

↲ WWW is a sharing system for informations;
 ↲ Internet is a medium in which all info are floating.

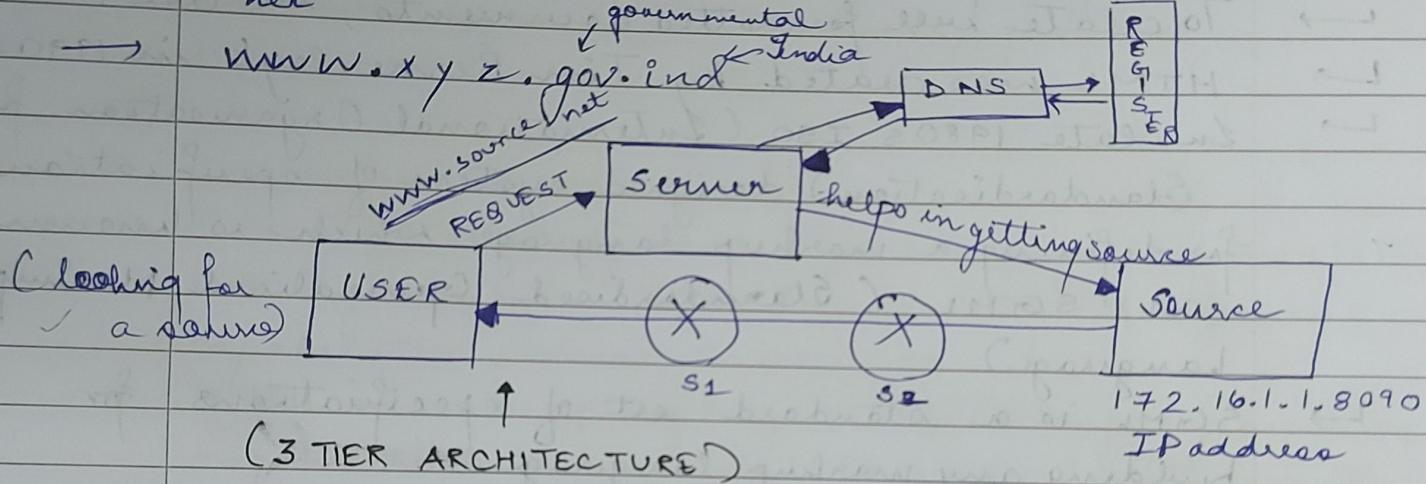
• com → for commercial sites

• edu

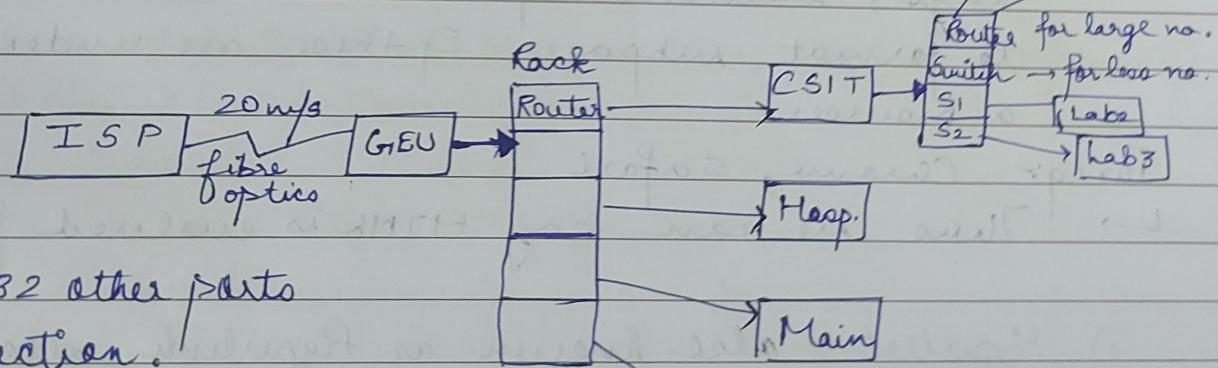
• org

• net

→ www.x.y.z.gov.in → governmental India



↳ ISP → Internet Service Provider



- * Router has 32 other ports for connection.
- * Switch has less than 32 ports

Hub → switch of small level.

Introduction to HTML.

- ↳ HTML → HyperText Markup language.
- ↳ To share info world wide, we need internet
- ↳ To share info over internet, we need to create or design a website
- ↳ Website is collection of interlinked webpages.
- ↳ Webpage is a well formatted document used to share info.
- ↳ To create well formatted documents, we use HTML.
- ↳ HTML was created by Tim Berner Ley's in 1990s
- ↳ In late 1980's ISO (International Organisation for Standardisation) designed system of specification for defining a markup language which is known as SGML (Standardised Generalised Markup language).
- ↳ SGML is a standard set of specifications for building any markup language.
- ↳ A markup is a lang. which consists set of tags, elements & attributes which is used to format webpages & these are understood by a browser.

For eg:- Chrome, safari

- ↳ Thus we can say HTML is evolved from SGML.

- i) Hypertext → also known as hyperlink, allows us to navigate within same / diff. webpages.
- ii) Text → actual info/content to be displayed

- iii) Markup → refers to ~~text~~ tag/ control info to format the content. HTML tags are used by browsers.
To format content on webpage.

- iv) Browser → A c/w for interpreting & displaying HTML files.
↳ Markup lang. is a tag based lang.

FEATURES OF HTML.

- i) Simple & small lang →
- ii) A markup language
- iii) Powerful lang for formatting web pages.
- iv) Supported by multiple browsers/ platforms.
- v) Flexible & Maintainable

LIMITATION OF HTML →

- Through HTML if we design any page they are static not dynamic. HTML has limited designing capabilities, thus we use CSS
- HTML is a scripting lang ie it has no programming capabilities thus we use JS which enables us to create & design dynamic & interactive webpages.

BASIC HTML TAGS → for formatting textual information

<html> // paired tag
<head> // head

<title> My first HTML Document </title>

</head>

<body>

(strong) or Hello World

Bold

Break line

 or <i>Hello World</i>

Italic

Break line

<u> Hello World </u>

Underline

Break line

Hello World)

Nested Tag

 <i> Hello </i>

<i> <u> Hello </u> </i>

<i>

<u>

Hello World.

</u>

</i>

.

Maintaining
indentation

Q

Write $C_6H_{12}O_6$ & $(a+b)^2 = a^2 + b^2 + 2ab$.

→

<html>

<head>

<title> My Document </title>

</head>

<body>

$C ₆ H₁₂ O₆$

subscript //

$(a+b)^{sup}2 </sup> = a^{sup}2 </sup> + b^{sup}2 </sup> + 2ab$

← "Graphic Era"

"
"

or

// quotes <q> Graphic Era </q>

Our sections are best

Imp. Q.

Design "Best"

Our sections are <mark> Best </mark>

// Strike <strike> Hello </strike>

or

 Hello

`<s>` → For striking (Strike Tag)

g)

*
* *
* * *
* * * *
* * * *

Display in HTML. (Using symbolic expression).

↳ `<html>`

`<head>`

`<title> My Doc </title>`

`</head>`

`<body>`

`<pre>` // preformatted tag

*

* *

* * *

* * * *

* * * * *

`</pre>`

`<body>`

`<html>`

Hello Hello Hello Hello

↳

`<body>`

`
`

`<big> Hello </big>`

`
`

Hello

`
`

`<small> Hello </small>`

Heading Tag (H₁-H₆)

<body>

<h₁> Big </h₁>

<h₂> Gurl </h₂>

:

<h₆> Small </h₆>

Inline Tag & Block level Tags.

→ <body>

<h₁> Hello</h₁> Bye. // Block level Tag.

Our sections are h₁ h₂ h₃ h₄ h₅ h₆ // Inline → No margins

off

Hello

Bye

Our Sections are h₁ h₂ h₃ h₄ h₅ h₆

Block level

Inline elements w/ Block Elements

Inline level Tag

Block level Text

- o) This never starts from a new line.
- o) This will cover only the space as bounded by tags in HTML elements.
- o) These do not have top & bottom margin.
- o) eg:- <a>, ,
, <button>, <code>, , <input>, , <textarea>
- o) This always starts from a new line
- o) This covers space from left to right as far as possible.
- o) These have top & bottom margin.
- o) eg:- <body>, <Article>, <aside>, <div>, <fieldset>, <figcaption>, <figure>, <form>

Types of Tags

- i) Paired
- ii) Unpaired
- iii) Block Level
- iv) Inline Level

30.1.28

Align attribute → By default "left"

<body>

<body>

<h1 align = "right"> HTML Text </h1>

<h2 align = "Center"> HTML Text </h2>

<h2 align = "left"> --- . </h2>

<h1>

</h1>

</body>

 Tag → paired Tag, Inline Level Tag, Text Formatting Tag

default
 $e = "3"$

 <body>

 HTML Text

size will be changed HTML in

 HTML

 HTML

</body>

Colour Names & Codes

<body>

 HTML

color = "#ff0000" → red.

color = "#00ff00" → green

color = "#0000ff" → blue.

color = "#ffffff" → white

color = "#000000" → black

color = "#00ffff" → cyan green + blue

color = "#ffff00" → yellow red + green

color = "#ff00ff" → magenta red + blue

#808080 → grey

o Binary → 0, 1 = $\text{Base } 2$

o Decimal → 0-9 = 10

o Hex → 0-9, a, b, c, d, e, f = 16

R G B

8 bits

Red = $2^8 = 256$

0-255

255	0	0
0	255	0
0	0	255

8 bits

Green = $2^8 = 256$

" "

8 bits

Blue = $2^8 = 256$

" "

255	0	0
0	255	0
0	0	255

ff0000

$$\begin{array}{l} \boxed{f} \boxed{f} \boxed{0} \boxed{0} \boxed{0} \\ \rightarrow 0 \times 16^1 + 0 \times 16^0 = 0 \end{array}$$

$$\rightarrow 0 \times 16^1 + 0 \times 16^0 = 0$$

$$\rightarrow f \times 16^0 + f \times 16^1 = 250 + 15 = 255$$

→ HTML Background Color attributes <bg>

<body & bgcolor = "#808080" & text = "red">

Red HTML page

Yellow HTML page

Red HTML page

</br>

→ HTML Body Background attribute

<body background = "cat.jpg" > } Repetation of
</body> images in webpage
in x & y column.

→ <body background = "cat.jpg"

no repeat style = "background-repeat: no repeat;" >

x column style = "background-repeat: repeat-x;" >

<p> → Paragraph Tag <Block Level>

<body>

<p> Para 1 </p>

<p align = "Right"> para 1 </p>

<p align = "Left"> para 2 </p>

<p align = "Center"> para 3 </p>

<p align = "Justify"> para 1 </p>

g) Convert a word document into html document

hr tag <Block level>

<hr>

<body>

<hr/>

<hr color = "red"/> </hr>

<hr color = "red" width = "100%"/>

<hr color = "red" width = "20%"/>

⇒ ↗ is interactive but Ef dynamic but pixel is not.

<hr color = "red" width = "20%" size = "10"/>
By default this line starts in "center"

<hr color = "red" width = "20%" size = "10" align = "right"/>

img tag <Inline> <Unpaired>

** Image should be in same folder where your webpage is.

→

→

→

<marque> Tag <Block level>

<body>

<marque> HTML Text </marque> // right to left

<marque bcolor = "red"> HTML </marque>

↳ We can't use align attribute instead use "width" <p align = left>

<marque bcolor = "yellow" width = "50%"/> HTML </marque>

Movement is known as behaviour. By default behaviour = scroll.

→ `<marque>`

"behavior = "scroll" > HTML </marque>

→ `<marque>`

"behavior = "slide" >

→ `<marque>`

"behavior = "alternate" > //bounce.

→ For changing direction By default "left".

→ `<marque>`

direction = "right" > Tag </marque>

`<marque>` "behavior = "slide" direction = "down" > Tag </marque>

`<marque>` "behavior = "alternate" direction = "up" > Tag </marque>

list Tag in HTML < Block Level >

↳

Ordered → list which has numberings and are in format.

↳

<body>

 coffee

 tea

 cold-drink

 juice

</body>

OUTPUT

1. Coffee

2. Tea

3. Cold Drink

4. juice

or

<body>

<ol type = "I">

 A

 B

</body>

I A

II B

→ <body>

<ol type="1" start="5">

 A

 B

 C

5 A

6 B

7 C

</body>.

O/P.

↳ Unordered list →

<body>

<ul type="disc">

 A

 B

 C

 .

O/P

• A

• B

• C

•

</body>.

- Anchor Tag → hyper means navigation or Jumping. If we want to navigate from one webpage to another we will create hyperlinks or from one section to another within same webpage. A hyperlink can be text, Image or any other section. To use hyperlink in HTML we use Anchor Tag.

→ < a > Tag -- < /a > → This is a paired tag and inline. It uses a special attribute hyperlink reference ie ' href'. This helps us to navigate to location within or outside webpage.

→ < html >

< head >

< title > First page < /title >

< /head >

< body >

This is first page < /br >

< a href = "Secondpage.html" > Go to second page

< /a >

< /body >

- target attribute → by default : "self" ie second page will open within the same tab (if we want to open a new tab then
< a target = " _ blank " > -- < /a >)

o) hyperImage → Images which are hyperlinkable

```
<html>
  <head>
    <title> Page </title>
  </head>
  <body>
    <a href = "About me .html">
      <img src = "myphoto.jpeg" />
    </a>
  </body>
```

- Page Navigation In HTML within same page.
↳ i.e. if we have 2 sections within the same page.

- <Within the page>

```
<body>
  <a name = "TopSection"></a>
  <p> Top Section </p>
  <p><a href = "#BottomSection"> Go to BottomSection</a></p>
    [ 25 times <br> ]
```

```
<a name = "Bottom Section"></a>
<p> Bottom Section </p>
<p><a href = "#TopSection"> Go to Top Section </a></p>
```

- <Outside the page>

Second page.

<body>

 Go to Top
Section of First page .

 Go to Bottom
Section of First page .

</body>

- Embed Tag → <embed>.

- ↳ like a universal tag which is used to access any kind of file in our webpage:
- ↳ It is an inline and unpaired Tag.

<body>

<embed src = "abc.jpeg" >

<embed src = "xyz.pdf" style = "width: 100vw ;
height : 100vh ;" >

◦ vw → viewport width

◦ vh → viewport height

- ↳ Anchor Tag behaves differently with different browsers

→ opening mailbox to send feedback through Anchor Tag

→ ``
Send Feedback ``

→ ``
Send Feedback ``

→ ``
Send Feedback ``.

~~map~~ tag → To create hotspot areas or clickable sections in an image we use image map tag.
With the help of this tag we can have more than one section clickable. Map tag is a paired tag. This uses area tag which is an unpaired tag.

`<body>`
`<map name = "Imagemap">`
`<area shape = "rect" coords = "x, y, x+width
y+height"`
`<area shape = "circle" coords = "centerx, centery, radius"/>`
`</map>`
``
`</body>`

g) WAP in HTML to create 4 equally divided clickable sections in an image using map tag

<body>

```
<map name="imagemap">
<area shape="rect" coords="0,0,200,200"
      href="www.google.com" target="_blank"/>
<area shape="rect" coords="200,0,400,200"
      href="www.google.com" target="_blank"/>
<area shape="rect" coords="0,200,200,400"
      href="www.google.com" target="_blank"/>
<area shape="rect" coords="200,200,400,400"
      href="www.google.com" target="_blank"/>
```

</map>

```

</body>
```

<body>

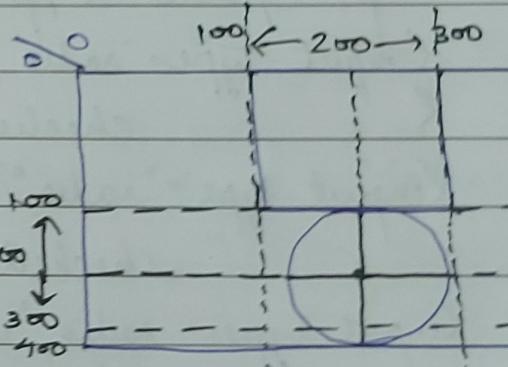
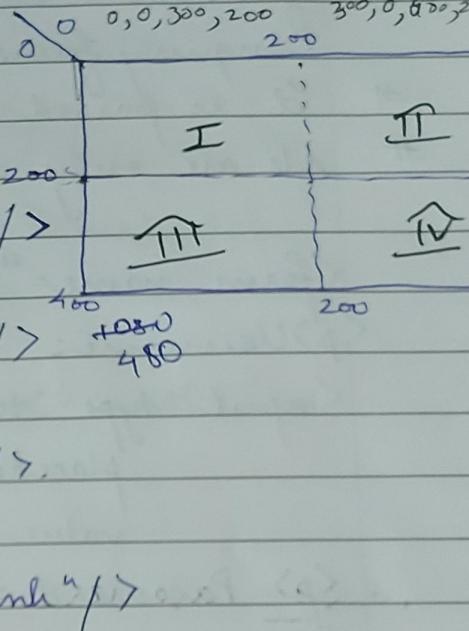
⇒ <map name="imagesmap">

```
<area shape="rect" coords="100,0,300,100"
      href="www.google.com" target="_blank"/>
<area shape="circle" coords="200,200,100"
      href="www.google.com" target="_blank"/>
```

</map>

```

</body>
```



FORMS IN HTML

- With the help of these we can create login form, sign in form, data entry, feedback form, online application & enquiry forms.
- # It is paired & block-level.
- # We are only submitting here, validation is not done.

```
<form name="frmDataEntry" action="processing.html" method="get">
<p> Username: </p>
<input type="text" name="txtUserName" id="txtUserName"
placeholder="Enter your Name" maxlength="20" required
value="" disabled /> // For validation required is used

<p> Password: </p>
<input type="password" name="pwd" id="pwd" placeholder=
"Enter your password" required />

<p> Gender: </p>
<input type="radio" name="genderRDB" id="genderRDB" value="M"
checked />
<input type="radio" name="genderRDB" id="genderRDB" value="F"
checked />
<input type="radio" name="genderRDB" id="genderRDB" value="O"
checked />

<p> Skills </p>
<input type="checkbox" name="checkSkill" id="checkSkill" value="C"
checked />
<input type="checkbox" name="checkSkill" id="checkSkill" value="C++"
checked />
<input type="checkbox" name="checkSkill" id="checkSkill" value="JAVA"
checked />
<input type="checkbox" name="checkSkill" id="checkSkill" value="Python"
checked />
```

Convert string into numeric datatype in form - e.g. Address Field.

map →

- OFF → to reduce scrolling
- virtual → according to user's choice in next line
- physical → as a wrap text

</p> Qualification </p>

<select name="sltqualification" multiple>

<option value="I"> Intern </option>

<option value="G"> Graduate </option>

<option value="PG"> Post Graduate </option>

</p> Address </p>

<textarea rows="10" cols="10" map="virtual">

</textarea>
.

Q) Write difference b/w use 'get' & 'post' method.

<input type="Submit" value="SUBMIT" />

<input type="Reset" value="RESET" />

If 'get' is used then there are security issues, password is reflected. So, we use "post".

HTML TABLE TAG <table>

It is a paired tag and block-level tag.

Marksheet

It consists of 3 subtags

S.No	Name	Marks	Result
1			
2			
3			
4			

◦ <th> → table heading

◦ <td> → table data (1 cell) combination of td = tr

◦ <tr> → table rows

<center>

<table border="1" cellspacing="0" cellpadding="10">

<caption> Marksheets </caption>

<tr> <td> <td> <td> <td>

<td> ^{colspan="2"} <td> ^{rowspan="2"} <td> ^(delete this) <td>

✓ delete 3 cells for

colspan. <td> <td> <td> <td>

<td> <td> <td> <td>

</tr> </tr> </tr> </tr>

Vertical alignment

align → 3 elements (left, right, center)

vAlign → (top, bottom, middle)

FRAMESET TAG IN HTML →

- o It is possible to divide a webpage into diff. sections of webpages. For doing so, we require frames
- o The browser can be divided into diff. sections by help of `<frameset>` tag & `<frame>` tag
- ✓ To use `<frameset>` tag, we need to omit `<body>` tag.
- o The 2 imp attributes within `<frameset>` tags are:
 - * rows
 - * cols

We can give comma (,) separated values in `rows` & `cols` either in pixel or percentage. No of values indicate no. of frames.

→ `<Frame>` tag is a subtag inside `<Frameset>` tag, `src` attribute w.r.t `<frame>` tag indicates a web page to be displayed in respective frame.

It is `<name>` attribute indicates name of frame in anchor tag.

- o It is an unpaired tag

frame1	frame2	frame3	frame4
<code><body></code>	<code><body></code>	<code><body></code>	<code><body></code>

`<html>`

`<head>`

`<title> HTML </title>`

Frame1	Frame2	Frame3	Frame4
<code></code>	<code></code>	<code></code>	<code></code>

P

Frame1

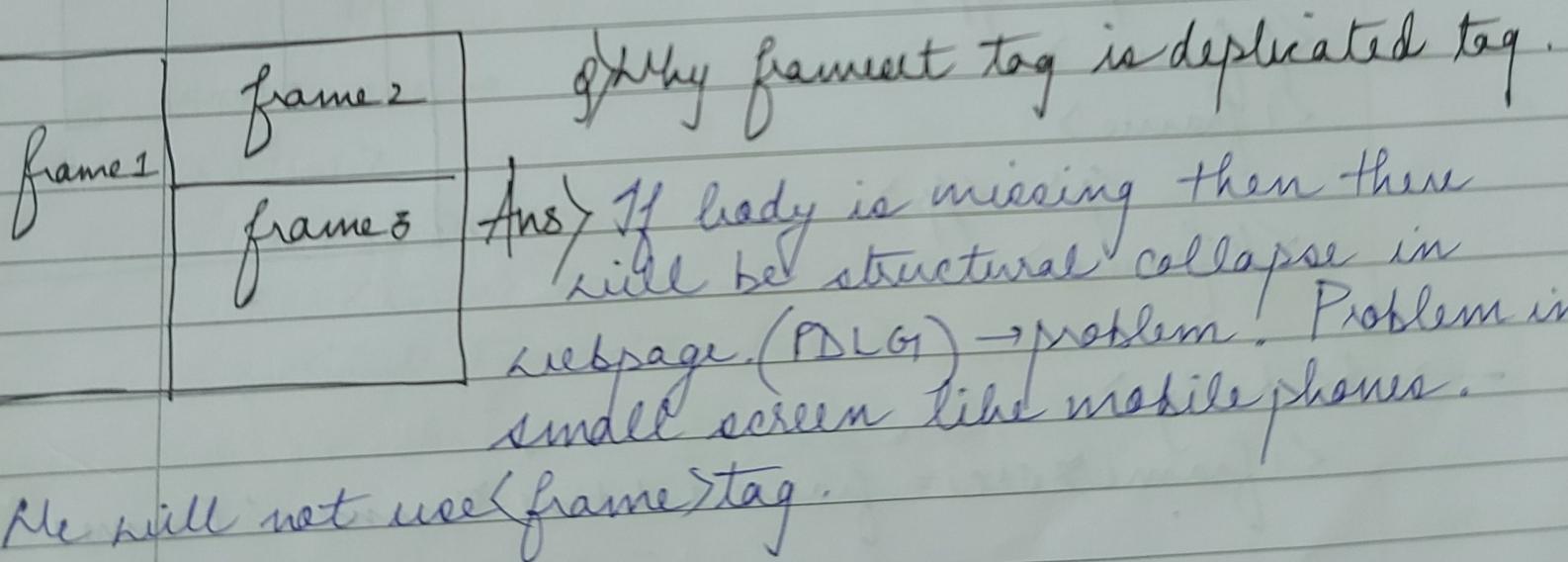
`</head>`

`<frameset cols="40%", 60%">` * rest of the space
`<frame name="Frame1" src="frame1.html"/>`
`<frame name="Frame2" src="frame2.html"/>`

```
<frameset name="Frame1" rows="10%, 40%, *">>  
  <frame name="Frame1" src="frame1.html"/>  
  <frame name="Frame2" src="frame2.html"/>  
  <frame name="Frame3" src="frame3.html"/>
```

Nesting of Frames

```
<frameset cols="10%, 40%, *">>  
  <frameset name="Frame1" src="frame1.html"/>  
    <frameset rows="80%, *">>  
      <frame name="Frame2" src=""/>  
      <frame name="Frame3" src=""/>  
    </frameset>  
  <frame name="Frame4" src=""/>  
Syntax Wrong !!
```

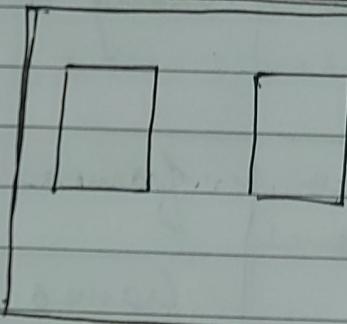


<iframe> Tag

- To include <frameset> we need to omit <body> tag, bcoz this breaks the structure of HTML document, also small screen devices do not support <frameset> tag. Hence it is deprecated or out of use.
- For an alternative, we use <iframe> tag. It is used to display local & remote web pages.
- It is used inside <body> tag. It is paired of inline-level tag.

```
<body>
  <iframe>
    <p> </p>
  </iframe>
  <iframe>
    <p> </p>
  </iframe>
```

OUTPUT



```
→ $<body>
  <iframe> <src= "frame1.html" name= "Frames">
    </iframe> <a href= "frame3.html" target = "frame1"></a>
    <iframe> <src= "frame2.html" >
    </iframe>
```

To change color we need to use bgcolor in <Frames1.html> Now we use href in Frames1 Click Me .

↳ IFRAME VS object vs EMBED

- These 3 tags are used to include files in HTML document.
- <Embed> &<Object> tags are static whereas <Frame> tag is dynamic tag.
- Because <frame> tag supports or easily updates itself with the updation of plug-ins or visual s/w config.

→ <object data = "example.jpg">

→ <embed src = "eg.jpg" />

→ <iframe src = "eg.jpg">

STYLE TAG →

<html>

<head>

<title> Document </title>

<style type = "text/css">

h1 {

 width = 100px;
 height = 100px;

 p {
 width = 100px;

 }

</style>

</head>

<body>

<h1>

<h1>

h1

OUTPUT

<p>

<p>

p

<h1>

<h1>

h1

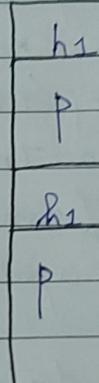
<p>

</p>

p

<div> → block level tag

```
<body>
  <div>
    <h1>   </h1>
    <p>     </p>
  </div>
  <div>
    <h1>   </h1>
    <p>     </p>
  </div>
```



) One webpage different sections of a webpage.

g For decorating a text we cannot use <div> tag. Instead we use tag.

XHTML (Ex HTML)

- 1) It is defined as XML application
- 2) It is a stricter & cleaner HTML.
- 3) It is compatible to HTML 4.01 & supported by all browsers
- 4) XHTML is used to define & organise the page content but not to format or style it.
- 5) XHTML uses colon :
 - a) Elements & attributes of ~~HTML~~ XHTML
 - b) Syntax of XML (Extensible Markup Lang).
 - c) It is an improved version of HTML.
 - d) It is originally a HTML document defined as XML application

Q) Why we need XHTML?

→ The following bad HTML doc. will work in most browsers but it does not follow actual HTML rules

```
<html>
  <head>
    <body>
      <p>
        <br/>
      </body>
    </html>
```

- But browsers running on hand-held devices which has small computing power and can't interpret bad markup language.
- HTML is designed to structure and display data while XML is designed to describe & structure data.
- XHTML specifies that everything must be marked-up correctly.

XHTML PAGE SYNTACTIC RULE →

1. XHTML elements must be properly nested - eg:-

<i> _____ </i> X
<i> _____ </i> ✓

2. XHTML elements must be closed always.

eg :-

<p>	_____	X
 	_____	X
	_____	X
<p> </p>	_____	✓
 	_____	✓

3. XHTML elements must be in lower-case.

4. XHTML elements must have an HTML root element - <html> which contains <head> & <body> tags.

5. Attribute names must be in lower-case.

6. Attribute values must be quoted " ".
7. The 'id' attribute replaces the 'name' attribute
8. XHTML DTD defines mandatory elements.
9. Attribute minimisation is forbidden.

STRICTLY CONFIRMING DOCUMENT →

- A strictly confirming XHTML Doc. is an XML Document that requires only the facilities described as mandatory in this specification. Such a document must meet all of the following requirements.
10. Must confirm to constraint expressed in one of 3 DTDs. (DTD are Strict, Transitional, Frameset) If following elements have prohibition applied to all depths of nesting.
- a) Anchor Tag: must not contain other anchor elements
 - b) Pre Tag: must not contain img, object, big, small, sub or sup.
 - c) Button Tag: must not contain input, select, textarea, label, button, form, fieldset, iframe or index
 - d) Label Tag: must not contain label.
 - e) Form Tag: must not contain form

2. The root element must be HTML
3. The root element must contain an XML ns declaration of XML namespace which is defined to be `HTTP://WWW.W3.ORG/1999/xhtml`

* `1999/xhtml`

→ `<html xmlns="https://www.w3.org/1999/xhtml" xml:lang="en" lang="en">`

4. There must be DOCTYPE declaration in the document prior to root element. The public identifier included in DOCTYPE declaration must refer to 1 of 3 DTDs.

The public identifier must not be used to override any parameter.

NOTE → The XML NS attribute is required in XHTML, invalid in HTML 4.01 & optional in HTML 5. The HTML validator at [HTTP://w3.org](http://w3.org) does not complain when XML ns attribute is missing. This is because namespace is default & will be added to <HTML> Tag even if you do not include it.

DTD (DOCUMENT TYPE DEFINITION)

1. A DTD specifies syntax of a document written in some language (HTML, XHTML)
2. It specifies:
 - structural hierarchy
 - names & types of element
 - content type of element
 - attribute, names & values
- 3) XML 1.0 has 3 DTDs namely strict, transitional framest.

DOCTYPE → Should be specified in first line.

- It tells the web browser which lang. is been used for set of instructions
- We can use transitional which allows more flexibility than strict.
- Use a DTD to identify type of markup language in a webpage.

i) XHTML 1.0 TRANSITIONAL → This is least strict specification.

It allows use of CSS of traditional format.

ii) XHTML 1.0 STRICT → requires exclusive use of CSS

iii) XHTML 1.0 FRAMESET → requires pages using XHTML frames

HTML V/S XHTML

HTML	XHTML
→ It is not case sensitive.	→ It is case sensitive.
→ For internet media type is text/html.	→ It is application/XHTML+xml
→ It is less expressive	→ Comparatively more expressive
→ Application of SGML	→ Application of XML
→ has document file format	→ has markup language as type format.
→ requires lenient HTML specific parser.	→ needs to be parsed with standard XML parser.
→ There is no concern . well formed constraints	→ It is concerned with well formed constraints

XHTML DOCUMENT STRUCTURE →

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional //EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  
<html xmlns = "http://www.w3.org/1999/xhtml">  
  <head> <title> </title> </head>  
  <body> </body>  
</html>
```

META TAGS IN HTML →

- It provides meta data (data/info about any data) about HTML document.
- Meta data will not be displayed on a webpage, it is only understandable by machine (browser).
- Meta elements are used to specify page description, keywords, author of document, last modified etc
- Meta data can be used by ~~to appoint browser how to display~~ Browser (how to display content of released page)
- Search Engines (keywords)

- Meta tags are used inside `<head>` element.

→ The following are the attributes of meta tag with value of description

ATTRIBUTE	VALUE	DESCRIPTION
1) charset	character-set	It specifies character encoding for HTML document e.g. ASCII codes
2) content	text	gives the value associated with http-equiv for name attribute
3) http-equiv	content-type default-style refresh	provides an HTTP header for the info/value of content attribute
4) name	authorname description, keyword	specifies name for meta data.

"
UTF → Universal character set
transformational format of 8b

`<!DOCTYPE >`

`<html>`

`<head>`

`<meta charset = "UTF-8" >`

`<meta name = "description"`

`content = "example" >`

`<meta name = "author" -`

`content = "Rahul" >`

`<meta http-equiv = "Refresh" content = "20" >`

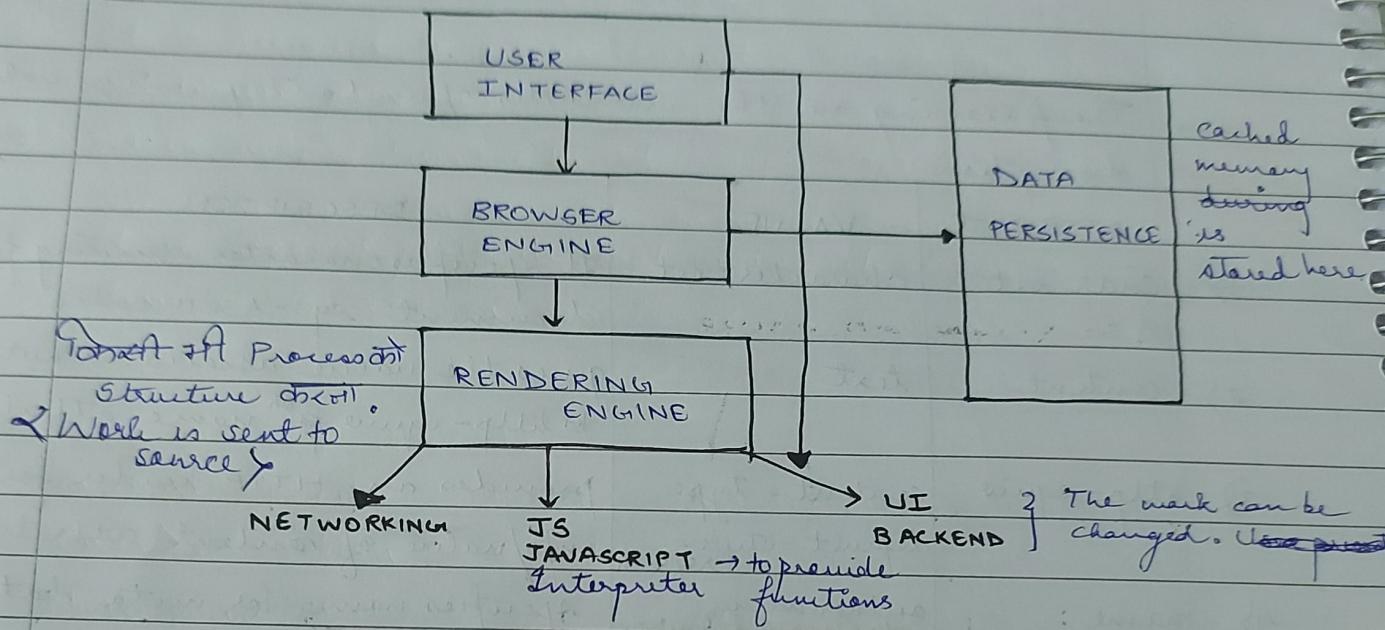
`</head>`

`<body>`

`</body>`

`</html>`

WEB BROWSER & IT'S ARCHITECTURE



WEB BROWSER is an application software that is installed on a computer to provide access to www (World Wide Web)

How browser works ?

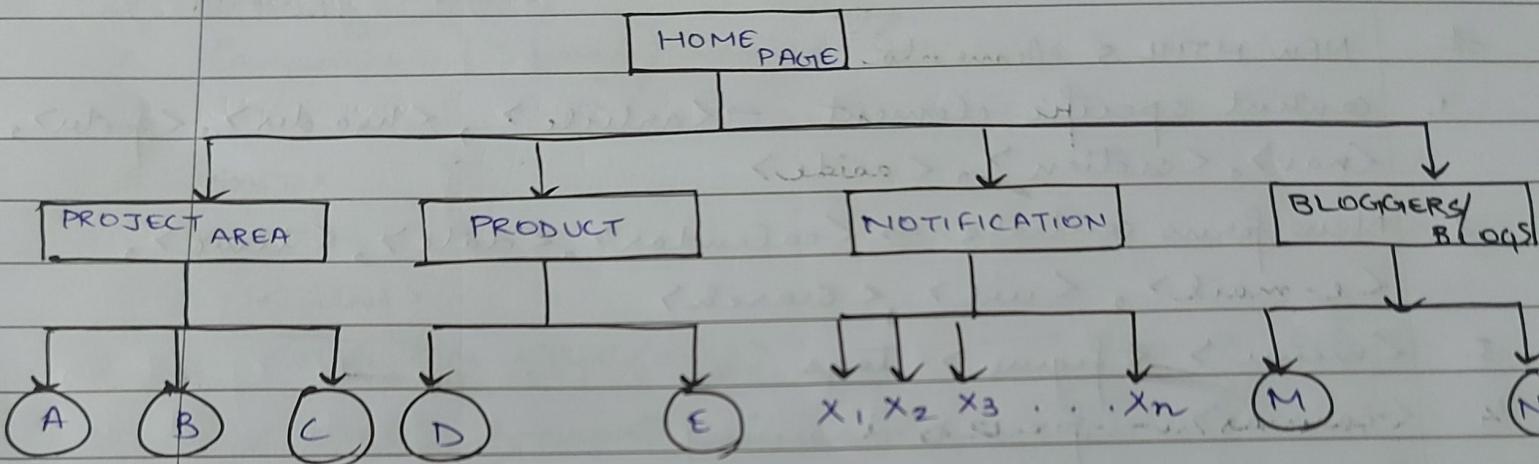
1. Contact to DNS server
2. Contact to server
3. Rendering
 - Loading HTML
 - Parsing
 - Apply Styles
 - Construct frames
 - Frames layout
 - Frame's Painting

#

WEBSITE STRUCTURE

- Website structure is imp for user to understand flow of webpages.
- The home page in a webpage needs to have links to most imp sub pages.

HIERARCHICAL STRUCTURE



#

Advantages

- Better user experience
- Proper site link
- Easier to manage pages
- Faster Search Engine Crawling

#

Parts of a Website

1. Header
2. Navigation Bar
3. Main Section
4. Side Bar
5. Footer

o) Previous version was HTML 4.01 which appeared in 1993
of this new version is still building up more... / /

HTML 5 FEATURES →

1. It is a new standard of HTML
2. It is designed to deliver content without requiring additional plug-in.
3. With help of this we build complicated application which can run on client side browser also.
4. It is cross platform that means it supports big screen devices as well as small screen devices.

NEW HTML 5 elements

1. Content specific element → <article>, <header>, <footer>, <nav>, <section>, <aside>
2. New form controls → <calendar>, <date>, <time>, <e-mail>, <url>, <search>
3. <menu> <figure> tags
4. <canvas> - for 2D drawing
5. <audio & visual video> elements

Elements removed in HTML 5 →

1. <frame>
2. <center>
3. <dir> <big>
4. → bold
5.
6. <frameset>
7. <acronym>
8. <applet>
9. <basefont>
10. <strike>

p We will use <link> tag when we use external CSS...