12/09/2023

FULL STACK WEB DEVELOPMENT

→ Interconnection of networks is Internet.

→ World Wide Web: Service which allows access to any doc on any other sys.

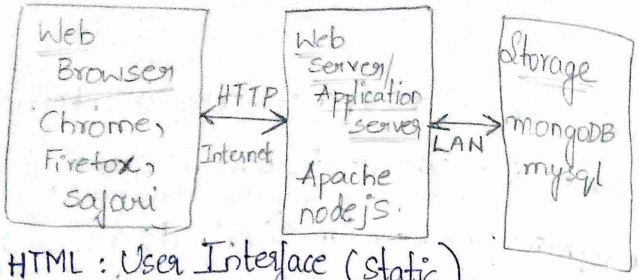
* Gopher: protocol for distributing, searching & retrieving docs over internet.
* Usenet: distributed discussion system.
* Telnet: network protocol used for remote access to other computers over a network
* FTP: File Transfer Protocol.
* NewsGroups: discussion forum on a particular topic.

→ ~~Web~~:

→ WWW: Allows users to access websites, web pages and web applications using web browsers.

protocol used:
http → hypertext transfer protocol.

→ Architecture



→ HTML : User Interface (static).

→ Scripting languages are used to make the page dynamic on client side.

Ex: JavaScript, VB Script.

→ For server side dynamism ; we write backend programs which also allows us to access database.

→ UI logic
→ Business logic
→ Data logic
} Stack

React : library for frontend.
Nodejs : backend.

13/09

15/09/2023 HTML → Hypertext Markup Language.

→ Webpages are developed using HTML

* Tables
→ cell spacing : spacing between cells.
→ cell padding : spacing in the cell.
→ Comment <!-- --> ;

* FORMS
→ meta : description and keywords about this page.
character set : which encoding set we are using.
UTF-8 : popular encoding set.
http-equiv : which browser is used.
name : viewport → user interface.
content : device-width sets acc. to device width.

→ Input types → button, checkbox, color, date, email, file etc.

→ No need of labels, if we use placeholders.

→ <input type = "radio" name = "gender" /> Male
<input type = "radio" name = "gender" /> female

name helps to group the attributes

→ checkbox ⇒ to select multiple options

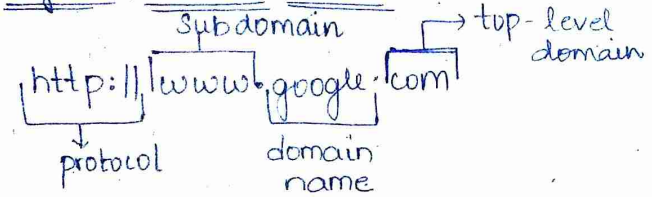
→ query string :- contents of the submitted form will be sent to servers

19/09/2023

→ Browser accepts only a certain length of data in a single line.

* Forms
  - → method : 'get', 'post'
  - → action : executes whenever we click Submit which allows us to go to next page.

* Uniform Resource Locator :



Uniform Resource Identifier (URI)

→ HTTP is a application layer protocol.

→ Communication between client computers and web servers is done by sending & receiving http requests and responses

→ HTTP : connectionless protocol & stateless.

* Connection oriented : All the packets (divided message) is sent over a single specified path.

* Connection less : All the packets does not have a specified path to reach the destination.

* For every operation on the server that uses HTTP protocol ; requests are sent by the user and responses generated gives us the O/P. After every operation ; the connection with the server is terminated.

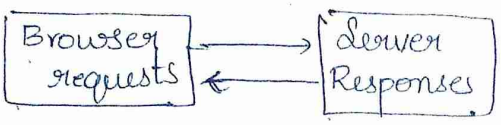
* Types of connections :
  - → Non-persistent HTTP : atmost one obj is sent over TCP connection
  - → Persistent HTTP :
    - ↓ multiple objs can be sent over a single TCP connection b/w client & server.

* HTTP Methods :

→ Get : get the resource from server.
→ Put : sending document to the server.
→ Post : Sending data to the server.
→ Delete : delete an obj on server
→ Head : request info about a web page.
→ Trace : trace path from client to server

→ option: determine server's capabilities

* put, delete methods are used to reflect the changes made to the page; on the server as well as on the other machine.



* Format of message: Message format is constructed by the browser
  - → Start line = Request / Response
  - → Header fields
  - → Message Body. (Username/IP address).

* Encoding the message and reducing the size of the message helps to avoid traffic over the network.

* http error 405: client error.

* Status Codes:
  1) 1XX → Informational
  2) 2XX → Success
  3) 3XX → Redirection
  4) 4XX → Client error
  5) 5XX → Server error

20/09/2023 Semantic tags: No action, it just gives the meaning to the data

→ <article>

→ <nav>: all the anchor tags can be enclosed in this which are displayed in HTML.

→ tag used to logically divide the HTML page.

<aside> → aligning the text to one side of the page.

<article> → set of data under a single heading

→ &copy ⟹ copyright content

→ <audio>, <header>, <svg>

→ <video>, <footer>, <meter>

→ <progress> ⟹ creates a progress bar.

→ <datalist>: used to create dropdown.

* style sheets are used to style rather than browser default.
  - ↳ Not have to code styling on every element.

* Cascading Style Sheets:

→ Types of Styling:

* * External → for a set of webpages
  * Internal → for a particular page
  * Inline → for a particular tage

* → Cascading:
  - If the same tag has different stylings, it is applied by the nearest colour only.

Selector → tag

{ 1 2 3 } → property : value pairs

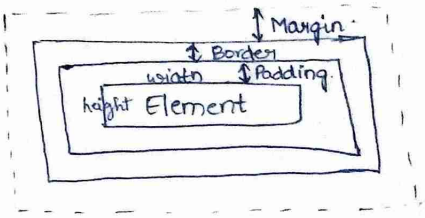
→ Ex:

```
body {
    font-family : Tahoma, Arial, sans-serif
    color : black
    background : white
    margin : 8px }
```

Ex: aside, p { color : blue }

↓ 2 tags ↓ same styling

* .a { color : yellow }
  ↓ class selector
* #idone { - - - }
  ↓ id selector
* pseudo selectors.
* CSS Box Model:



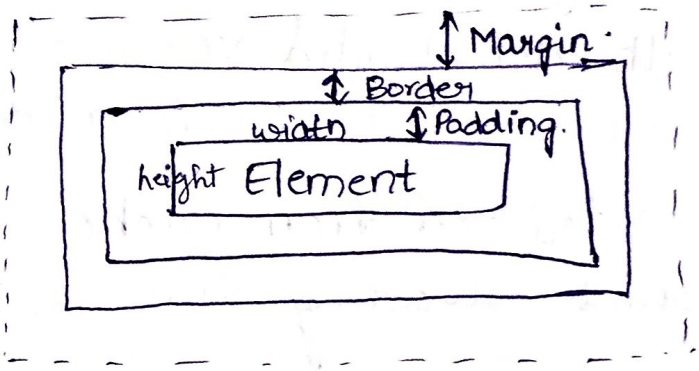
* .a { color: yellow }
  ↓
  class selector

* #idone { - - - }
  ↓
  id selector

* pseudo selectors.

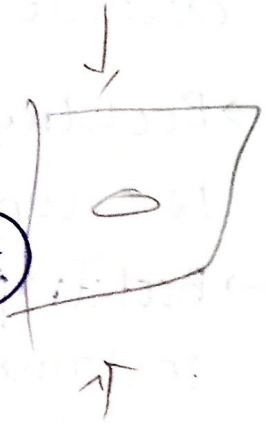
→ * {
  border-style: solid; }

* CSS Box Model:



22/09/2023

* TYPES:
→ User Agent Style Sheet (default)
→ External StyleSheet
→ Internal
→ Inline



* Types in Position property:
(default)
→ static = not effected by top, bottom, left & right properties
→ Relative = top, bottom, left & right are considered to position
  specific coordinate
→ absolute = remove normal flow & gets overlapped with oth.
→ fixed.

26/09/2023

* Responsive Web Page:

→ @media screen and (min-width: 240px) and (max-width: 480px) { p{color: red;} }

→ Adjusts the webpage according to the device width / form factors.

→ Bootstrap helps in developing a responsive webpage.

→ : We use the downloaded version of bootstrap; as a result while accessing it, it is retrieved from cache due to which loading time is reduced.

→ Bootstrap grid system allows up to 12 columns.

→ Bootstrap contains different types of containers - xl, xxl, md, sm, etc

→ Bootstrap also has breakpoints (6) which make the page responsive.

→ Responsive Web page uses html and css only.

→ Viewport is user's visible area of a webpage.

* Viewport varies with the device and will be smaller on a mobile phone or computer screen.

↳ This can be handled using <meta> tag.

<meta name="viewport" content="width = device-width, initial-scale = 1.0">

- initial-scale = 1.0 → Sets the initial zoom value.
- width = device-width → Sets width of the page to follow screen width.

* We can use CSS media queries to apply different styling for small and large screens.

* Webpages are designed based on grid-view which is useful in placing elements in the page.

* Responsive grid-view often has 12 columns and has a total width of 100% and resize acc. to device width.

→ @media → includes a block of CSS properties only if a certain condition is true.

→ We add a breakpoint where certain parts of the design will behave differently on each side of the breakpoint.

30/09/2023

* Javascript: → lightweight programming language.
→ Programming and Scripting language.
→ Object Oriented language
→ Dynamically typed programming language that is the code is interpreted at runtime.
* does not have explicit datatypes

→ Benefits:
- → Usability
- → Efficiency
- → Event driven → responding to clicks.

→ Javascript: * var i=0; (let can also be used)
* console.log(i); → printing statement and prints in console.

* Primitive datatypes:
- → Strings
- → Numbers
- → Boolean.

Composite datatype:
- → Objects
- → Arrays.

→ <script></script> : can be written anywhere in HTML code and any no. of times we can use.

→ typeof( ) returns the datatype.

→ const b=20; → declaring a constant.

→ funtion uppercase(){
console.log(a.touppercase()); }

→ function multiply(c,d){
return c*d; }

Console.log(multiply(20,10)); // 200

const m = multiply(5,6);

Console.log(m) // 30

→ Arrays :— let a = [10,20,30];
function printarray(){
for (i=0; i<a.length;i++)
{console.log(a[i]);

printarray();

Another way of Array:

```
let b = new Array('hello',20);
console.log(b);
```

→ By declaring the variable with let or const these are declared locally.

→ otherwise they are global.

→ if (date.getHours() <=12) { } → pop-up box will be ~~printed~~ shown.
alert("Good Morning")

Q) Write a program to design a calculator using javascript for +, -, *, /.


```
<html>
<head>
<title> Calci </title>
<script> let n1 = document.get
ElementbyId ("Number1");
let n2 = document.getElementbyId ("Number2");
let operator = document.getElementbyId
("Operator");
const res;
if (operator == '+')
{ res = Number(n1.value) + Number(n2.value)
}
else if (operator == '-')
{ res = Number(n1.value) - Number(n2.value)}
else if (operator == '*')
{res = Number(n1.value) * Number(n2.value)}
else
{res = Number(n1.value) / Number(n2.value)}
~~Console.to~~
</script>
<body>
<form method="get", action=" ">
<input type = "number" id = "Number1" />
<input type = "number" id = "Number2" />
<input type = "text" id = "Operator" />
<input type = "button" value = "Calci" />
</form />
```

Results:—