication if there is any.

The Request Phase

The general form of an HTTP request is as follows:

1. HTTP method Domain part of the URL HTTP version
2. Header fields
3. Blank line
4. Message body

The following is an example of the first line of an HTTP request: GET /storefront.html HTTP/1.1

Response ex

GET / test.html

HTTP/0.9 200 ok → ①

Connection : close

Context type : text/html ; charset = UTF-8

Context length : 742 } ②

<!doctype html>

<html>

⋮ ③

</html>

HTTP Request Methods

Method	Description
GET	Returns the contents of the specified document
HEAD	Returns the header information of the specified document
POST	Executes the specified document, using the enclosed data
PUT	Replaces the specified document with the enclosed data
DELETE	Deletes the specified document

GET and POST are the most frequently used. The format of a header field is the field name followed by a colon and the value of the field. There are four categories of header fields:

1. General: For general information, such as the date
2. Request: Included in request headers
3. Response: For response headers
4. Entity: Used in both request and response headers

The Response Phase

The general form of an HTTP response is as follows:

1. Status line
2. Response header fields
3. Blank line
4. Response body

The status line includes the HTTP version used, a three-digit status code for the response, and a short textual explanation of the status code. For example, most responses begin with the following: HTTP/1.1 200 OK

The general meanings of the five categories specified by these first digits are shown in Table.

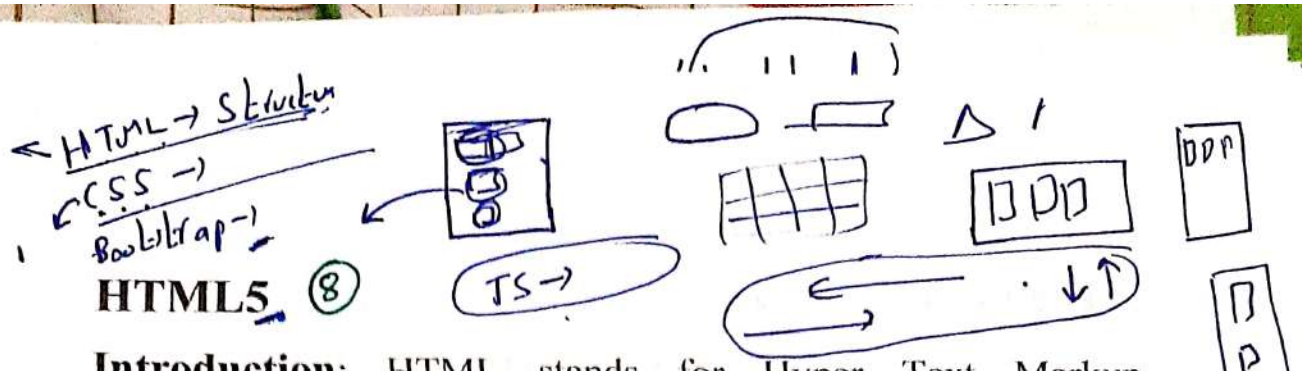
First digits of HTTP status codes

First Digit(format)	Category
1xx	Informational
2xx	Success
3xx	Redirection
4xx	Client error
5xx	Server error

100 continue
101 switch protocol
200 ok
202 accept
301 moved permanently

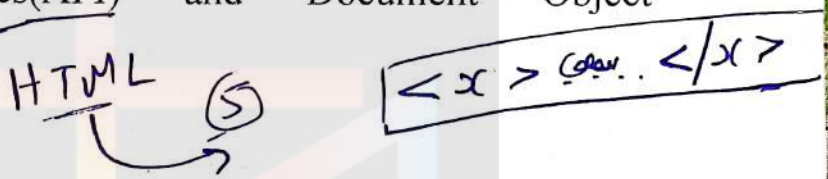
One of the more common status codes is one user never want to see: 404 Not Found, which means the requested file could not be found.

400 Bad request
404 not found
502 Bad Gateway
504 Gateway time out



HTML5 ⑧

Introduction: HTML stands for Hyper Text Markup Language. It is used to design web pages using markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. Markup language is used to define the text document within tag which defines the structure of web pages. HTML 5 is the fifth and current version of HTML. It has improved the markup available for documents and has introduced application programming interfaces(API) and Document Object Model(DOM).



Features:

It has introduced new multimedia features which supports audio and video controls by using
<audio> and <video> tags.

There are new graphics elements including vector graphics and tags.

Enrich semantic content by including <header> <footer>, <article>, <section> and <figure> are added.

Drag and Drop- The user can grab an object and drag it further dropping it on a new location/Geo-location services- It helps to locate the geographical location of a client./

Web storage facility which provides web application methods to store data on web browser. Uses SQL database to store data

offline.

Allows to draw various shapes like triangle, rectangle, circle, etc. Capable of handling incorrect syntax.

Easy DOCTYPE declaration i.e. <!doctype html> Easy character encoding i.e. <meta charset="UTF-8">

<html>
.
.
.
</html>

<!doctype html>
<meta charset="UTF-8">

Advantages:

All browsers supported.

More device friendly.

Easy to use and implement.

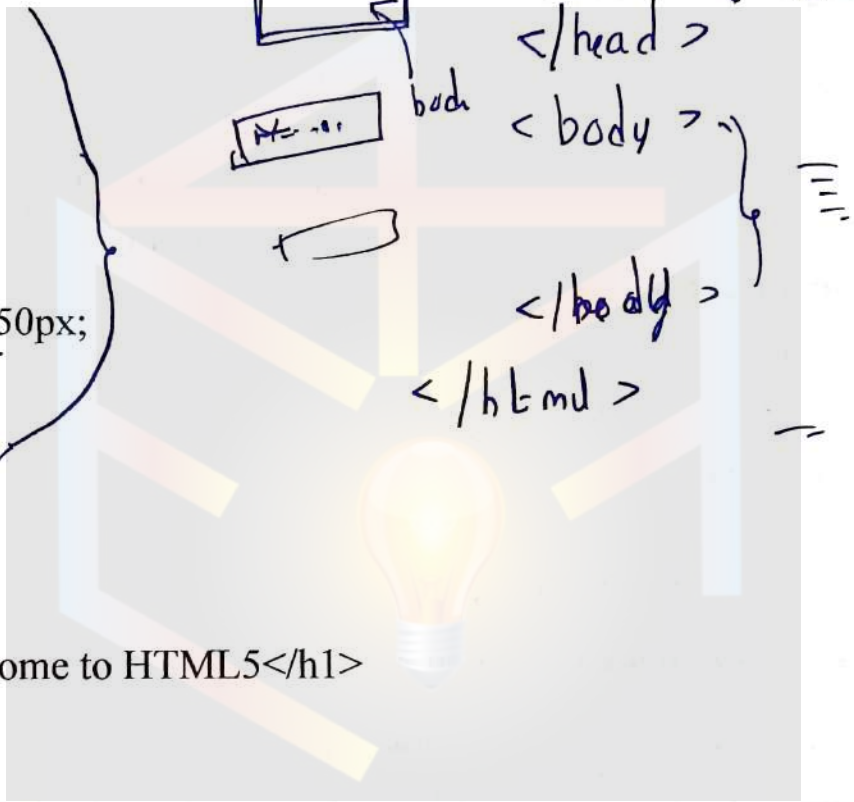
HTML 5 in integration with CSS, JavaScript, etc can help build beautiful websites. Disadvantages:

Long codes have to be written which is time consuming.

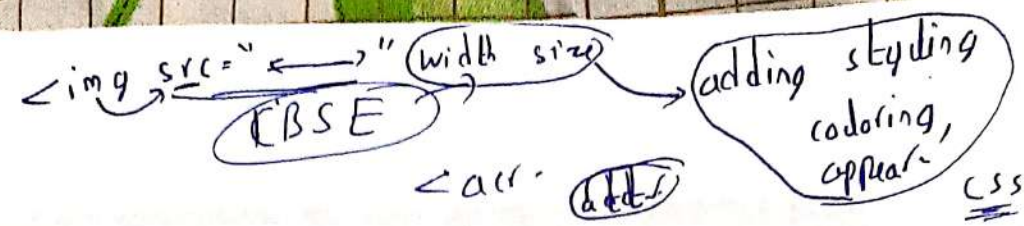
Only modern browsers support it.

Example:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML 5</title>
<styl
e>
h1
{
font-size:50px;
}
</style>
</head>
<body>
<h1>welcome to HTML5</h1>
</body>
</html>
```



Removed elements from HTML 5: There are many elements which are depreciated from HTML 5 are listed below:



<acronym>	<abbr>
<applet>	<object>
<basefont>	CSS

Hi
Hi
Hi

commonly used tags (HTML) ⑨

<h1> ... </h1> .... <h6> ... </h6> (heading)
 ... (paragraph)
 ... (bold)
 ... (italics)
<s> ... </s> (strike)
<u> ... </u> (underline)

(CSS)

<h1>Hi</h1>
<h1>
:
h6>

 (break) } stand alone tag.
<hr> (horizontal line)

lists, img, forms, link, audio, video, tables

_{....} ^{...}

 hello   Hello

<u> hello </u>

<h1> </h1>

<hr>

$x_{②} + x_{③} =$
$x^{②} + y^{②} =$ <hr />

REMOVED ELEMENTS	USE INSTEAD ELEMENTS
<big>	CSS
	CSS
<dir>	
<font>	CSS
<frame>	
<frameset>	
<noframes>	
<isindex>	
<strike>	CSS, <s> or <del>
<tt>	CSS

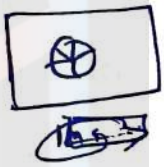
New Added Elements in HTML 5:

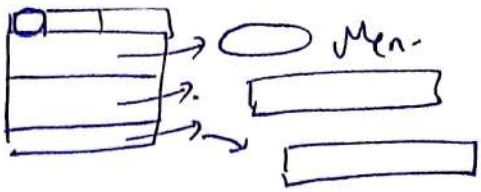
<article>: The <article> tag is used to represent an article. More specifically, the content within the <article> tag is independent from the other content of the site (even though it can be related).

<aside>: The <aside> tag is used to describe the main object of the web page in a shorter way like a highlighter. It basically identifies the content that is related to the primary content of the web page but does not constitute the main intent of the primary page. The <aside> tag contains mainly author information, links, related content and so on.

<figcaption>: The <figurecaption> tag in HTML is used to set a caption to the figure element in a document.

<figure>: The <figure> tag in HTML is used to add self-contained content like illustrations, diagrams, photos or codes listing in a document. It is related to main flow but it can be used in any position of a document and the figure goes with the flow of the document and if remove it then it should not affect the flow of the document.





<header>: It contains the section heading as well as other content, such as a navigation links, table of contents, etc.

<footer>: The <footer> tag in HTML is used to define a footer of HTML document. This section contains the footer information (author information, copyright information, carriers etc). The footer tag are used within body tag. The <footer> tag is new in the HTML 5. The footer elements require a start tag as well as an end tag.

<main>: Delineates the main content of the body of a document or web app.

<mark>: The <mark> tag in HTML is used to define the marked text. It is used to highlight the part of the text in the paragraph.

<nav>: The <nav> tag is used to declaring the navigational section in HTML documents. Websites typically have sections dedicated to navigational links, which enables user to navigate the site. These links can be placed inside a nav tag.

<section>: It demarcates a thematic grouping of content.

<details>: The <details> tag is used for the content/information which is initially hidden but could be displayed if the user wishes to see it. This tag is used to create interactive widget which user can open or close it. The content of details tag is visible when open the set attributes.

<summary>: The <summary> tag in HTML is used to define a summary for the <details> element. The <summary> element is used along with the <details> element and provides a summary visible to the user. When the summary is clicked by the user, the

content placed inside the <details> element becomes visible which was previously hidden. The <summary> tag was added in HTMl 5. The <summary> tag requires both starting and ending tag.

<time>: The <time> tag is used to display the human-readable data/time. It can also be used to encode dates and times in a machine-readable form. The main advantage for users is that they can offer to add birthday reminders or scheduled events in their calender's and search engines can produce smarter search results.

<bdi>: The <bdi> tag refers to the Bi-Directional Isolation. It differentiate a text from other text that may be formatted in different direction. This tag is used when a user generated text with an unknown directions.

<wbr>: The <wbr> tag in HTML stands for word break opportunity and is used to define the position within the text which is treated as a line break by the browser. It is mostly used when the used word is too long and there are chances that the browser may break lines at the wrong place for fitting the text.

<datalist>: The <datalist> tag is used to provide autocomplete feature in the HTML files. It can be used with input tag, so that users can easily fill the data in the forms using select the data.

<keygen>: The <keygen> tag in HTML is used to specify a key-pair generator field in a form. The purpose of <keygen> element is to provide a secure way to authenticate users. When a from is submitted then two keys are generated, private key and public key. The private key stored locally, and the public key is sent to the server. The public key is used to generate client certificate to authenticate user for future.

<output>: The <output> tag in HTML is used to represent the result of a calculation performed by the client-side script such as JavaScript.

<progress>: It is used to represent the progress of a task. It is also define that how much work is done and how much is left to download a things. It is not used to represent the disk space or relevant query.

<svg>: It is the Scalable Vector Graphics.

<canvas>: The <canvas> tag in HTML is used to draw graphics on web page using JavaScript. It can be used to draw paths, boxes, texts, gradient and adding images. By default it does not contains border and text.

<audio>: It defines the music or audio content.

<embed>: Defines containers for external applications (usually a video player).

<source>: It defines the sources for <video> and <audio>.

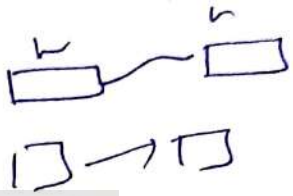
<track>: It defines the tracks for <video> and <audio>.

<video>: It defines the video content.

1. Hyperlinks is one of the very core features of HTML, they enable you to jump from one webpage to another.

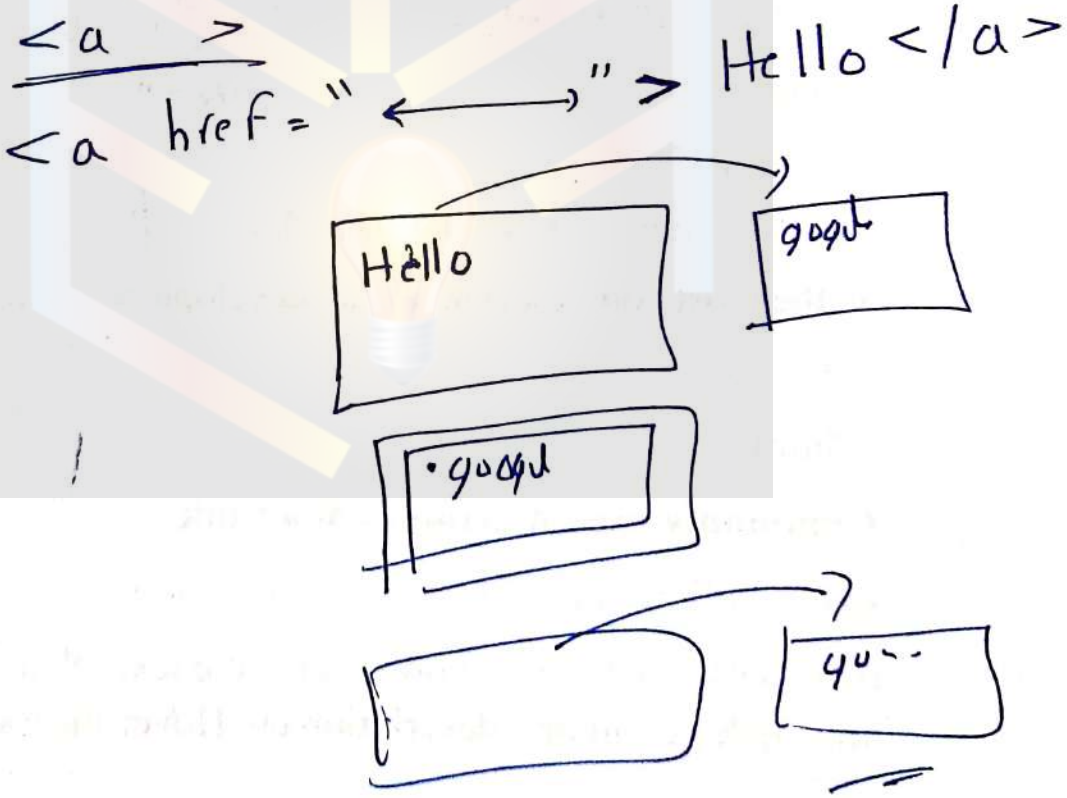
2. The very idea of *World Wide Web* is built around Hyperlinks. All day to day activities like Browsing, Surfing, Downloads depends upon links.

Types of Links: (10)



Header Links: using <link> element within the <head> element. **Anchor Links**: using anchor element<a> , within <body> element

We will study in detail about ANCHOR LINKS <a> in this lesson.



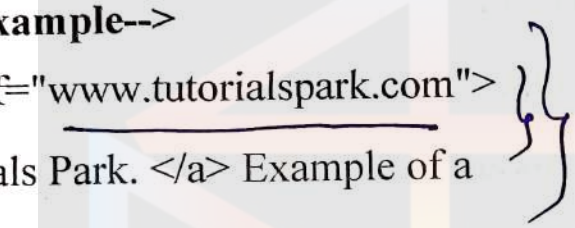
How to make links in html?

1.Any text can be transformed into a hyperlink by encapsulating it within anchor tag <a> . 2.The attribute href contains the URL of the webdocument,to which the clickable text links. Syntax:

 Related text.

<!-- Example-->

 Tutorials Park. Example of a Link:



```
<!DOCTYPE HTML>
<html>
<body>
<a href="http://www.tutorialspark.com">Tutorials Park </a>
is the most comprehensive web development website on the internet.
</body>
</html>
```

Commonly used Attributes of a Link.

href: Defines the URL or the location to which the link is created.

title: Title contains a short descriptive text related to the link,such as authors, description etc.Hover the tool tip over the

link to see the title.

id : To assign an unique identifier to the link.Two elements on same page cannot have the same id.

class: To assign an identifier to an element,but multiple elements can have the same class name unlike attribute id.

HTML Link Target

Link element attribute "target" denotes the target of the browsing context(i.e Tab, new Window, iframe) in which the link should open.

target: "⟶"

_blank

Target attribute Values:

_blank: To open the webpage in a new Tab or Window.

self: To open the clicked document in the frame as the current one.

parent: To open the Web document in the parent frameset of the current frame. If no parent available then it behaves as _self.

top: To open the web document in the full window removing all other frames.

<frame>: To open the webdocument in the specified or named frame EX:

```
<a href=" " target="_blank ">
Follow us on facebook for
new updates. (opens in new
Window)</a>
```

Absolute and Relative Links

1. Absolute link URLs contain entire Destination address, including the Protocol(http,https,ftp etc) and domain name. The Url doesn't change no matter where the webpage it appears on is kept. Hence, its called Absolute Links.

2. Relative Links do not contain the entire Destination address. Its address is relative w.r.t to the Document on which is appears.

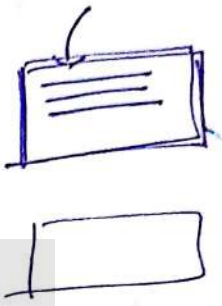
a. The Url http://www.tutorialspark/html/html5_links is an absolute link.

b.The Url ./html/html5_links is a relative link w.r.t to a document on the same website.

c.Relative links will appear broken if the address of the webpage on which the link appears changes.

EX:

```
p> This one <a
href="http://en.wikipedia.org/wiki/HTML5">he
re </a> is an Absolute hyperlink.</p>

<p> While this one <a
href="HTML5_Video.php">here</a> is
a relative hyperlink.</p>
```



Images

Images can improve the design and the appearance of a web page. The image is inserted into the display of the document by the browser.

The <IMG> Tag

Building on this idea of attributes I can introduce you to the IMG tag. To insert an image in a page you can use the following HTML snippet – I will explain each attribute in turn.

Image Formats

The two most common methods of representing images are the Graphic Interchange Format (GIF) and the Joint Photographic Experts Group format. Most contemporary browsers can render images in either of these two formats. Files in both formats are compressed to reduce storage needs and provide faster transfer over the Internet.

GIF uses 8-bit color representations for pixels, allowing a pixel to have 256 different colors. Files containing GIF images use the .gif (or .GIF) extension on their names. GIF images can be made to appear transparent.

The JPEG format uses 24-bit color representations for pixels, which allows JPEG images to include more than 16 million different colors. Files that store JPEG images use the .jpg (or .JPG or .jpeg) extension on their names.

Portable Network Graphics (PNG) is a good replacement for both GIF and JPEG because it has the best characteristics of

each (the possibility of transparency, as provided by GIF, and the same large number of colors as JPEG). One drawback of PNG is that, because its compression algorithm does not sacrifice picture clarity.

The <img /> tag

In HTML, images are defined with the <img> tag.

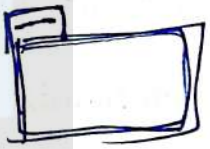
The src Attribute

This stands for 'source' – i.e. where is your image file? This can be an image file on your own website, or on another website, or on a CDN (Content Delivery Network).

If you are embedding an image from your own site, you can specify the full URL to the image or, as in this example, a path relative to the root of your website. If you are using files on your own computer with these HTML5 tutorials, then you can either use the full folder path to the image, or you can use a path relative to where your .html file is.

This makes your options something like this:

src="https://www.domain.com/images/my-company-logo.png" – the full URL to the image



```
<!DOCTYPE html>
<html>
  <head>
    <title> img test </title>
  </head>
  <body>
    <img src="xyz.jpg" alt="image"
    width="500" height="500px">
  </body>
</html>
```

test

px px

img
img

src="https://www.not-my-website.com/images/their-company-logo.png" – a URL to an external site's images

src="images/logos/my-company-logo.png" – take the location of my .html file, find a folder in the same location called 'images', then look for a folder in 'images' called 'logos', then find 'my-company-logo.png'

Valid File Formats

There are quite a large array of image formats these days. However, the most common ones used in an <img> tag are

GIF: hsa a limit to the number of colours you can use, but supports animation and transparency JPG or JPEG: no limit on the number of colours but does not support transparency.

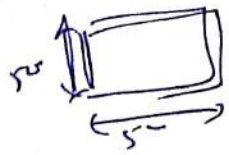
PNG: creates a high quality image, which can lead to a high file size, but supports transparency. There are other formats but these are enough to get you started.

Embedding Images From Other Websites

Before you consider doing this, please make sure that, first and foremost, **you have the right** to use that image. Reaching out to the site to ask permission would be a good starting point.

Another consideration is that their server may prevent you doing this. Embedding another website's images is called **hotlinking** and uses another person's bandwidth. For that reason always

look to host the images on your own web hosting, or subscribe to a CDN.



The width and height Attributes

Sounds obvious, but this sets the height and width of your image. This is measured in , for but you can equally specify a , the element.

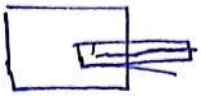
If you do not specify a height and a width, your browser will attempt to load the image and the size it has been created. I say 'attempt' because CSS can limit how large an image can be. Also another limiting factors might be the width of a table cell. So you are better to take control of your page and define the height and width of your image.

The alt Attribute

alt is short for 'alternative' and should be used as a text description of your image. This serves a couple of purposes:

Accessibility: Providing a text description of your image will help those who browse either with images turned off, or with screen readers (for those with eyesight challenges) to understand what that image is.

SEO: Search engines use the alt attribute to help determine the nature of the image and helps those images to show up for relevant queries in image search results. Carefully implementing this can also influence how the page the image is on to rank for relevant terms.



The title Attribute

The title attribute will display a 'tooltip' when someone hovers their cursor over the image. Again this should be descriptive, but best to keep this shorter than the alt attribute as tooltips tend to flash up for only a few seconds.

The align Attribute

The align attribute determines where on your page the image will sit. If you do not specify an align attribute, then it will align to the left of your page but your text will not wrap around your image.

You can specify **left, right, or center** but support for center is patchy and cannot really be relied on. I will cover in the CSS Tutorials how to force this center alignment.

If you specify left or right then your image will sit on the appropriate side of your page and **your text will wrap around** the image, at least within the limit of the size of the browser window loading the page.

hspace and vspace

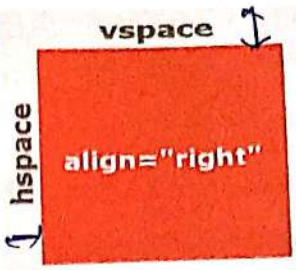
These two attributes give your image space, so that there is a gap between the image and the elements around it. Here's a demonstration of how align, vspace and hspace work:

Padding

Because align has been specified, the image sits to the right of the page, and text like this is forced to wrap around it.

hspace being specified allows some space between where the text ends and where the image begins.

The vspace ensures other page elements are spaced better above and below the image.



avid
12
vide

Multimedia

Multimedia on the web is sound, music, videos, movies, and animations. Multimedia comes in many different formats. It can be almost anything you can hear or see. Examples: Images, music, sound, videos, records, films, animations, and more.

Web pages often contain multimedia elements of different types and formats. The most common way to discover the type of a file, is to look at the file extension. The first web browsers had support for text only, limited to a single font in a single color.

SMTP

<audio src="..." ></audio>

HTML5 has introduced two new multimedia tags, AUDIO and VIDEO, for displaying the audio and video streams on a Web page.

You can play the multimedia files, which are stored in your local computer, on the Web page by specifying their location. The src attribute is used to specify the multimedia file to play it on the Web page.

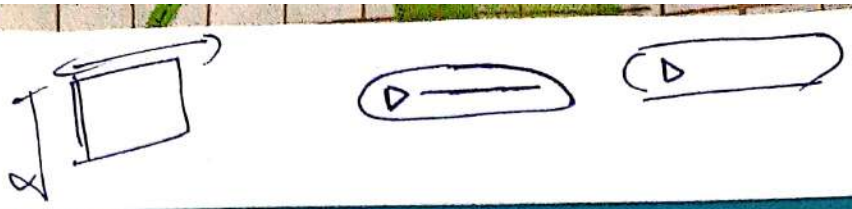
If the Web browser does not support AUDIO and VIDEO tags, then the text defined between the starting and the closing tags of these tags are displayed on the Web page.

Attributes of AUDIO Tag

The AUDIO tag of HTML5 supports only three audio file formats i.e. .oog, .mp3, .wav Following table shows the attributes of the AUDIO tag

Attribute	Description
autoplay	Plays the audio file as soon as the Web page loads
controls	Displays the controls on the Web page, such as play and pause buttons
loop	Replays the audio file
preload	Specifies whether the audio file is preloaded on the Web page or not
src	Provides the location of the audio file to play

<audio src="audio.mp3" autoplay="true" controls loop="3" >



Attribute	Description
audio	Controls the default state of the video's audio channel
autoplay	Plays the audio file as soon as the Web page loads
controls	Displays the controls on a Web page, such as play and pause buttons
height	Specifies the height of the VIDEO tag
loop	Replays the video file
preload	Specifies whether the video file is preloaded on the Web page or not
poster	Provides an image to be displayed when the video file is not available
src	Provides the location of the video file to play
width	Specifies the width of the VIDEO tag

You can use the VIDEO tag to display a video file on the Web page. The VIDEO tag supports the .gov and .mp4 file formats.

Following table describes attributes of the VIDEO tag

You can also use the SOURCE tag within the opening and the closing tags of the VIDEO tag to provide the source of the video file.

The SOURCE tag is used in a situation when the location of the video file is not confirmed. In this case, the VIDEO tag plays the first video file located in the specified path. The following code snippet shows the use of the VIDEO tag :

```
<VIDEO src="video.ogv" autoplay="true" loop="3" controls>
```

</VIDEO>

In the above code snippet, we have defined a video.ogv file in the src attribute. We have also set the autoplay attribute to true, which implies that the video file start playing as soon as the Web page loads. the loop attribute is set to 3, which implies that the video file will be played three times. In addition, the controls attribute displays the controls on the video player.

document in the full

body of the window framename - Opens the

linked document in a named frame

Lists (13)

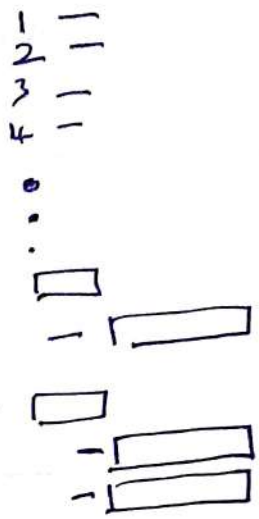
List are used to group related contents together in a structured manner making content easy to read and understand.

Types of Lists

Ordered List: To group a set of related items in a specific numbered order. Unordered List: To group a set of related items in no specific order

Definitions List: To group a set of related terms and their definitions.

Nested List: To create a list within another list(i.e nested).



Ordered List

1. An Ordered list has each item numbered, this is useful to provide sequential instructions.Ordered list is used extensively in formal documents.

2. The Ordered element < ol > encapsulates the complete list, and list element < li > encapsulates each of list items .

List Example

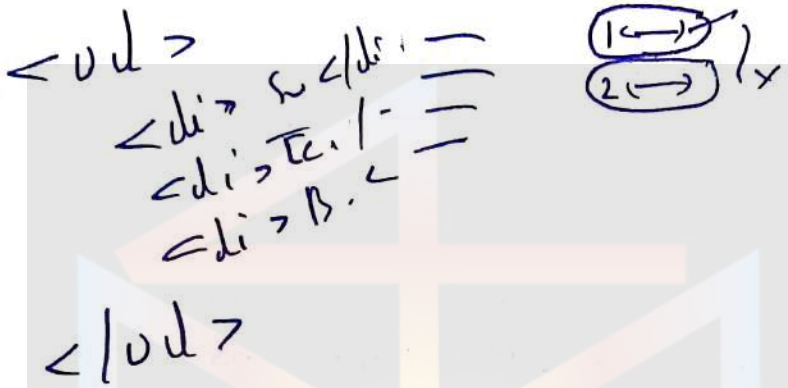
```
<ol>
<li>Macbook Air</li>
<li>iPhone</li>
<li>iPad</li>
<li>iMac</li>
</ol>
```

Unordered List

1. Unordered list is similar to ordered list but the numerals are replaced with bullet points.

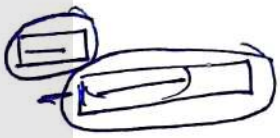
2. Unordered list is created within unordered tag < ul > and similar to ordered list ,list items are placed within list tag < li > .

3. Unordered list attribute " list-style-type" property can be used to replace bullet points with disc, circular and square bullets.



List Example

```
<ul>                    <!-- Unordered Element */-->
<li>BMW.</li>
<li>MERCEDES.</li>  <!--list tag -->
<li>PORSCHE.</li>
<li>AUDI.</li>
</ul>
```



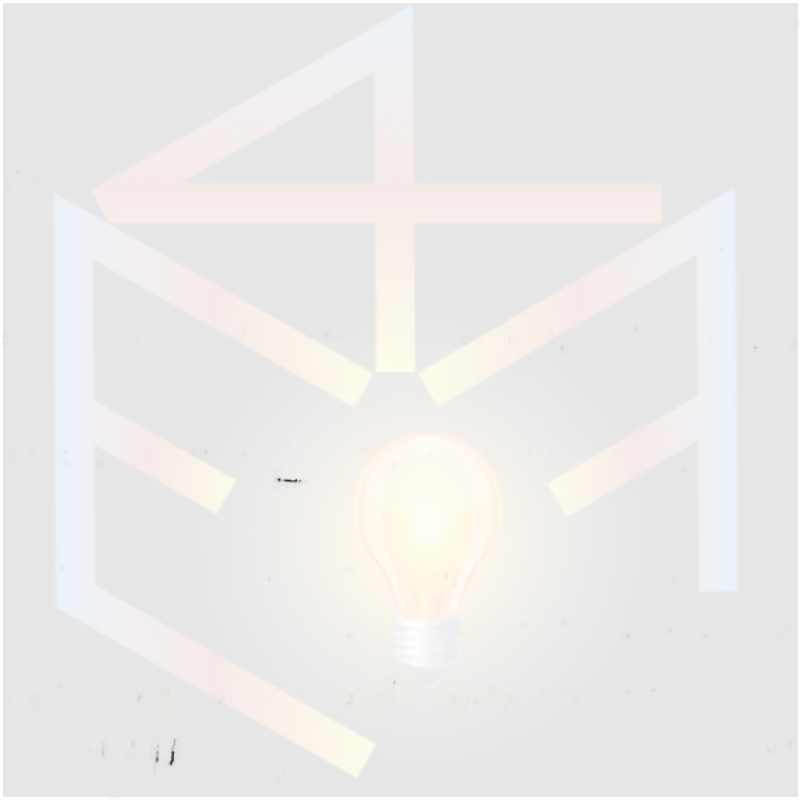
Definition List

1. The Structure of definiton list is a term followed by its description or definition.
2. Its created with a Definition list tag < dl > encapsulating the complete list .
3. Within each < dl > tag you have a pair of definition term <dt> and definition description element <dd>.
4. <dt> contains definition term, while <dd> contains the definition or description.

List Example

```
<dl>
<dt>iPAD</dt>
<dd>- A family of Tablets by APPLE Inc.</dd>
<dt>iPhone</dt>
```

<dd>- A Series of Smart Phones by Apple.</dd>
<dd>- First Launched in USA. </dd>
<dt>Macbook Pro</dt>
<dd>- Macintosh Portable computers from Apple.</dd>
<dt>Windows</dt>
</dl>



Nested List

1. Nested list is created by inserting a list within a list.The Sole idea behind this is to create a sublist for a main list item .

3. The Sublist list is put inside the

element of the main list . Example of

Nested List

```
<ul>
<li>Linux</li>
<li>Windows</li> // main list item.
<ul>
<li>Windows 95.</li>
<li>Windows 98.</li>//sublist.
<li>Windows XP.</li>
<li>Windows 7.</li>
</ul>
</li>
<li> Mac OS X</li>
</ul>
```

- Linux
- Window
  - W 95
  - W 98
  - W XP
  - W 7
- MacOS

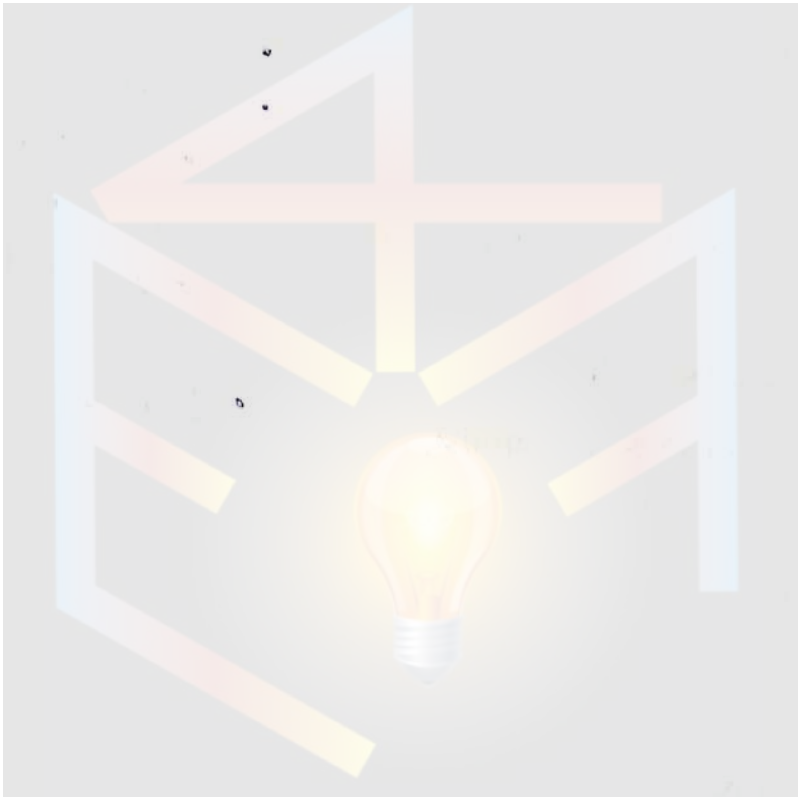
ordered List: Starting with user defined order

```
<ol start="4">
```

4
5
6
7
8

```
<li>Macbook Air</li>
<li>iPhone</li>
<li>iPad</li>
<li>iMac</li>
</ol>
```

4 M
5 1
6 1
7 1



Tables (4)

In HTML Table will be created by using <table>table data goes here..</table> tag. We know that table contains Rows and Columns, those are defined with tr and td.

<tr> stands for Table Row which is used to make a Row.

<td> stands for Table Data that is used to make a ~~Column~~. Table heading can be defined by using <th>Name</th>

Cellpadding and Cellspacing is used to adjust the white space in table cell. Cellspacing defines the width of the border. cellspacing="0" cellpadding="15" Cellpadding represents the distance between cell borders and the content within.

<caption> Books Information</caption> tag will serve as a title and show at the top of the table.

Example table

```
<table>
 <tr>
  <th>Month</th>
```

Month	Σ
Jan	1000

x	y
10	20

x y

10 z

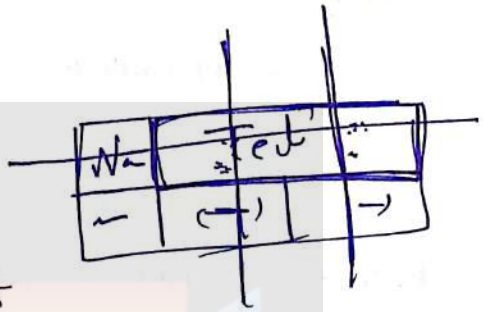
border = "1px"

```
<table>
<tr>
<th> x </th>
<th> y </th>
</tr>
<tr>
<td> 10 </td>
<td> 20 </td>
</tr>
</table>
```

```
  <th>Savings</th>
 </tr>
 <tr>
  <td>January</td>
  <td>$100</td>
 </tr>
</table>
```

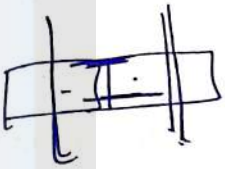
Col span : (column span)

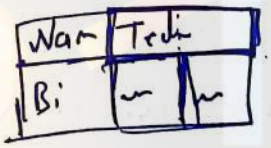
Row span : (Row span)



Example:

```
<table border= "1px">
 <tr>
  <th>Name</th>
  <th colspan="2">Telephone</th>
 </tr>
 <tr>
  <td>Bill Gates</td>
  <td>55577854</td>
  <td>55577855</td>
 </tr>
</table>
```

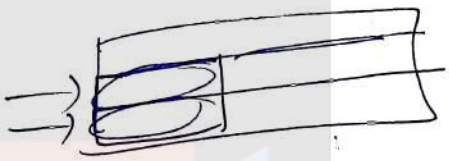




Name	Telephone	
Bill Gates	55577854	55577855

Row span

```
<table border= "1px">
 <tr>
  <th>Name:</th>
  <td>Bill Gates</td>
 </tr>
 <tr>
  <th rowspan="2">Telephone:</th>
  <td>55577854</td>
 </tr>
 <tr>
  <td>55577855</td>
 </tr>
</table>
```



Name:	Bill Gates
Telephone:	55577854
	55577855

Forms and forms styling

15

Forms (16)

HTML forms are required, when you want to collect some data from the visitor.

ex: registration, login, feedback etc..

uname = ramu

uname +

components / tags

<form>
⋮
</form> } indicating form

<input> } Indicating i/p (no closing tag)

input's are of multiple types

i) text :- collecting an text (like name) (1 line)

<input type="text" name="uname">

ii) email :- collecting an email (@ mandatory)

<input type="email" name="email">

iii) password :- collecting password (••••••)

<input type="password" name="pass">

iv) Submit :- a button to submit

<input type="submit" value="text">

v) radio :- select only 1 item among many

<input type="radio" id="gender"
name="gender" value="male" />
Male
<input type="radio" id="gender"
name="gender" value="female" />
female
<input type="radio" id="gender"
name="gender" value="other" />
other

vi) Checkbox :- select 0 or more among many

<input type="checkbox" name="vechicle1"
value="Bike"> I have a Bike
<input type="checkbox" name="vechicle2"
value="Car"> I have a Car

→ To label the i/p, we use

<label> enter your name </label>

textarea :- to collect address / large data

```
<textarea name="review" row="4"
col="50">
</textarea>
```

ex in image

CSS (Cascading style sheet). (18)

→ It is a styling language which is used to describe the looking & document written in

```
<!DOCTYPE html>
<html>
<head>
 <title>Form in HTML</title>
</head>
<body>
     <h2>Registration form</h2>
     <form>
         <label>Enter your full name</label><br>
         <input type="text" name="name"><br>
          <label>Enter your email</label><br>
          <input type="email" name="email"><br>
          <label>Enter your password</label><br>
          <input type="password" name="pass"><br>
          <label>confirm your password</label><br>
          <input type="password" name="pass"><br>
          <br><label>Enter your gender</label><br>
          <input type="radio" id="gender" name="gender" value="male"/>Male  <br>
          <input type="radio" id="gender" name="gender" value="female"/>Female <br/>
          <input type="radio" id="gender" name="gender" value="others"/>others <br/>
           <br>Enter your Address:<br>
          <textarea></textarea><br>
          <input type="submit" value="sign-up">
  </form>
 </body>
</html>
```

Enter your nar

1. Create a form using HTML to accept the details of a student: Name of the Hotel, Items (checkboxes), Quantity, Rating (radio button), Feedback (allow multiple lines) and provision to attach any other file or image.

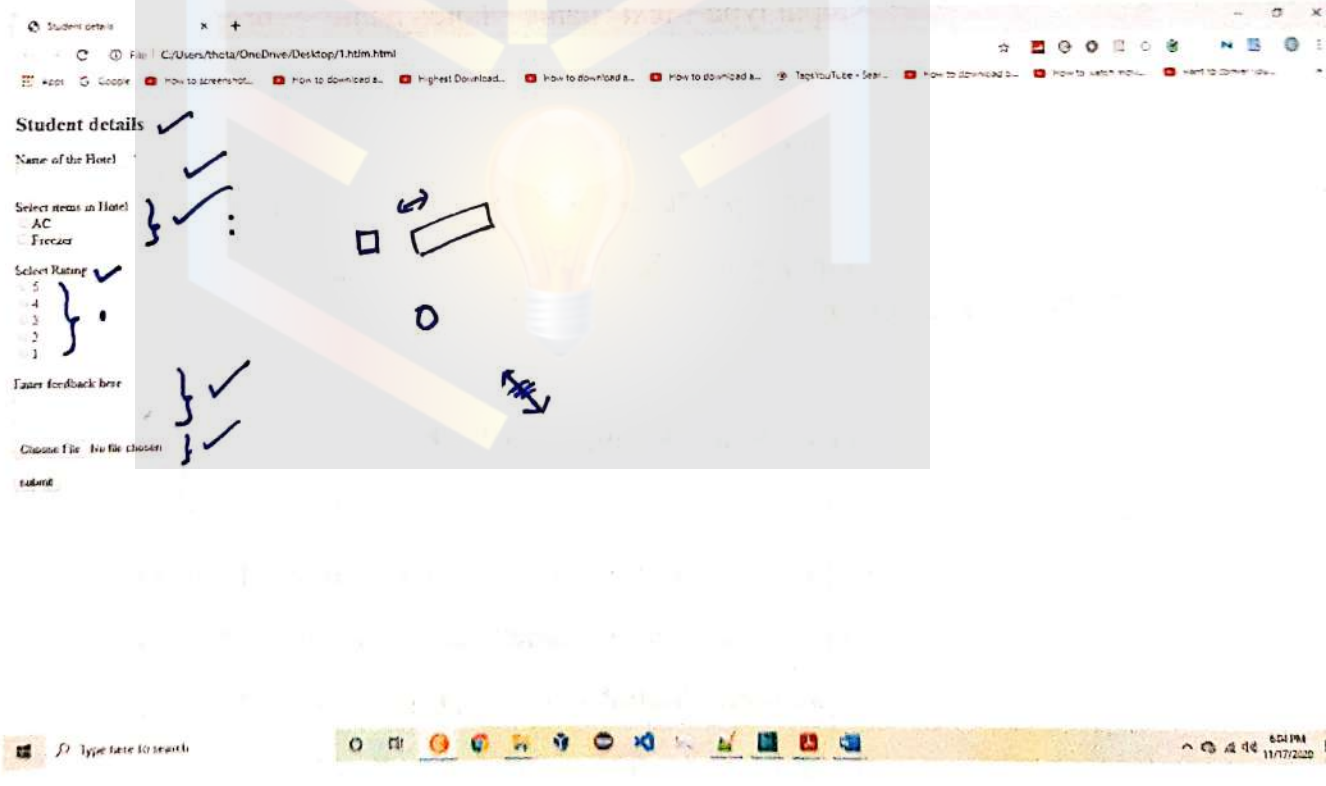
Ans.

```
<!DOCTYPE html>
<html>
    <head>
        <title>Student details</title>
    </head>
    <body>
        <h2>Student details</h2>
        <form>
            <label>Name of the Hotel</label><br>
            <input type="text" name="hotel_name"><br>
            <br>
            <label>Select items in Hotel</label><br>
            <input type="checkbox" name="item1" value"AC">AC<br>
            <input type="checkbox" name="item2"
value"AC">Freezer<br>
            <br>
            <label>Select Rating</label><br>
            <input type="radio" name="rating" value="5">5<br>
            <input type="radio" name="rating" value="4">4<br>
            <input type="radio" name="rating" value="3">3<br>
            <input type="radio" name="rating" value="2">2<br>
            <input type="radio" name="rating" value="1">1<br>
```

```
<br>
<label>Enter feedback here</label><br>
<textarea></textarea>
<br><br>
<input type="file" name="file"><br>
<br>
<input type="submit" value="submit">
```

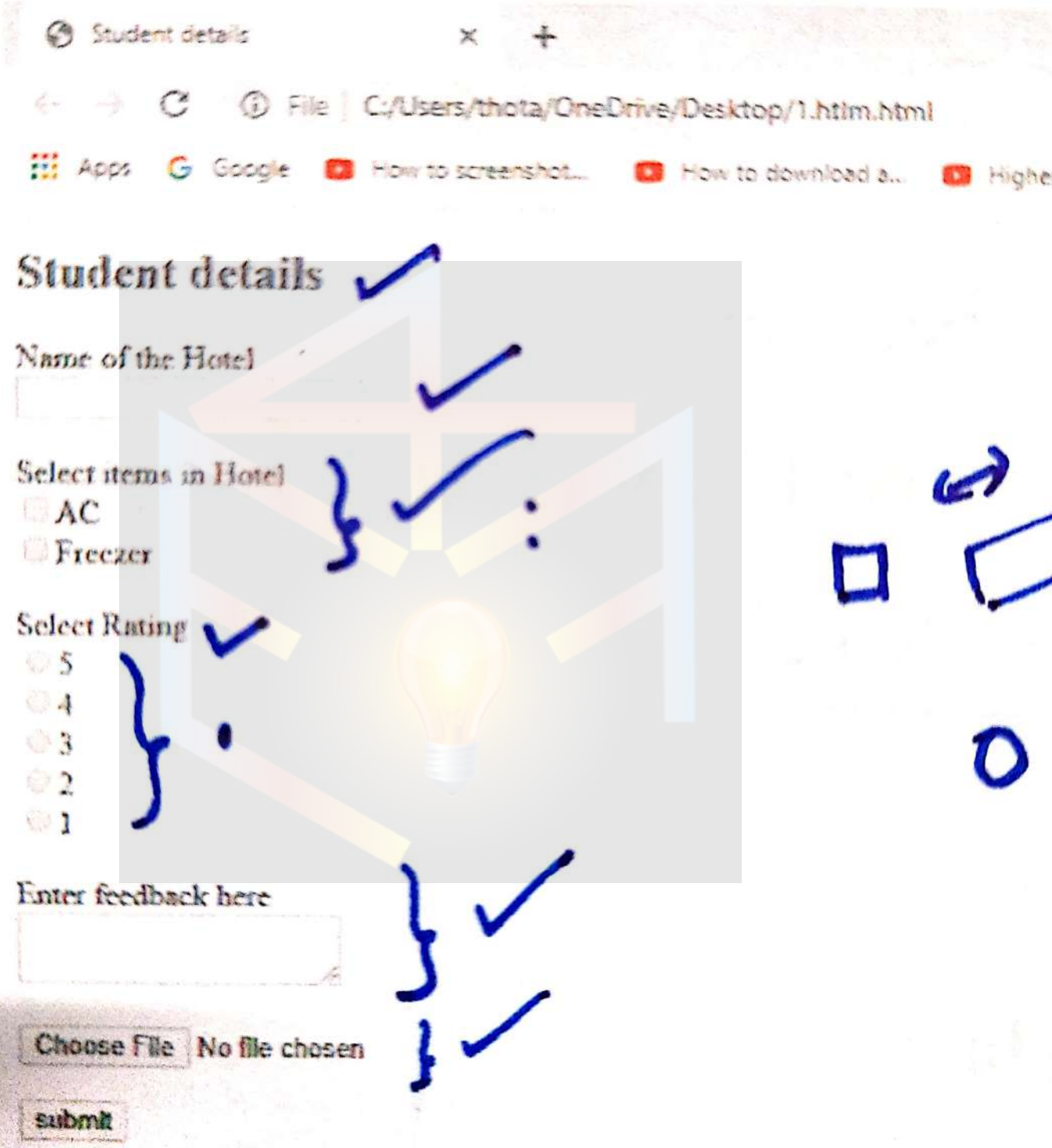
```
</form>

</body>
</html>
```

o/p:-



</html>

o/p:-

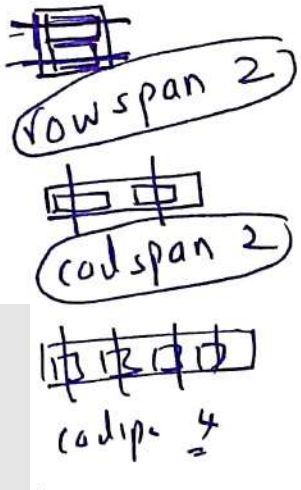


```
            <td>UVW</td>
            <td>4</td>
            <td>66</td>
        </tr>
        <tr>
            <td>ABC</td>
            <td>Science</td>
            <td colspan="2">5</td>
        </tr>
        <tr>
            <td colspan="4">Student details mentioned above</td>
        </tr>

    </table>

</body>
</html>
```

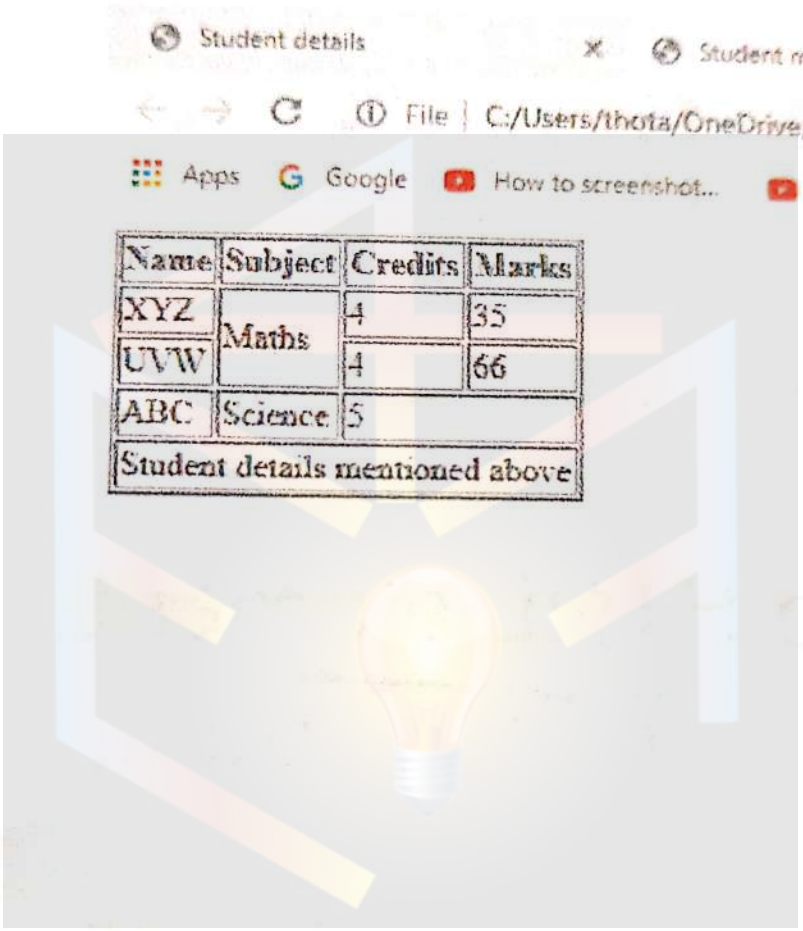
o/p:-

2. Write a HTML program to create a table as per the format below.

Name	Subject	Credits	Marks
XYZ	Maths	4	35
UVW		4	66
ABC	Science	5	
Students details mentioned above			




```
<!DOCTYPE html>
<html>
    <head>
        <title>Student marks and credits</title>
    </head>
    <body>
        <table border="1px">
            <tr>
                <th>Name</th>
                <th>Subject</th>
                <th>Credits</th>
                <th>Marks</th>
            </tr>
            <tr>
                <td>XYZ</td>
                <td rowspan="2">Maths</td>
                <td>4</td>
                <td>35</td>
            </tr>
            <tr>
```



col = 50 >
</textarea>

ex in image

CSS (Cascading style sheet). (18)

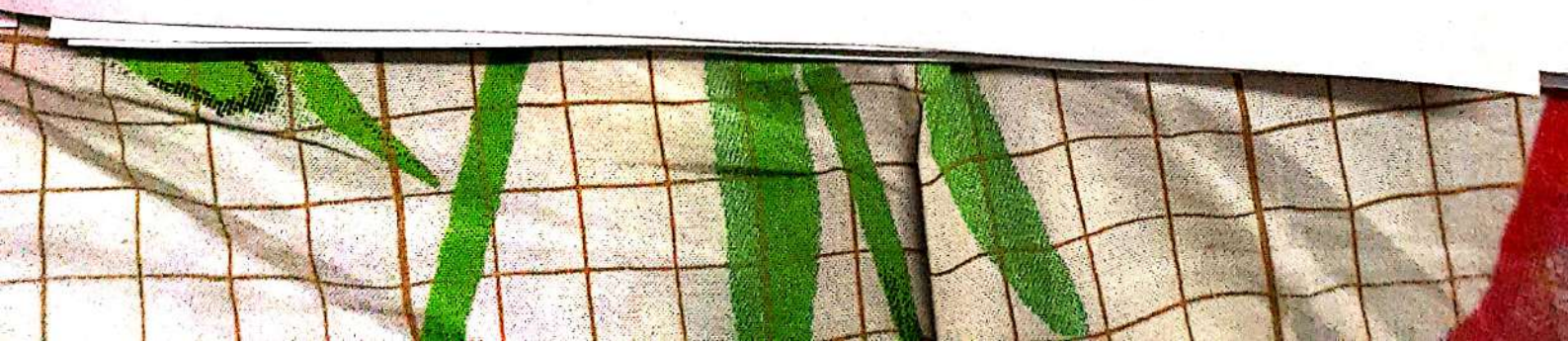
→ It is a styling language which is used to describe the looking & formatting of a document written in markup lang.

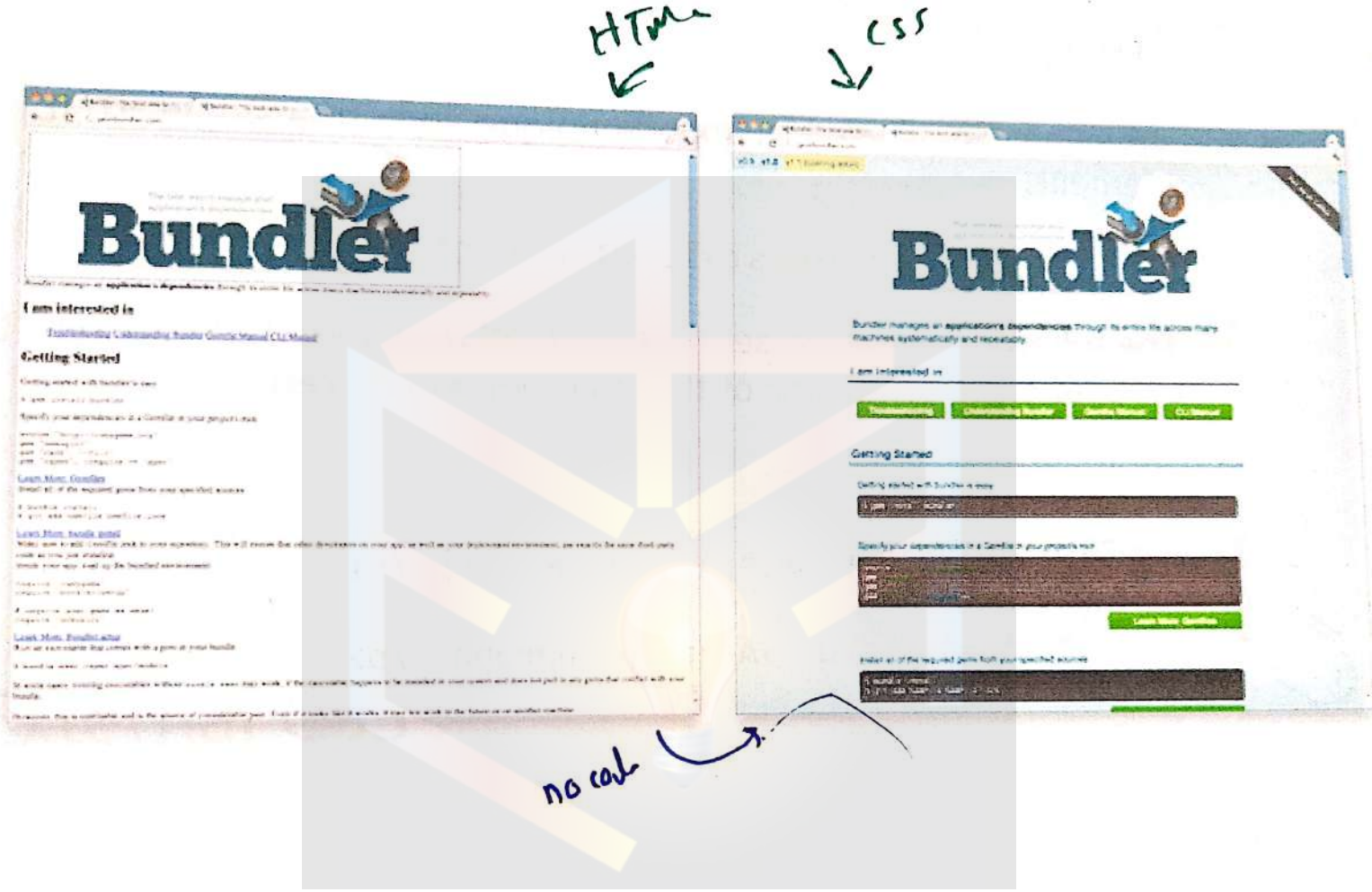
→ We can change the look of the website

css can be added in 3 ways

→ inline (inside html component)
→ Internal (inside head in same html file).
→ External. (external file)

ex: color, font-family, font-size, border, padding, margin, bag.col~





Using CSS

CSS can be added to HTML documents in 3 ways:

- **Inline** - by using the `style` attribute inside HTML elements
- **Internal** - by using a `<style>` element in the `<head>` section
- **External** - by using a `<link>` element to link to an external CSS file

The most common way to add CSS, is to keep the styles in external CSS files. However, in this tutorial we will use inline and internal styles, because this is easier to demonstrate, and easier for you to try it yourself.

Inline CSS

An inline CSS is used to apply a unique style to a single HTML element.

An inline CSS uses the `style` attribute of an HTML element.

The following example sets the text color of the `<h1>` element to blue, and the text color of the `<p>` element to red:

Example

```
<h1 style="color:blue;">A Blue Heading</h1>

<p style="color:red;">A red paragraph.</p>
```

Internal CSS

An internal CSS is used to define a style for a single HTML page.

An internal CSS is defined in the <head> section of an HTML page, within a <style> element.

The following example sets the text color of ALL the <h1> elements (on that page) to blue, and the text color of ALL the <p> elements to red. In addition, the page will be displayed with a "powderblue" background color:

Example

```
<!DOCTYPE html>
<html>
<head>
<style>
body {background-color: powderblue;}
h1   {color: blue;}
p    {color: red;}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

External CSS

An external style sheet is used to define the style for many HTML pages.

To use an external style sheet, add a link to it in the <head> section of each HTML page:

Example

```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="styles.css">
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

The external style sheet can be written in any text editor. The file must not contain any HTML code, and must be saved with a .css extension.

Here is what the "styles.css" file looks like:

"styles.css":

```
body {
  background-color: powderblue;
}
h1 {
  color: blue;
}
p {
  color: red;
}
```

```
6   <!-- External/linked CSS file -->
7   <link href="styles.css" rel="stylesheet" type="text/css" />
8
9   <!-- Global/embedded style statement -->
10  <style>
11      p {
12          font-size: 24pt;
13      }
14  </style>
15  </head>
16
17  <body>
18  <!-- Local/inline style -->
19  <p style="font-family: Helvetica;">Example of 3 “cascading
    style sheets.”</p>
20  </body>
21  </html>
```