

"Web Designing"

* Introduction =

→ Creating, Maintaining websites & web applications.

→ websites 2 types की होती हैं-

A. Static = जो fixed content दिखाती है } without scripting & Database
जैसे- Basic HTML Pages.

B. Dynamic = ऐसी websites जो interactive होती है और user के inputs & actions
के according change होती रहती है
जैसे- Social Media & E-commerce Sites.

* Major Components of Web Development →

A. Front-End (Client Side) ⇒

- ऐसा Part जो User directly देख सकता है।
- जो Browser में Render होता है।
- HTML, CSS & JavaScript.

C. Full Stack ⇒

- Front End + Back End.

B. Back-End (Server Side) ⇒

- ऐसा Part जो User directly नहीं देख सकता है।
- Back-End & Server गुरु Manager/React
- PHP, Python, Java, Ruby, Node.js.

* Node.js =>

- JavaScript का RunTime Environment.
- इसका Main Use JavaScript को Server Side पर Run करना होता है, जबकि Traditionally पहले Client Side पर Run होती है।
- इसका Use करके यदि User front End पर JavaScript पर Work करता है, तो वह अपना Back-End भी JavaScript में लिख सकता है, जिससे learning curve easy हो जाता है।
- Node.js में Asynchronous programming होती है Means जब किसी Task को Request करते हो, तो Node.js का process उस Task को Request करता है तभी उसका Task Regular चलता रहता है।
- Means जब First Task complete होता है, तब उसका Response मिल जाता है।

* Features of Node.js =>

A. Event Driven Architecture =>

→ Node.js का architecture Event driven होता है, जो इसकी performance को fast बनाता है, जब Real Time Applications build करनी हो।

B. Single Thread =>

→ Node.js में singleThreaded event loop का use होता है, जियहे resources को efficiently manage कर सकते हैं।

C. NPM (Node Package Manager) =>

→ Node.js का एक important tool जो different modules & libraries को install करने में help करता है।

* GIT *

- Distributed Version Control System.
- Code को Track & manage करने की facility देता है।
- यदि किसी Team में कार्य करते हैं, तो Git allows करता है, कि प्रत्येक Member अपने code को & changes को एक Codebase में बिना किसी conflict के Merge कर सकें।
- इसका use करके हम हमारे code के सारे versions को easily track कर सकते हैं।
- यदि कुछ गलत कर दिया हो, तो पुराने version पर वापस जा सकते हैं।

- जब किसी Team के साथ कार्य करते हैं, तो आपके changes को अन्य Member के changes के साथ Merge कर देना है, जिससे Team Members efficiently कार्य कर सकते हैं।
- Git में इस Branches create कर सकते हैं और इन सभी Branches को Main Branch के साथ Merge कर easily Manage किया जा सकता है।

* HTML →

→ HyperText Markup Language.

→ Standard Markup language जिसका use webpages को structure देने में किया जाता है।

→ HTML Content Browser में display होता है।

→ Main Task Text, Images, Videos & links को define करता होता है ताकि Browser उसे अच्छे लो display कर सके।

★ HTML Versions ⇒

Version	Year	Inventor	Feature
1.0	1991	Tim Berners-Lee	Basic Structure of web pages.
2.0	1995	IETF (Internet Engineering Task Force)	Standard HTML, Basic forms & Tags (<u>!DOCTYPE</u>)
3.2	1997	W3C (World Wide Web Consortium)	Tables, Scripting & CSS Support Starts
4.0	1997	W3C	Multimedia, Separate CSS, Frames
4.01	1999	W3C	Minor update & fixes, Improved Version of HTML 4.0

XHTML 1.0 (XML + HTML) (Extensible HTML)	2000	W3C	XML Based version of HTML, Strict Syntax
HTML 5	2014	দিন কর্মসূলী & WHATWG (Web HyperText Applications Technology Working Group)	Stability, fully Multimedia Support, Scalability, local storage, New Elements. <u>!DOCTYPE</u>

* GIT *

- Distributed Version Control System.
- Code की Track & manage करने की facility देता है।
- यदि किसी Team में कार्य करते हैं, तो Git allows करता है, कि प्रत्येक Member अपने code को & changes को एक Codebase में बिना किसी conflict के Merge कर सकें।
- इसका use करके हम हमारे code के सारे versions को easily track कर सकते हैं।
- यदि कुछ गलत कर दिया हो, तो पुराने version पर वापस जा सकते हैं।

- जब किसी Team के साथ कार्य करते हैं, तो आपके changes को अन्य Members के changes के साथ Merge कर देता है, जिससे Team Members efficiently कार्य कर सकते हैं।
- Git में इस Branches create कर सकते हैं और इन छोटी Branches को Main Branch के साथ merge कर easily Manage किया जा सकता है।

* HTML →

- HyperText Markup Language.
- Standard Markup language जिसका use webpages को structure देने में किया जाता है।
- HTML Content Browser में display होता है।
- Main Task Text, Images, videos & links को define करता होता है ताकि Browser उसे संरचना के display कर सके।

* HTML Versions ⇒

Version	Year	Inventor	Feature
1.0	1991	Tim Berners-Lee	Basic Structure of web pages.
2.0	1995	IETF (Internet Engineering Task Force)	Standard HTML, Basic forms & Tags (<u>!DOCTYPE</u>)
3.2	1997	W3C (World Wide Web Consortium)	Tables, Scripting & CSS Support starts
4.0	1997	W3C	Multimedia, Separate CSS, Frames
4.01	1999	W3C	Minor update & fixes, Improved Version of HTML 4.0

XHTML 1.0 (XML + HTML) (Extensible HTML)	2000	W3C	XML Based Version of HTML, Strict Syntax
HTML 5	2014	ডিস কৰ্মসূলী & WHATWG (Web HyperText Applications Technology Working Group)	Stability, fully Multimedia Support, Scalability, local storage, New Elements. <u>!DOCTYPE</u>

* Basic Structure of HTML Page =

```
<!DOCTYPE html>
└<html>
  └<head>
    └<title> My Page </title> </head>
  └<body>
    <p> This is paragraph. </p>
  </body>
</html>
```

- * <!DOCTYPE> ⇒ यह Browser की बताता है, कि Page HTML5 में लिखा है।
- * <html> ⇒ HTML का root element.
⇒ सारी Coding <html> </html> के बीच लिखी जाती है।
- * <head> ⇒ यह section document के बारे में additional information / Meta information जैसे - Title, Link to scripting / CSS, Character encoding को contain करता है।
- * <title> ⇒ Page का Title define करता है, जिसे Browser की Title Bar में display किया जाता है।
- * <body> ⇒ यह section Page की Complete coding & visible content को रखता है।

* HTML Tags =

- HTML के Main Content को define करने के लिए Tags का use होता है।
- 2 प्रकार
 - 1. ऐसे Tags जिन्हें close करने की Need नहीं होती हैं (**
**)
 - 2. ऐसे Tags जिन्हें close करना ज़रूरी होता हैं (****, **<P>**, ****)
 - ↳ Container Tags.

* HTML Headings =

- 6 Types
- **<H1>**..... **<H6>**
- **<H1>** = सबसे बड़ी
- **<H6>** = सबसे छोटी

* Paragraph <P>

- Text को एक Paragraph के रूप में display करने में use.
- <P> Tag को </P> से close करता जाती होता है

* Links in HTML

- HTML में किसी hyperlink को define करने में use.
- इवलिए लिए <A> = anchor Tag का use होता है।
- जैसे - Click
 ↓
 HyperText Reference

* Image in HTML ⇒

- HTML page में images को display करने में use.
- tag के साथ image का source भी देना चाहिए है।
- जैसे-
 - ↓
Image का
पूरा Path
 - इस image के unavailable होने पर
किसी AlternateText को display
के लिए "ALT" का use होता है।

* Image in HTML ⇒

- HTML page में images को display करने में use.
- tag के साथ image का source भी देना चाहिए है।
- जैसे-
 - ↓
Image का
पूरा Path
 - इस image के unavailable होने पर
किसी AlternateText को display
के लिए "ALT" का use होता है।

* Lists ⇒

A. Unordered List (UL) ⇒

→ Bulleted List (★, ○, □)

→ tag से start कर प्रथम list item की tag में लिखते हैं

→ उदाहरण

```
<ul>
  <li> Ravi </li>
  <li> Shyam </li>
</ul>
```

 tag के साथ type attribute लगाकर pattern change किया जा सकता है

जैसे: <ul type="circle"> ○
 <ul type="square"> □

YT=6PM

B. Ordered List →

- Number वाली list (1, A, a, I)
- इस list को tag से start कर प्रत्येक list item को tag से लिखते हैं
- By default 1, 2, 3 वाली list display होती है, परन्तु tag के साथ type का use कर 123, ABC, abc, I II III भी ले सकते हैं

→ डॉसे -
 Ram
 Shyam

1. Ram
2. Shyam

* List को किसी Random Number से start करने के लिए tag के साथ start="1" का use किया जाता है

* Description List =>

→ Formally Definition List

→ Starts with container tag `<dl>`

 `<dt>` = description term

 `<dd>` = description details

→ ડોલી - `<dl>`

`<dt>` HTML `</dt>`

`<dd>` HyperText Markup language `</dd>`

`</dl>`

* Table ⇒

→ HTML में Table बनाने के लिए `<table>`, `<tr>`, `<th>`, `<td>` tags का use किया जाता है।

→ उदाहरण -

```
<table>
<tr>
<th> Name </th>
<th> Age </th> </tr>
<tr>
<td> Ram </td>
<td> 20 </td> </tr>
</table>
```

Table Row

Table Header

Table Data

Name	Age
Ram	20

* Important Attributes ⇒

A. class ⇒ किसी element को CSS द्वारा style देने में use.

B. id ⇒ Elements को uniquely identify करने के लिए use.

C. style ⇒ Directly CSS style को apply करना।

जैसे - <p style="color:blue;"> HTML </p>

D. Href ⇒ HyperText Reference

⇒ use Anchor Tag के साथ।

E. Src ⇒ Images & videos का source URL देने में use.

F. Alt ⇒ Img tag जब किसी image को display नहीं कर पाता है, तो उसकी जगह alternate Text display करने में use.

* Input Types in HTML ⇒

A. Text ⇒ Text को input में लेना।
⇒ <input type="text">

B. Password ⇒ Password को input में लेना।
⇒ <input type="password">

C. RadioButtons ⇒ Multiple data में से किसी 1 को choose करना।
⇒ <input type="radio">

D. checkbox ⇒ More than one data को choose करता |

⇒ <input type="checkbox">

E. Textarea ⇒ long text input में लेगा |

⇒ <textarea name="Message" rows="4" cols="5">
</textarea>

* Doctype Declaration ⇒

→ HTML5 के लिए <!Doctype html> का use होता है।

↓
Living Standard

इसका कोई भी next version नहीं आयेगा, अटी update होगा।

→ Older versions के लिए <!Doctype html|public> का use होता है।

* HTML V/S XML =>

HTML	XML
<ul style="list-style-type: none">1. HyperText Markup language2. Data display किसा होगा।3. Itself a markup language.4. Not case sensitive.5. Presentation language.6. Own predefined tags.7. सारे Tags को close करना ज़रूरी नहीं होता है।8. Static	<ul style="list-style-type: none">1. Extensible Markup Language2. Data store & Transport किसे होगा।3. framework to define markup language.4. Case sensitive.5. Neither presentation language nor programming language.6. Tags can be defined according to need.7. सारे Tags को close करना ज़रूरी दोता है।8. Dynamic

* HTML V/S XML =>

HTML	XML
1. HyperText Markup language	1. Extensible Markup language
2. Data display कैसा होगा।	2. Data store & Transport कैसे होगा।
3. Itself a markup language.	3. framework to define markup language.
4. Not case sensitive.	4. Case sensitive.
5. Presentation language.	5. Neither presentation language nor programming language.
6. Own predefined tags.	6. Tags can be defined according to need.
7. सारे tags को close करना ज़रूरी नहीं होता है।	7. सारे Tags को close करना ज़रूरी होता है।
8. Static	8. Dynamic

* HTTP Methods ⇒

- HyperText Transfer Protocol
- Use = web browser & web server के बीच communication में।
- HTTP Methods का Use करके Client, Server से request करना है, कि वह क्या करना चाहता है?
- 5 Methods ⇒
 - A. Get
 - B. Post
 - C. Put
 - D. Patch
 - E. Delete

A. Get →

- Server से Data Receive करता |
- महां Data को Retrieve तो करता है but उसमें changes नहीं करता है।

B. Post →

- एवर पर New Data send करता या Create करता |
- महां Data को Request Body में send करता है।
- Get Method को Cache किया जा सकता है, परन्तु Post को नहीं।
- इसका Use किसी form को submit करने व New data add करने में किया जाता है।

C. Put →

- Server पर available data को update करता |
- Request body में data send करता है।
- यदि Server पर Resource available नहीं हो , तो यह नया Data बना देता है।

D. Patch →

- Server पर partially update करता |
- Put के जैसा ही है , but only limited fields ही update करता है।
- कम Data send करता है।

E. Delete ⇒

- Server पर available data को delete करता |
- Server के सारे Resources को delete कर देता है।

* HTTP Status Code *

A. 100 → Informational →

- * 100 = Continue
- * 101 = Switching Protocols

B. 200 → Success →

- * 200 = OK
- * 201 = Created
- * 202 = Accepted
- * 204 = No Content

C. 300 → Redirection ⇒

* 301 = Moved Permanantly

* 302 = Found

* 303 = See Other

* 304 = Not Modified

* 307 = Temporarily Redirected

* 308 = Permanantly Redirected

D. 400 → Client Errors ⇒

- * 400 = Bad Request
- * 401 = Unauthorised
- * 402 = Payment Required (future use, not in common)
- * 403 = Forbidden
- * 404 = Not found
- * 408 = Request Timeout

E. 500 → Server Errors ⇒

- * 500 = Internal Server Error
 - * 502 = Bad Gateway
- | | |
|--|---------------------------------|
| |
* 503 = Service Unavailable |
|--|---------------------------------|

★ XML →

- Extensible Markup Language
- इसका use Data को structured format में store & Transport करने में होता है।
- यह Data को different systems & platforms पर Transfer करने में use किया जाता है, जो portability define करता है।
- XML Structured & detailed informations provide करता है, जबकि JSON Data को simplified & briefly provide करता है, बल्ले इसका use web Applications में अपारा करते हैं (JSON = JavaScript Object Notation)
- JSON का use Javascript भाषा किया जाता है।

★ DTD ⇒

- Document Type Definition
- Document के structure को define करने में use.
- DTD में बहुत सी तरह के XML Document में लारे important elements & attributes available होती हैं।
- 2 प्रकार
 - A. Internal DTD ⇒ XML Document में define
 - B. External DTD ⇒ अलग file में रखकर XML Document से link कर दिया जाता है।

* PHP →

- PHP HyperText Preprocessor
- Server Side Scripting language
- यह HTML के साथ Add होकर Dynamic Web Pages बनाता है
- Database interaction में use.

— PHP + MySQL

- PHP free & open source है
- PHP flexible है, because इसका code HTML, CSS, Javascript के साथ भी
बड़े में लिखा जा सकता है।
- इसका code <? PHP . . . ? > के बीच लिखा जाता है।

<?php
echo "Hello";
?>

* JQuery ⇒

- Fast, Easy & Lightweight JavaScript Library.
- इसका Use HTML document Traversing, Animations, Event handling में किया जाता है।
- इसका कार्य webpage के interactive elements को control करना होता है।
- Multiple Browser Compatible.
- less coding.
- इसका Use करके CSS Properties को भी manipulate कर सकते हैं।

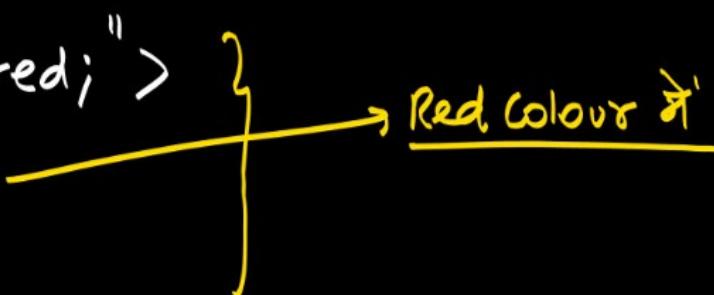
*CSS →

- Cascading Style Sheet
- इसका use किसी website को सुन्दर व खारेजक डिर्केने में किया जाता है।
- इसे HTML के साथ use किया जाता है ताकि HTML Content को बीचतर कर Browser पर display करती है।
- इसका Main use font, size, colour, border & ^{animation} property को set करने में किया जाता है।
- Developed in 1996 (by Hacon wiun Lee)
- HTML के साथ CSS तीन प्रकार द्वे use में किया जाता है-
 - A. Inline CSS
 - B. Internal Style Sheet
 - C. External Style Sheet

A. Inline CSS →

→ HTML Tags के लाए ही use but "style" Tag लगाना पड़ता है।
उद्देश्य-

```
<P style="color:red;">  
    whoshubham sir  
</P>
```



Red colour में

B. Internal Style Sheet ⇒

→ <Style> tag को create करता पड़ता है।

→ Always in <head> tag.

उदाहरण -

```
<html>
<head>
<style>
body {
background-color: blue;
}
</style>
</head>
```

TRICK

```
<body>
<h1> whooshubhamsin </h1>
</body>
</html>
```

C. External CSS

- प्रलिङ्ग से file create करनी पड़ती है।
- इसमें एक external stylesheet में CSS Code लिखकर Multiple HTML document में import किया जा सकता है।

जैसे-

```
<html>
  <head>
    <link rel="stylesheet" href="externalstyle.css">
  </head>
  <body>
    <h1>whoshubhamsir </h1>
  </body>
</html>
```

★ HTML v/s CSS ⇒

HTML	CSS
1. HyperText Markup Language	1. Cascading Style Sheet
2. Markup language दो web page का structure ready करती है	2. Style sheet दो web page को अलग-2 styles में display करती है।
3. इसमें Tags होते हैं	3. इसमें Selectors & Containers होते हैं।
4. No Types	4. 3 Types
5. HTML में CSS can be used.	5. CSS से HTML can't be used.
6. Lower Backup	6. Higher Backup
7. Extension = .htm or .html	7. Extension = .css