

## 1. INTRODUCTION TO WWW

WEB 1.0 / WEB 2.0 / WEB 3.0

URI / URL / URN

DNS CLIENT / SERVER MODEL

TCP / IP

HTTP / HTTPS

SMTP - POP3 - MIME - IMAP

WEBSITE VS PORTAL VS BLOG VS FORUM

HTTP REQUEST / RESPONSE HEADER

WWW - invented by TIM BERNERS LEE

- 1990 (WWW → BROWSER)

- 1991-94 (CERN)

- 1994 (W3C)

INTERNET - GLOBAL system of interconnected computer networks

WWW - collection of interconnected websites

- one of services that run on Internet

### WEB 1.0

web

Read only  
info sharing  
connect info.

static

Eg. HOTMAIL

### WEB 2.0

Social Web

write & Read  
interaction

connect people

dynamic

Eg. YOUTUBE

### WEB 3.0

Semantic Web

W-R-E execute

immersion

connect, knowledge

curiously undefined

Eg. iGoogle

## URL (UNIFORM RESOURCE LOCATOR)

http://www.google.com → domain

site name

separator

communication protocol

## URI (UNIFORM RESOURCE IDENTIFIER)

- ↳ a way to identify contents
- ↳ most common form of URI is <sup>web-</sup> page address  
i.e. URL C URI

## UNIFORM RESOURCE NAME

- ↳ to remove limitation URN was introduced
- ↳ no need to remember whole path
- ↳ only resource name is required

Eg. urn: def:// blue-laser → resource name  
↳ directory

## DOMAIN NAME SYSTEM (DNS)

→ It provides IP address from domain name.

PARTS → DOMAIN NAME

→ DOMAIN NAME SPACE

→ NAME SERVER

→ Root

→ Primary

→ Secondary

CLIENT SERVER MODEL - It describes how server provides resources and services to one or more clients

CLIENT → SERVER

↑ request

provides, stores info

## TCP (TRANSMISSION CONTROL PROTOCOL)

- ↳ breakdowns data into small packets & assembling

## IP (INTERNET PROTOCOL)

- ↳ takes care of communication between computers
- ↳ responsible for addressing, sending & receiving the packets

## TCP/IP FOR WEB

HTTP - Hyper text transfer protocol  
(80) - used for sending request & returning content

HTTPS - HTTP secured

(443) - handles sensitive info. like details of CARDS

FTP - FILE TRANSFER PROTOCOL

(21) - handles transmission of data

## TCP/IP FOR EMAIL

SMTP - Simple mail transfer protocol

(25) - push protocol  
- transmit only pure text

POP3 - Post office protocol

(110) - pull protocol  
- used by email for receiving emails  
- emails are downloaded when connects to mail server

MIME - Multipurpose Internet Mail extension

- If it exists than email can transmit voice, image, video & binary data.

IMAP - Internet message access protocol

(143) - stores messages online  
- It does not download emails unlike POP3

## HTTP HEADER REQUEST

<request lines> : method, url, version  
<general headers> : <sup>user</sup>server / host name  
<request headers> (connection)  
<entity headers> : (language)  
<empty line> ---  
<message body> optional

## METHODS

GET - used to get info from URL

POST - used to send data over network

PUT - replace current target resource with uploaded content

DELETE - removes all current target resource

CONNECT - establish connect with URL ↑

OPTIONS - describe commu. options for TR.

## HTTP RESPONSE FORMAT

<status line> : version, code, phrase  
<general header> : (data, server)  
<response header> : (connection)  
<entity header> : (last modified, content length)  
<empty line> : ---  
<message body> :

## CODE - PHRASES

300

301 moved permanently

400

bad request

404

not found

505

HTTP version not supported

## 2. HTML 4.0

HTML - HYPERTEXT MARKUP LANGUAGE

(-)- consists of elements

- describe structure

<!DOCTYPE> It says to browser, which version of HTML page is written.

<frame> - not supported in HTML 5

- defines particular window in frameset

- border, scrolling, resizable

- empty tag

Eg. <frame src="html.html">

<frameset> - HTML 5 X

- can hold more than 1 frame

- specifies rows, cols, % or pixel of space each frame will occupy.

FRAME - divides window into multiple parts which can load HTML document

iframe is able to float with content in a page,

we can show another html position ~~no~~ with iframe.

frameset splits window into parts

↳ cannot use body tag

can use body tag

### 3. HTML 5

- ↳ more accessible for user & machine
- ↳ semantic elements (header, footer, article)
- ↳ <audio><video>
- ↳ dynamic web page

TAGS IN HTML 5 - <header><footer><article><nav>  
 <section><aside><main><canvas>  
 <description><summary>

HTML 5 FORM - REQUIRED - INPUT TYPES -

- PLACE HOLDER
- PATTERN
- DISABLED
- READONLY
- MULTIPLE
- AUTOCOMPLETE

SEARCH

EMAIL

URL

TEL | NUMBER

RANGE / MIN-MAX

COLOR

DATE | MONTH | WEEK | TIME

<!DOCTYPE html>

<html> lang = "en">

<meta charset = "utf-8">

FOR AUTO REFRESHING

<meta http-equiv = "refresh" content = "30" />

AUDIO

<audio controls>  
 <source src = "horse.mp3" type = "audio" />

VIDEO

<video width = "500px" controls>  
 <source src = " .mp4" type = "video/mp4" />

## CANVAS

```
<canvas id="c1"  
height="300px" width="100px"  
border>
```

```
<script>
```

```
var can = document.getElementById("#c1");  
var context = can.getContext("2d");  
</script>
```

## LINE

- moveTo(0,0);
- lineTo(100,100);
- fillRect(left, top, width, height);
- strokeRect();

Eg.

```
pattern = context.createPattern(img, "repeat");
```

This will fill images instead of black colour

```
var img = new Image();  
img.src = "xyz.png";  
img.onload = function();
```

• createLinearGradient(x0,y0,x1,y1);

└ starting points

• createRadialGradient(x0,y0,z0,x1,y1,z1)

• arc(x,y, radius, startAngle, endAngle, anticlockwise)

```
context.beginPath();
```

```
context.arc(50,50,30,0,Math.PI*2; true);
```

```
context.closePath();
```

## IMPORTING IMAGE TO CANVAS

```
var image = document.getElementById("image id");
context.drawImage(image, 0, 0);
```

top  
bottom

## DRAWING TEXT

```
context.fillText('text', x, y)
```

FONT - value

DIRECTION - ltr, rtl

TEXT ALIGN - left, right, center

TEXT BASELINE - top, hanging, middle, alphabetic,  
bottom

## SVG (SCALABLE VECTOR GRAPHICS)

```
<svg width="100" height="100" > </svg>
```

```
<circle cx="50" cy="50" r="40" stroke="green"
stroke-width="2" fill="yellow" > </circle>
```

```
<rect x="10" y="10" width="50" height="50" stroke="black"
+ [rx8 ry] - rounded rectangle > </rect>
```

stroke: black

stroke: gray

stroke: gray

stroke: black, stroke-width: 2px

(filled shape)

(stroke, stroke-width: 2px, fill:none)

(dashed line)

## 4. CSS 2.0

- ✓ INTRODUCTION
- ✓ TYPES OF CSS
- ✓ CSS, ID & CLASS
- ✓ CSS BOX MODEL
- ✓ TEXT
- ✓ FONT
- ✓ LINK
- ✓ LISTS
- ✓ TABLES
- ✓ POSITIONING
- ✓ DISPLAY
- ✓ DIMENSION
- ✓ FLOATING
- ✓ PSEUDO CLASS
- ✓ PSEUDO ELEMENT
- ✓ IMAGE OPACITY

### NEED OF CSS

- to provide structure
- easy maintenance
- reduce multiple lines
- give styling & colors
- formatting & manipulation

### TYPES OF CSS

- EXTERNAL
- INTERNAL
- INLINE

<link rel="stylesheet" type="text/css" href="-css"/>

## 5. CSS 3.0

- ✓ SELECTORS
- ✓ PSEUDO-CLASSES
- ✓ PSEUDO-ELEMENTS
- ✓ BORDERS
- ✓ BOX EFFECTS
- ✓ BACKGND IMAGES
- ✓ TEXT & TYPOGRAPHY

- ✓ WEB FONTS
- ✓ 2D & 3D TRANSFORMATIONS
- ✓ TRANSITION
- ✓ ANIMATION
- ✓ USER INTERFACE
- ✓ FLEX BOX
- ✓ GRID

: active : first-child

: checked : first-of-type

: disabled : focus

: empty : hover

: enabled : invalid

: in-range : last-child

: link : last-of-type

: not (selection) : nth-child()

: optional : nth-last-child()

: required : nth-last-type()

: out-of-range : nth-of-type

: read-only : only-child

: write-only : valid

: root : visited

:: after :: before

:: first-line :: first-letter

DROP SHADOW

hori

blur-distance.

|

|

box-shadow : 2px 5px 0 0 green;

verti

spread

TEXT SHADOW

top

text-shadow : 2px 5px 1px green;

left

blur

Linear-gradient ( 30 degree, col, col )  
radial-gradient ( center, col, col )

## TRANSFORM

transform: translate (45px, 45px);  
                  LEFT    TOP

- : translate X
- : translate Y

transform: scale (2);

- : scale X
- : scale Y

transform: rotate ( $\pm 360$  deg);

- : rotate X
- : rotate Y  
      z

transform: skew ( 20deg, 40deg );

- : skew X
- : skew Y

transform & transition: 2s ease / in/out / linear

## ANIMATE

animation-name: example

@keyframe example

animation-duration: 4s

{ from { } to { }

- delay : 2s

} 0% left top back

- iteration-count: infinite

- direction : reverse / normal / alternate - reverse

name	timing	iteration
------	--------	-----------

animate: example 5s linear 2s infinite alternate;

duration	delay	direction
----------	-------	-----------

## CHAPTER 9: JAVASCRIPT

QUESTION

ANSWER

- ✓ what is variable? - variable is nothing but a monitor which work as a storage work of information or more of it variable - no name or ID will be given to variable
- ✓ what is string? - string is a sequence of characters

variable means predeclared function -  $f(x) = x^2$

what is return? - return statement returns -  $y = 2x + 3$   
returning back value outside -

what is local variable -  $x = 5$

global variable -  $x = 5$

using global variable -

what is scope & lifetime -  $T_1 \cap T_2 \cap T_3 \cap T_4$

global scope -  $T_1 \cap T_2 \cap T_3 \cap T_4$

local scope -  $T_1$

function scope -  $T_2 \cap T_3 \cap T_4$

block scope -  $T_3 \cap T_4$

variable declared in function

variable declared inside function -  $T_3$

variable declared in block -  $T_4$

variable declared in function -  $T_3$

variable declared in block -  $T_4$

variable declared in function -  $T_3$

## 6. JAVASCRIPT

- ✓ INTRODUCTION
- ✓ DATA TYPES & VARIABLES
- ✓ TYPE CONVERSION
- ✓ OPERATORS
- ✓ DECISIONS
- ✓ LOOPS
- ✓ FUNCTIONS
- ✓ ARRAYS
- ✓ COMMON MISTAKES

VALIDATION FORM

USER INTERACTION

STRING MANIPULATION

COOKIES

DEBUGGING & ERROR HANDLING

- JAVASCRIPT
- only validates programs at server side
  - control statement
  - validation
  - in-built functions
  - case sensitive
  - event handling

HOW TO PRINT IN BROWSER - `document.write`

- `console.log`

- `alert(" ")`

= assigns

== compares value

== = compares value & datatype

$2 < 12$  TRUE

$2 < "12"$  TRUE

$"2" < "12"$  FALSE

$"2" > "12"$  TRUE

JAVA

JAVASCRIPT

- strongly typed
- programming lan.
- .java
- runs on any JVM or browser
- more memory
- loosely typed
- scripting lang.
- .js
- runs only on browser
- less memory

JAVASCRIPT ARRAYS

→ `.length`

→ `.push()`; on `fruits[fruits.length] = _;`

→ `.sort()`

alphabetically

→ `.pop()`

→ `toString()` used with array & objects

`join()` only used with arrays

→ `.reverse();`

→ `.shift()` - removes first element

→ `.unshift()` - adds from first element

and shift other elements

→ `.slice(1,3);`

returns element

182

→ `splice(2,0," ", );` → elements to add

↳ elements to delete

↳ position of element

→ `.concat( );`

## JAVASCRIPT COMMON MISTAKES

- ↳ misplacing semicolon
- ↳ accidentally using `(=)` while instead of `(==)` in conditions
- ↳ Expecting loose comparison (using `=` instead of `==`)
- ↳ confusion between addition & concatenation (`+`)
- ↳ breaking JS string (\ for new line)

## ERROR HANDLING

- SYNTAX ERROR (parsing error)
- RUNTIME ERROR (calling undefined function)
- LOGICAL ERROR

```
TRY - CATCH - FINALLY
try { }
catch (e) {
    alert("Error - " + e.description);
}
finally { }
```

window.onerror = function (msg, url, line)

COOKIES - piece of data

- sent from 1 pc to another

- EXPIRE, DOMAIN, PATH, SECURE, Name = Value

set & retrieved as key-value ↪

## 7. DOM (DOCUMENT OBJECT MODEL)

- ✓ DOM TREE
- ✓ CHILD NODES (ASD & DESC) DOM
- ✓ FIND ELEMENT BY ID DYNAMIC SCRIPTS
- ✓ TAG NAME & CLASS DYNAMIC STYLES
- ✓ CREATE ELEMENT MANIPULATING TABLES
- ✓ DELETE & COPY CONTENT

DOM - tree of nodes corresponding to HTML elements  
- each node represents each element

document object represents WEB PAGE

### FIND HTML ELEMENTS

document.querySelectorAll("p.main") → 1<sup>st</sup>  
· querySelectorAll("p.main") → all  
· getElementById()

### CHANGE HTML ELEMENTS

element.innerHTML = (new HTML);

element.setAttribute(" " );

↳ For eg. href, src

element.style.property = " ";

↳ font-size, font

CSS

JS

margin-top marginTop

font-size fontSize

par = document.createElement("p");

var child = document.createTextNode("Hello");

par.appendChild(child);

element.appendChild(par);

↳ existing element

element.insertBefore(par, child);

↳ parent

↳ new

element.removeChild(par);

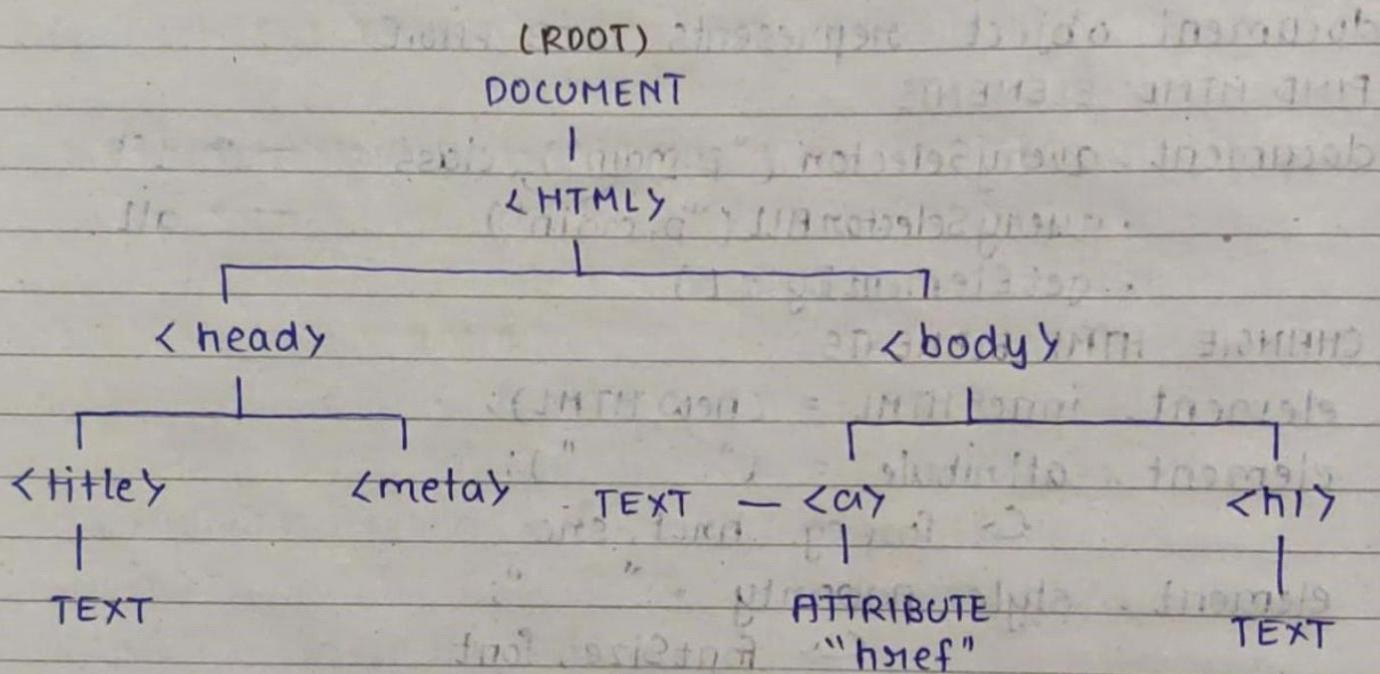
element.replaceChild (parent, child);

## JAVASCRIPT DOM EVENTS

```
element.addEventListener ('click', function());  
element.addEventListener ('hover', function());
```

```
element.removeEventListener ('click', function());
```

## HTML DOM TREE



→ every node have 1 parent except root

NAVIGATION B/w NODES - parentNode - childNode  
- firstChild - lastChild

van title = document.getElementById ("#id").innerHTML;

van title = document.getElementById ("").firstChild.nodeValue;

## 8. JAVASCRIPT FRAMEWORK

JQUERY

SELECTORS

ATTRIBUTES

EVENTS

EFFECTS

UI

ANGULARJS

NEED

STRUCTURE

MODULES

DIRECTIVES

CONTROLLER

DATA-BINDINGS

FILTER

JQUERY

↳ DISPLAY hide | show | toggle TABLE

↳ FADING fadeIN | fadeOUT EVENTS

↳ SLIDING slideIN | slideOUT

↳ ANIMATION animate

`$(selector).animate ( properties, duration, callback );`

`$(selector).stop();`

EVENTS - MOUSE

- KEYBOARD | SHIFT | CTRL | VARIOUS

- FORM

- WINDOWS → `$("#img").load (function() {});`

→ `$(window).resize (function() {});`

`$("span").text (x += 1);`

});

5

ANGULAR JAVASCRIPT - It is javascript framework

(client side)

- It extends HTML attributes with  
directives & binds with expressions

ADVANTAGES - use to build dynamic web apps

- data binding capacity

- more functionality with short code

- used to single page app. in clean way

- gives rich & responsive experience

DISADVANTAGES - Not secure

- Not degradable (i.e. if JS is disabled  
then angular.js)

## PROGRAMMING TUTORIAL

CORE FEATURE - MVC (MODEL VIEW CONTROLLER)

→ Pattern for dividing application into 3 parts

2-way  
data  
binding  
Template  
View  
Model

- MODEL (data to be merged with template)
- VIEW (what users see, template + model)
- CONTROLLER (code behind the view)

DIRECTIVES - extends HTML

- mostly starts with ng
- Eg ng-app, ng-model

MODULES - separates logic such as services, controllers etc. to maintain clean code

SCOPE - detect changes in model section

- connects controller with views

FILTERS - perform data transformation

CURRENCY | DATE | TIME | ORDER BY | UPPER-LOWER CASE

({{ctrl.\$index}}) {{ctrl.name}}  
{{ctrl.\$index}} {{ctrl.name}}  
({{ctrl.\$index}}) {{ctrl.name}}

blanks made  
so that & can  
be highlighted

AngularJS provides a built-in service called \$filter which provides filters for data transformation.

AngularJS provides built-in filters for transforming data.

Now look at the angular filter.

AngularJS provides a built-in filter.

AngularJS provides built-in filters for transforming data.

## 9. BASICS OF PHP: HYPERTEXT PREPROCESSOR

PHP - server side language

- dynamic & interactive webpage

- scripting language

<? php ... ?> → body

VARIABLES \$ variable\_name;

(automatically associates datatype)

\$ name = "Rudra"; → concats string  
echo "My Name is". \$name;

VARIABLE SCOPE → LOCAL

- GLOBAL

↓  
- STATIC

→ php stores all variables in array \$GLOBALS[i]

Eg \$x;

\$GLOBALS['x']

STRINGS - echo strlen (" ");

str\_word\_count (" ");

strrev (" ");

strpos ("Hello World", "World");

output = 6 ←

str\_replace (old, new, string);

ARRAYS \$ cars = array ("BMW", "HONDA");

\$ cars[0] ....

→ count (\$cars);

## CONNECTING & INSERTING IN MYSQL DB

```
<? PHP
```

```
$servername = "localhost";  
$username = "localhost"; "root";  
$password = "pwd";  
$db_name = "employee";  
  
$conn = mysqli_connect($servername,  
                      $username, $password, $db_name);
```

If above was connecting part

If now inserting data into table

```
$sql = "INSERT into emp (column1, column2)  
VALUES (101, "RUDRA");
```

```
$sql_result = mysqli_query($sql);
```

```
?>
```

## # BASIC QUESTIONS

### HTML 4

- NO <audio><video>
- not possible to draw shapes
- not allow JS to run on browsers
- NOT mobile friendly

### HTML 5

- <audio><video> tags ✓
- can draw circle, rectangle,
- allows JS to run in browser
- It is mobile friendly

CSS 1 - color, font, background, margin, border

CSS 2 - support concepts of media type  
- absolute, relative, fixed positions

CSS 3 - improvised visual styles  
- rounded corners - borders  
- flexbox & grid

JAVASCRIPT - client & server side  
- makes web page interactive  
- adds behaviour

Disadvantage - cannot be used for interacting networking applications  
- does not support multithreading  
- lack of debugging facility  
- reading & writing not allowed at client-side  
(security purpose)

JQUERY - javascript library  
- simple & easy

disadvantage - simplifies event handling

Disadvantage - large to download/ import  
- hides part of JS (makes hard to learn)  
- cannot use database (need PHP for that)

21-10-21-30

29-09-

21-09-2021

21-09-2021

21-09-2021

21-09-2021