Day 1

15-06-2021

https://[www.google.com](http://www.google.com) URL

http : protocol , hyper text transfer protocol

s : secure

www: world wide web

google : domain

com : commercial

req(http/https---🡪

Client Server

🡨--Res(http/https) HTML/HTML5

CSS/CSS3

HTML : Hyper text mark up language. JS(JavaScript)

HTML : it is use to display the content on browser

CSS : apply good look and feel or presentation logic on contents.

JS : Action on contents or events on contents or programming on web page.

VS Code

Html contains lot of pre-defined tags

Syntax

<tagName>

</tagName>

<tagname/> self closing tag

1, 2 , 3 HTML4

<!doctype html public =”url…..dtd”>

Document type definition

This file contains rules or structure of html page means that is root tag name, how many child tag it must contains, number of tags inside body etc.

HTML5

<!doctype HTML> optional

Heading tags

H1 to h6 tags

Attribute : attribute is known as properties of tags.

Every tag contains one or more than one attribute. Attribute in the form of key-value pairs we have to use attribute in opening tags.

<tagName key1=”value1” key2=’value2’ key3=value3>

</tagname>

Hyperlink tags

Using hyperlink we can connect more than one web page.

1. External hyperlink
2. Internal hyperlink or bookmark

External hyperlink

<a href=”tagetPageName.html”>Text</a>

Internal Hyperlink or bookmark

One Page application (HTML).

Images :

<img src=”nameoftheimage.format”/>

List tags

Ul

Ol

Table tags

Employee Details

**Id Name Salary**

100 Raj 12000

101 Seeta 14000

Forms tags

Before HTML 5

<input type=”text/password/radio/checkbox/button/reset/file/submit”/>

After HTML 5

<input type=”email/url/search/date/date-time/time/color”/>

By default every html page form method consider as a GET method.

If method is GET the information send through URL using URL rewriting technique.

URL?key1=value&key=value&key=value….

<file:///C:/Users/91990/Desktop/MEAN%20Stack/MEAN%20Stack/Programs/HTML%20and%20HTML5/home.html?uname=Ravi&pname=123>.

If method is GET. Data is not secure.

If we want to data secure the new have to use method as POST. If method is post the data will send through body part of request.

Body method after click on submit button

<file:///C:/Users/91990/Desktop/MEAN%20Stack/MEAN%20Stack/Programs/HTML%20and%20HTML5/home.html>

Performance wise get is faster than post. Using get we can send only 255 character data.

REST API

Get

Post

Put

Delete

But html or html5 support only get or post.

CSS : Cascading style sheet.

CSS provide lot of pre-defined attribute which help to apply good look and feel (presentation logic) to web page.

Using CSS we can achieve separation on concern.

Actual content Formatting style

Type of CSS

3 types

1. Inline css
2. Internal css or embedded css
3. External css

Inline CSS

<tagname style=”property:value;property:value;property:value;”>

</tagname>

Internal or Embedded CSS

Syntax

This tag must be inside head tag.

<style type=”text/css”> tag

selector {property : value}

</style>

Types of selectors

1. Universal selector : \* {property : value}
2. Specific selector : tagName {property : value}
3. Multi selector : tagName, tagName { property : value}
4. Local class selector : tagName.className{} {property : value}
5. Global class selector : .className {}
6. Id selector :

Class : group of tags.

Id : using id we can make the tag is unique.

<div>

<p class=”abc” id=”p1”>First Para</p>

<p class=”xyz” id=”p2”>Second Para</p>

<p class=”abc” id=”p3”>Third Para</p>

<p class=”xyz” id=”p4”>Fourth Para</p>

</div>

Box model

In JavaScript all html tag is know as DOM (Document Object Model).

DOM elements : it may be html, head, body, p, div etc.

According to box model every dom element follow box model concept.



Day 2

16-06-2021

JavaScript was object based interpreter scripting language.

Object based vs object oriented

Interpreter Vs Compiler

Scripting vs programming

OOPs : class, object, EPIA

JavaScript using ES5. It provide lot of pre-defined object as well as we can create user-defined objects.

We can write JavaScript code (internal or external JS file).

Syntax of script tag

<script type=”text/JavaScript”> opening tag

</script> closing tag

This script tag we have to write in between head tag, body tag or without any tags. But extension of file must be .html

Variable : variable is name which hold some value.

Syntax to declare the variable

var variableName;

data type : java script support loosely type data type.

Operator :

Arithmetic Operator : +, - , \*, /, %

Conditional operator : >, >=, <, <=, ==, ===, !=

Logical operator : &&, ||, !

Assignment operator : =

Increment and decmrement : ++, --

Ternary operator : condition ? true: false;

Type of operator : typeof

If statement

1. Simple if

if(condition) {

}

1. If else

if(condition) {

}else {

}

1. Nested if

if(condition) {

if(condition) {

}else {

}

}else {

}

1. If else if or ladder if

if(condition) {

}else if(condition) {

}else if(condition) {

}else {

}

switch:

syntax

switch(variableName) {

case label: block1;

break;

case label: block2;

break;

case label: block3;

break;

case label: block4;

break;

default : wrong block;

break;

}

Looping :

While loop

var i=0,n=10;

while(i<=n) { entry loop

do task

i++ or n--

}

Do while loop

var i=,n=10;

do {

do task

i++ or n—

}while(i<=n); exit loop

For loop

Syntax for loop

1 2 4

for(initiliazation;condition;increment/decrement) {

3

}

for(var i=0;i<=10;i++) {

document.write(“i ”+i+”<br/>”)

}

function : set of instruction to perform a specific task.

JavaScript function are divided into 2 types

1. Pre-defined global function
2. User-defined function

Pre-defined functions.

1. alert(“Msg”): it is use to display the alert message.
2. prompt(): this function is use to receive the value through keyboards.
3. parseInt(): it convert string to integer
4. parseFloat(); it convert string to float
5. eval(): it covert string to number.

do {

using alert display option

1: add, 2 : sub

switch()

case 1 Add display sum

case 2 Sub display sub

do you want to continue

}while()

Thank you

User-defined functions

Normal function style

function functionName(parameterList) {

function body;

}

1. function no passing parameter and no return type.
2. Passing value but no return type.
3. Function passing parameter and return value.
4. Function no passing parameter but return value.

Event : event is provide the interaction between user and component(dom elements).

Event provide bridge between html and JavaScript code.

JavaScript or DOM elements

JavaScript all event start with on prefix followed by name of the events.

onClick

onDblClick : button or any tags.

onMouseOver : image or map

onMouseOut

onKeyUp : textfield, password field.

onKeyDown

onChange : dropdown

onFocus : enter inside text field

onBlur : exit from text field.

onSubmit : validation with submit button

onLoad : body tag page loaded

onUnLoad : page refresh or close the application

etc

Listener : Listener are normal JavaScript function which help to listen the generated events.

**DOM : Document Object Model**

index.html

<html>

<head>

<title>Message</title>

</head>

<body>

<h1>Heading Message</h1>

<p>Paragraph Tag</p>

</body>

</html>

In browser memory it create DOM hierarchy

Html -🡪 root tag (Node )

Head body

Title h1 p

textNode – Message Heading

Paragraph Message

DOM API (Document Object Model) Application Programming interface.

Java, C,Python, C#, JavaScript provide pre-defined function or classes which help to read, write and update dom dynamically.

JavaScript DOM API

1st way to retrieve the value of text field.

document.formName.textfieldName.value

2nd way to retrieve the value of text field

document.getElementsByName(“textFieldName”)[index].value

We can write more than one tag have same name attribute.

3rd way to retrieve the value of text field

document.getElementById(“idName”).value

Form Validation

Using JavaScript

Using HTML5 features

Day 3

17-06-2021

object : object is any real world entity.

Properties or state have variables/fields/state

Person

Behaviour do/does functions/ methods

Bank

Animal

Car

Phone

Till ES5 there is no class keyword.

2 type of objects

1. User-defined object
2. Pre-defined or built in objects.

Using function we are creating user-defined objects.

Class : it a blue print of object or template of objects or use-defined data type which help to describe the objects.

Pre-defined objects

Java Script follow object hierarchy

object -🡪 property

behaviour

object -🡪 property

behaviour

object -🡪 property

behaviour

object

BOM Hierarchy : Browser Object Model

window : window is a top most object in BOM hierarchy

DOM Hierarchy : Document Object Model

Window object

document

alert()

prompt()

confirm()

open()

close()

setTimeout()

setInterval()

clearInterval()

asynchronous and synchronous communication

synchronous execution of statement

document.write(“1st statement”);

document.write(“2nd statement”);

document.write(“3rd statement”);

asynchronous execution of statement

asyn document.write(“1st statement”);

document.write(“2nd statement”);

document.write(“3rd statement”);

synchronous function call

fun1()

fun2()

fun3()

fun4()

asynchronous function call

fun1()

asyn fun2()

fun3()

fun4()

client server synchronous communication

asyn 1st Req

asyn 2nd Req

3rd Req

4th Req

Client Server

setTimeout

setInterval

clearInterval

These three function part of window object which help to achieve asynchronous operation on client side or view side.

CSS property using DOM CSS property

color document.getElement\*(“”).style.color

font-size document.getElement\*().style.fontSize

Basic Pre-defined objects

Date : This object is use to display date and time.

Day 4

18-06-2021

DOM Hierarchy

Index.html

<html>

<head>

</head>

<body>

<p>Welcome to <b>HTML</b> Web Page</p>

<font color =”” size =5 face=”arial”>

</font>

<div>

</div>

</body>

</html>

DOM API : Creating, Delete DOM element dynamically.

ES : ECMA Script European Computer Manufacture Association.

Programming concept

OOPs concept

ECMA is a concept.

JavaScript is one of the implementation of ES or ECMA.

ES5 Version

ES6 or ES2015

JavaScript using ES6 features.

Till ES5 to declare the variable we are using var keywords.

But from ES6 JavaScript we can declare the variable using let, var and const keywords.

var : using var keyword we can re-declare same variable once again with same vale or different type of values.

var a=10;

a=20; // Re-initialization

var a = 30; // re-declaration

int a=10;

a=20;

int a=20; //Error

using let keyword we can’t re-declare same variable once again with same value or different values.

Using var we can declare global scope.

Using let we can declare local or block scope.

const : it use to declare constant value.

Types of functions

function :

1. Normal function :
2. Expression style function :

If function is normal function declaration we can call that function before function declaration.

Expression style function we can’t call before function declaration.

Arrow function : arrow function is a replacement of anonymous function style.

Arrow function by default return the value without return keyword.

Callback function : passing the function or function body or function itself to another function as a parameter is known as callback function.

Basic or utility JavaScript objects.

Array :

Arrays is use to store more than one value of same or different types.

In JavaScript array is not a fixed memory size.

Syntax to declare the array

Literal style

let variableName = []; empty array

let variableName = [100,200,300,400,500];

memory creation style

let variableName = new Array() length 0

let variableName = new Array(10) length 10

let variableName = new Array(100,200,300,400)length 4

normal loop

for in loop

for of loop

forEach() : forEach() function takes callback function as a parameter(may be normal function, expression style function or arrow function) which help display the value one by one.

splice() : using this method we can add, remove and update elements in between of array.

Splice(indexPosition,deleteCount,updateElement…..)

Day 5

21-06-2021

sort() methods

created user-defined object using function style

function Employee() {

this.id=100;

this.name “Ravi”;

this.dis= function() {

}

}

let emp1 = new Employee();

document.write(emp1.id);

emp1.dis();

create user-defined object literal style.

Rest and spread operator : …

Rest and spread operator with array

Rest operator is use to store zero,1 or many array value while de structure of array.

…variableName

Spread operator with array

Rest and spread operator with function

JSON

JavaScript Object Notation

Req Java(req)

Client HDFC XML/JSON SBI

Res .net(Res) Asp.net

**Java**

**Spring boot**

Asp.net

Php

Python

Node JS

Web Service :

**JavaScript Object Notation (JSON)**

Json store the data in the key value formats.

Key must be unique and type must be string.

Vale may be number, Boolean, string, array, array of objects

{“key1”:valu1,”key2”:”value2”,”key3”:value3}

Json Conversion methods.

Converting to string to json

    let empInfo = JSON.stringify(emp);

// string to json

    let jsonData1 = JSON.parse(empInfo);

Promise :

Promise is a pre-defined object part of JavaScript which help to handle asynchronous events of data. Data may be string format, xml format, json format etc.

Promise resolve or rejected depending upon the conditions.

Full fill or rejected or pending.

User-defined promise

Pre-defined promise

To retrieve the data from promise object we have use then() and catch() callback functions.

If promise resolved then() call back function get called which help to read data from promise.

If promise rejected catch() call back function get called which help to read error message.

fetch() : It is a pre-defined function provided by JavaScript which help to consume REST API.

Return type of fetch() method is promise objects.

1st Task

Display fake json data in proper table format.

2nd Task

TextField -🡪 1 keyup events call function and take the value of textfield using dom API and append that value through url and display that records details in same page below textfields.

If id correct display records or display no record founds.



ES6 OOPs Concept.

class :

constructor : constructor is a type of special function which help to create the objects.

While creating the constructor we have to write the function name as constructor. Constructor get called automatically when we create the object.

Polymorphism

Compile time : JavaScript ES5 or ES6 doesn’t support

Run time : function overriding.

The function have same name and same signature is known as function overriding.

Day 6

22-06-2021

Scope objects : scope object is use to share the data between more than one js pages.

sessionStorage :

localStorage :

setAttribute(“key”,value);

getAttribute(“key”);

removeAttribute(“key”);

sessionStorage store the value till the application open. Once application close the value from sessionStorage get destroy.

localStorage store the value permanently.

async and await keywords are part of ES7 features.

It is use to handle promise data.

ES5 and ES6

fetch(“url”).then().then().then().catch()

TypeScript : TypeScript is super set of JavaScript. Typescript is one of the implementation of ES6 features.

TypeScript support data types concept.

Browser can’t understand or we can’t use typescript file (ts) file in html page directly.

So we have to convert TS to JS then include generated js file in html or view page.

Transpiler : It is type of compiler which help to convert one format of file into another format of file.

typescript : tsc : converting ts to js : Angular

babel : JSX : converting JSX to Plain JavaScript

want to convert ES6 to ES5.

jQuery : jQuery external JavaScript library which contains lot of pre-defined function which internally connected to each other to read, write and update DOM element easily.

Few are library and few are framework.

Ext JS

Backbone JS

Coffee JS

React JS

Angular

Vue JS

Node JS : Node JS is not a library or not a framework.

Node JS is a run time environment for the JavaScript program + library or Framework.

Before Node JS JavaScript use for Client Side scripting language.

Frontend backend

HTML,CSS,JavaScript Java JEE

jQuery Servlet/JSP

Spring boot

Asp.net

Php

Python

Node JS

After Node JS JavaScript we can client side as well as server side scripting language.

Node JS provide lot of pre-defined modules. Those module may be local module, external module which help to create server side programming language using JavaScript.

Before Node JS if we want to run any JavaScript code

1. We have to write script tag inside a html code
2. We have to create external JS file and include in html page.

After node we can run External JavaScript program using command prompt.

Node JS : mainly use to create the Server side programing language.

In Node JS script program we can’t use window and document objects.

To convert ts to js we require tsc

npm : node package manager. Npm command is use to download external dependencies or modules written in JavaScript.

npm is like mvn(Maven) in Java.

Syntax

npm install –g moduleName : globally

npm install moduleName : locally

install typescript external module using npm command.

npm install –g typescript

TypeScript features.

1. let, const and var
2. dataTypes : typescript support strict data types concept.

Syntax data type

let variablename:datatype;

let a:number=100;

let b:number=100.20;

let name:string =”Ravi Kumar”;

let result:boolean = true;

let num = 100; // valid

num =”Ravi”;

let id:number;

let fname:string;

let age:number;

let msg:any;

msg=100;

msg = “Ajay”;

msg = true;

1. array with data types.

ES5 or ES6 to declare array variable

let/var variableName = 100;

let/var variableName = [100,200,300,400,500]

let/var varibleName;

array can hold any type of values.

Function

1. function with number of parameter must be match
2. function with number of parameter as well as type of parameter must be match.
3. Function with specific return type or no return type or generic (any) return type.
4. Function with optional parameter. Optional parameter must be last parameter in functions if you are using only one variable as optional parameter. If we want more than one optional parameter so we have declare from right to left but no gap between two variables.
5. Function with default initialization.
6. Rest operator and spread operator with function

Oops Concept in typescript using ES6

object

class

Encapsulation : binding or wrapping data (variable) and code (function) in a single unit.

class

access specifiers

**public :** if variable or function are public we can access those variable and function through object of that class.

**private** : if variable private we can’t access those variable outside class directly as well as with the help of objects.

**Constructor short cut initialization.**

**Object : few methods**

**Inheritance :**

1. Single inheritance : one super class and one sub class
2. Multilevel inheritance : one super class and n number of sub classes one by one
3. Hierarchical inheritance : one super class and n number of sub class directly connected to super class.
4. Typescript doesn’t support multiple inheritance using classes but it can support using interface.

**interface**

interface interfaceName {

id:number;

name:string; //normal variable

dis():void; // incomplete functions

}

**Module :** module is a collection of variable, function, classes or interfaces.

Module is like a package in Java.

According to typescript .ts file itself is known as module.

Import and export keywords.

Typescript configuration **tsconfig.json**

tsconig.json hold all configuration details for you application.

To create the configuration file using command we have to run the command as

**tsc –init**

Set

Map

PName

DOB : type=”date”

Group

Blood group

Blood grouse reading

<65 below

65 to 100 normal

100 > above

****

Day 7

24-06-2021

Webpack : webpack is a static module bundle webpack teats all files and assets as modules.

Using webpack we can create dependencies graphs which describe how modules are related to each other using import (require) or export.

npm install –g webpack

npm install –g webpack-cli

after installation we have to create static bundle file.

**webpack.config.js**

**git :**

**github**

Sub Version control tools :

Version Control system : Version control system that records changes on files or folder and projects or application.

Local Version Control : RCS : Revision control system.

Centralized version control :SVN

Remote repository

Team 1 Team 2 Team 3

Distributed Version control system : GIT

Remote repository

Local rep local rep local rep

Team1 Team2 Team3

Git is type of Distribution version control which keep tracks of software version and allow many developers and team to work on given projects within a maintaining connection of common network.

This make only folder as local repository

git init

git status

git add : This command is use to add the file in staging area.

**Git** : it is software which provided set of commands which help to maintain the local repository as well as interact with remote repository.

Github : remote repository

AWS : code commit

Azure

Oracle cloud

Etc

How to create local repository from existing remote repository

**git clone url :** first time fresh copy of the existing

repository.

**git pull :** to get the updated content from remote repository.

Git is sub version control system

**Git branch :**

Git branch is like pointer which hold more than one commit or snap short of our changes.

By default branch is master or main

View all branches or default branch name

**Git branch**

To create new user-defined branch

**git branch branchName**

command to switch the branch

**git checkout branchName**

creating as well switch to user- defined branch

**git checkout –b branchName**

25-06-2021

.divTag {

color:red;

font-size:24px;

}

.fontClass {

}

.pTagClass {

}

Bootstrap : bootstrap is a open source CSS framework for web application as well as mobile application which help to create responsive web application.

Bootstrap provide lot of pre-defined css classes respective dom elements.

Div, p, table, button, form etc.

**Bootstrap classes**

Container and container-fluid

Container : it known as fixed width container. It take full with for web page.

Container-fluid : it leave left and right side space.

<div id=”footer”>

</div>

<div id=”nav”>

</div>

<div id=”aside”>

</div>

<div id=”main”>

</div>

<div id=”footer”>

</div>

Html5 sematic tags

<header>

</header>

<nav>

</nav>

<aside>

</aside>

<footer>

</footer>

**Bootstrap grid layout**

Grid layout allow us display the dom elements up to 12 column across the page.

You and divided the container/container-fluid in row and columns. By default each row consider as 12 columns.

Screen size

xs <576px

sm >=576px

md >=768px

large >=992px

xl >=1200px

**DOM API :**

jQuery is a external JavaScript library function which contains lot pre-defined function which internally connected to each to do Dom operation very easily.

Once the document loaded

$(“select dom using tagname,id selector,class selector”).doActionOnTag();

document.getElementByTagName(“\*”).length

Library as well as framework.

jQuery

Coffee js

Backbone js

Ext JS

Angular JS

Angular Framework

React JS

Vue JS

Office JS

**Node JS**

Node JS is not a library or not a framework.

Node JS run time environment for JavaScript library or Framework.

Java -🡪JRE

JavaScript 🡪 Node Js

Front end Server Side Technologies

HTML Servlet, JSP and EJB

CSS Spring Framework and

JavaScript Boot

Bootstrap Asp.net

jQuery Php

Phython web application

Node JS

After Node JS we can use JavaScript to develop to Server side technologies.

“Node’s goal is provide an easy way to build scalable network programming”.

Javascript callback and asynchronous operation

Node JS is a open source, platform independent or cross platform which help to developer server side as well networking application.

It is written use JavaScript. Application level.

It provide an event driven architecture and non blocking io operation. Node js achieve the features as optimize and scalability.

Node JS use Google JavaScript V8 engine to execute the node JS Programs.

Unique in Node

1. After node JS JavaScript we can client side as well as server side.
2. All server are thread base but node js a event base architecture.
3. Node JS running on V8 engine.
4. It is not library or not a framework.
5. Node JS is not a multi threaded.

Node JS we can’t access DOM and BOM hierarchy.

Node JS is not use to create GUI application.

Node is a platform for writing JavaScript application outside browser.

REPL terminal : Read Eval Print Loop

Global objects

console

process

**Node JS Modules**

Modules is a simple or complex functionality organized in single or multiple JavaScript files which can be reused through the Node application.

Types of Modules

Core Modules

fs

os

path

url

http

etc

var referenceName = require(“moduleName”);

load the module

let

Local module (user-defined modules)

Third party modules

**fs module** fs is a core module which is use to do the synchronous as well as asynchronous file handling operation.

readline : it is a external module which help to take the value through keyboards. readline module is type of async module.

Npm install –g moduleName globally

npm install moduleName locally

readline-sync : It is use to take the value through keyword as synchronous manner.

npm install readline-sync

URL :

url module provide the URL details in server side.

<http://localhost:9090?name=Ravi&salary=12000>

action =”home.html”

localPath:home.html?user=Ravi&pass=123

when we fill the form if method is GET it use query param concept.

url?key=value&key=value&key=value

**http** : http is a pre-defined core module which help to create server using JavaScript. Using this module we are going to create web page on server side technologies like in Java Servlet/JSP, Aps.net or php or python etc.

all non-node js server are thread base

Tomcat -------------🡪 Servlet/JSP

1,000 etc client request can handle at same time.

1st client send request to server --------🡪1 thread

All 1,000 client send the request and doing some task in that application.

Thread base server are lock or block the request.

Node JS Server are use even driven architecture.

class Booking {

int avl=1;

}

Booking b1 = new Booking();

Thread t1

Thread t2

Thread t3

