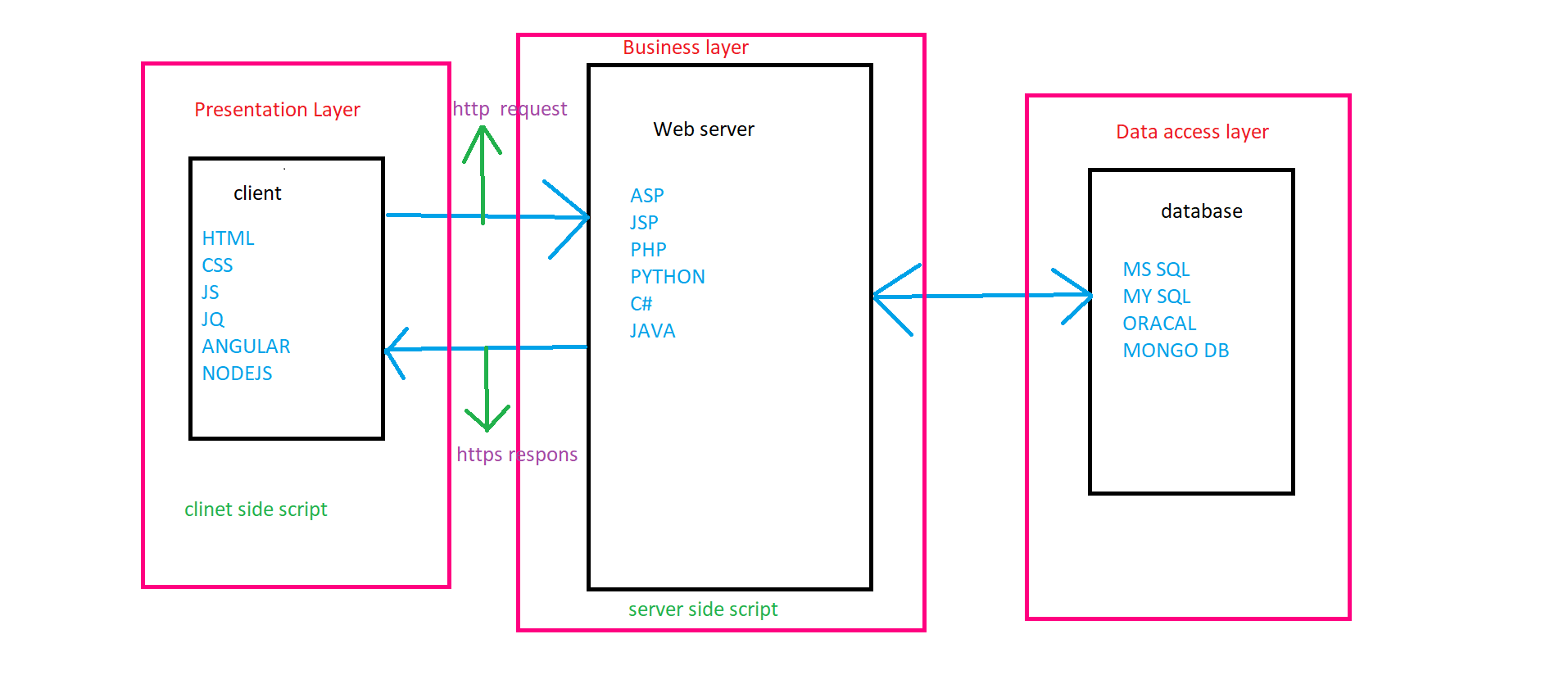
Javascript

Invented by **Brendan Eich**

create in may 1995

created by **Netscape corporation**

****

**Client side script**

This script which is running within the browser is called as client side script.

***Exercise:-***javascript ,VBscript,jquery, angular js.

**Server side script**

The script which is running within the web server is called as server side script.

***Example:-***

ASP=IIS(inertnet informstion system)

JSP=Tomcat/Glassfish

PHP=Apache

Python=Apache

**Introduction of javascript**

1. Javascript is the scripting language of HTML and the web.
2. Javascript is a client side scripting language.
3. Javascript is a dynamic ,untype language.
4. Javascript is interpreted scripting language.
5. Javascript is a case sensitive language.

**Features of javascript**

1. Js provide client side validation
2. Js is simple
3. DOM traversing
4. DOM manipulation
5. Event handling
6. Fast execution
7. Cross browser support

**Window:-** window is a primary object of javascript providing some property and method.

1. Alert()----window.alert()
2. Confirm()---window.confirm()
3. Prompt()---
4. Open()
5. Location()
6. Setinterval()
7. Clearinterval()
8. Settimeout()

**Prompt:-** prompt function is used to take input on runtime .

Return type of prompt is string.

Prompt function is located in window object.

***Syntax:- var a=prompt(“field name”,”placeholder”)***

**VAR:-**var variables are globally scoped.

Var variable can be updated and re-declared.

Example:- var a=10

a=30

**CONST:-**const variable neither be updated nor re-declared.

Example:- const a=20

const a=30

**LET:-**let variable can be updated but not re-declared.

Example:-let a=20

a=30

let a=40

**== vs === :-**

== equality operator

Here value is important but not type.

=== strict equality operator

Here both value and type is important.

10==”10”true

10===”10”false

**Type Casting || Type Conversion:-** the process of converting one type value to another type is know as type casting or type conversion.

1. parseInt()
2. parseFloat()

**Locatin:-**location property is used to redirect location one page to another page.

***Syntax;***

***Window .location =”https://www.techpile.in”***

**Confirm:-**confirm box is used to take confiramation with user .it has two .(ok and cancel)

If user click on ok button the it return true and if user click on cancel button it return false.

***Syntax:***

***Window.confirm(“massage”);***

Example:

var a=confirm("are you sure want to open techpile website")

if (a==true)

{

location="https://www.techpile.in";

}

***Open:-***open function is used to open a website on new tab or new window.

***Syntax;***

***Window.open(“url”,”\_blank”,”height=value,width=value”)***

Example:

var a=confirm("are you sure want to open techpile website")

if (a==true)

{

open('https://www.techpile.in','\_blank','height=400px,width=400px')

}

**GetElementById():-** getelementbyid function is used to select htmlelement on the basis of id.

Return type of getelementbyid function is current object.

Getelementbyid function is located in document object.

Document object is the buil-in-object of DOM providing some property and method to handle html tag.

***Syntax;***

***Document.getElementById(“Id of HTml Elenment”)***

Example:

<div id="dv"></div>

<script>

var a=document.getElementById('dv')

a.style.background='blue'

a.style.height='500px'

a.style.width='500px'

a.style.borderRadius='50px'

</script>

**Function in javascript**

Function is a group of reuseble codedesigned to perform particular task rependently.

A javascript function is executed when call it.

***There are two type of function in js:***

1. **Built -in-function**

The function which are coming along with system interpreter are know as built-in-function.

Example:

Alert()

Prompt()

Open()

setInterval()

etc.

1. **User-define-function**

the function which are developed by user according to business logic are know as UDF.

***There are two type of UDF:***

1. Function (mandatory)
2. Return (optional)

Case1.

Function without parameter

***Syntax:***

***Function functionName()***

***{***

***Line-1***

***Line-1***

***Line-3***

***……….***

***Line-n***

***}***

Value:-

----------value property is use to get/set the value of selected from control.

***Syntax:***

**Get:**

***Document.getElementById(‘text’).value***

***Set :***

***Document.getElementById(“id of html element”).value=”jajgfdbsf”***

**InnerText:-**

----------------innerText property is used to set/get normal text without extra spacing of selected html tag.

***Syntax :***

**Get:**

***Document.getElementById(“txt”).innerText***

**Set :**

***Document.getElementById(“id of html id”).innerText=” text”***

**InnerHTML:-** innerHTML property is use to set/get content with tagand extra page of selected HTML element.

***Syntax;***

**Set:**

***Document.getElementById().innerHTML=” ”***

**Get:**

***Document.getElementById().innerHTML***

**TextContent:-**textcontent property is use to set/get normal with extra space of selected HTML element.

***Syntax;***

**Set:**

***Document.getElementById().textContent=” ”***

**Get:**

**Document.getElementById().textContent**

**Setinterval():-** setInterval() is use to execute a group of statement for every given time period.

It is located in window object.

It has two parameters(callback function ,time period)

***Syntax;***

***setInterval(“function name()”,1000)***

***or***

***setInterval(function name,1000)***

example:

setInterval(demo,2000)

**setTimeout:-** setTimeout function is used to execute once a group of statement after given time period.

***Syntax:***

***setTimeout(“function name()”,1000)***

***or***

***setTimeout(function name,1000)***

example:

setTimeout(demo,2000)

**case2**

function with parameter

**Parameter:-**

Parameter inputs to the function ,if a function contains parameter ,then at the time of calling compulsory we have to provide value to the parameter.

***Syntax;***

***Function functionname(parameter,parmeter…………..n)***

***{***

***Line1***

***Line2***

***……***

***……***

***}***

Example:

Function add(x,y)

{

alert(x+y)

}

Add(10,30)

**Function with return statement**

--------------------------------------------

Return

--------------

Return statement is used to return output to the caller function after execution of business logic.

Example:

function add(x,y)

{

var res=x+y

return res

}

a=add(10,30)

alert(a)

**ClearInterval:-** clearInterval function is used to stop functionalty of setInterval.

HH:MM:SS

**Date object:-** date is predefine object providing some method related to current date and time.

***Syntax:***

***Var obj=new Date()***

***getDay (0-6)***

***getMonth (0-11)***

***getFullYear (2021)***

***getSeconds (0-59)***

***getMinutes (0-59)***

***getHours (0-23)***

***getDate (1-31)***

***getMilliseconds (0-999)***

**Array in Javascript**

**Array in JS:-** IN js array is collection of heterogeneous datatype element.

Indexing of array is start from 0 and last index is length-1.

***Syntax:***

***Var array=[item,item,……….,itemN]***

Example:

Var name=[“ram”,10,23.4,true]

Alert(arr[0])//ram

Alert(arr[1])//10

Alert(arr[2])//23.4

Alert(arr[3])//true

**Length:-**length property is used to return length of array or string.

***Syntax:***

***Alert(arr.lenght)//4***

***Alert(arr[arr.lenght-1])//true***

**Print():-**print function is used to print all document of body section **of** page.

It is located in window object.

***Syntax:***

***Window.print()***

**Concat():-** concat function is used to concat two or more array and return new array.

***Syntax:***

***Var arr1=[item1,item2…….]***

***Var arr2=[item1,item2…….]***

***Var arr3=[item1,item2…….]***

***Var arr4=[item1,item2…….]***

***Arr1.concat(arr2)***

***Arr1.concat(arr2,arr3,arr4)***

**Output:**

**New combine array**

**IndexOf():-** indexof function is used to return index of firstoccurrence element in given array.

If given element is not present in array then it return-1.

***Syntax:***

***Arr.indexOf(item)***

Example:

Var arr=[“HTML”,”CSS”,”JS”,”HTML”]

Alert(arr.indexOf(“js”))//-1

**Reverse():-** reverse function is used to reverse element of given array.

***Syntax:***

***arrName.reverse()***

example:

<sctipt>

Var arr=[“HTML”,”CSS”,”JS”,”HTML”,”PYTHON”,”PHP”]

Alert(arr.reverse[])

</script>

**Sort():-**sortfunction is used to sort element of given array.

***Syntax;***

***Var arr=[“RAM”,”MOHAN”,”GOLU”,”RAVAN0”]***

***Var arr1=[20,30,40,50,10,5]***

***Document .write(arr1.sort())***

***Document .write(arr.sort())***

**Slice():-** slice function is used to return a part of array on the basis of start index and end index.

Negative index is acceptable .

***Syntax:***

***arrayName.slice(startindex,endIndex)***

**Splice():-** splice function is used to add/remove

Element of given array.

***Syntax:***

***splice(start)***

***splice(start, deleteCount)***

***splice(start, deleteCount, item1)***

**Pop():-**pop function is used to remove last element of array and return remove element.

***Syntax:***

***ArrayName.pop()***

Example:

Var arr=[“RAM”,”MOHAN”,”GOLU”,”RAVAN”]

Alert(arr.pop())

**Push():-**push function is used to add element at the last position of array and return length of updated array.

***Syntax :***

***arrayName.push(item)***

example:

var arr=[“RAM”,”MPHAN”,”GOLU”,”RAVAN”]

alert(arr.push(“SOHAN”))

**Includes():-**includes function is used to check given element present in array or not.

If given element present in array then it return true otherwise return false.

***Syntax:***

***Case1:***

***arrayName.includes(item)***

***Case1:***

***arrayName.includes(item,startIndex)***

example:

var arr=[“RAM”,”MPHAN”,”GOLU”,”RAVAN”]

alert(arr.includes(“ram”))//false

alert(arr.includes(“ram”.toUpperCase()))//true

alert(arr.includes(“RAM”,2))//false

**join():-** join function is used to join element of array on the basis separator.

***Syntax:***

***arrayName.join(separator)***

example:

var arr=[“RAM”,”MPHAN”,”GOLU”,”RAVAN”]

arr.join(“||”)//RAM||MOHAN||GOLU||RAVAN

**forEach():-**forEach method is used to call function for every element present in array.

***Syntax:***

***arrayName.forEach(currentvalue,index,arrayName)***

***index and arrayNmae(optional)***

Example :

var arr=[“RAM”,”MPHAN”,”GOLU”,”RAVAN”]

arr.forEach(demo)

function demo(value)

{

Alert(“value”)

}

**Entries():-**entries function is used to return array iterator object with key /value pares.

***Syntax:***

***Var i=arrayName.entries()***

***For(var a of i)***

***{***

***}***

Example:

<script>

var arr=["HTML","CSS","JS","JQUERY","BOOTSTRAP"];

var i=arr.entries();//i is array iterator object

for(var x of i)

{

alert(x)

}

</script>

**Values():-**values function is used to return array iterator object with array item .

***Syntax :***

***Var i=arrayName.values()***

***For(var a of i)***

***{***

***}***

Example:

var arr=["HTML","CSS","JS","JQUERY","BOOTSTRAP"];

var i=arr.values();

for(var x of i)

{

alert(x)

}

**IsArray:-**

**String in Javascript**

**IndexOf:-**function is used to return index of first occurrence character or string in given string.

***Syntax;***

***Var str=”techpile technology”***

Example:

Var str=”techpile technology”

Str.indexOf(“t”)//-1

Str.indexof(“T”)//0

Str.indexOf(“t”,3)//9

Str.indexOf(“o”,3)//14

Str.lastIndexOf(“o”)//18

Str.lastIndexOf(“t”,6)//0

Str.lastIndexOf(“l”,14)//6

**Slice:-**slice function is used to return a part of string in given string on the basis of start Index and End index.

Negative index is acceptable here.

Example:

Vat str=”Techpile Technology”

Str.slice(9,18)//Technology

Str.slice(3,7)//hpil

Str.slice(-4,-2)//lo

Str.slice(-5)//ology

**Substring :-** substring function is used to return a part of string in given string on the basis of start Index and End index.

***Syntax;***

***Str.subString(startIndex,Endindex)***

Example:

Var str=”Techpile Technology Pvt. Ltd”

Alert(str.substring(9,12))//Tec

Alert(str.substring(3))// hpile Technology Pvt. Ltd

**Substr:-** substr function is used to return a port of string in given string on the basis of startIndex and length.

***Syntax:***

***Str.substr(startindex,length)***

Example:

Var str=”TECHPILE”

Alert(str.substr(2,4))//CHPI

**Trim:-**trim function is used to remove both side white space of string.

***Syntax:***

var str=” TECHPILE ”

alert(str.length)//10

var str1=str.trim();

alert(str1.length)//8

**split:-**split function is used to split a string into array of substring based on separater.

***Syntax:***

***Str.splite(separater)***

Example:

Var a=”TCHPILE TECHNOLOGY PVT. LTD.”

a.split(“ ”)//[“TECHPILE”,”TECHNOLOGY”,”PVT.”,”LTD”]

a.split(“L”)//[“TECHPI”,”E TECHNO”,”OGY PVT”,”TD”]

**charAt**:-charAt function is used to return charcter at specified index in a given string.

**Syntax:**

***stringName.charAt(index);***

example:

var str=”techpile”

alert(str.charAt(5));//i

**charCodeAt:-** charCodeAt function is used to return Unicode value of given character at specified index.

***Syntax:***

***stringName.charCodeAt(index)***

example:

var str=”ABC”

alert(str.charCodeAt(2))//67

**includes:-**includes function is used to check whether given string present in given sentence or not.

It return true if given string present in sentence otherwise return false.

**Case-1**

***Syntax:***

***Var str=”TECHPILE TECHNOLOGY”***

***Alert(str.include(“tech”))//flase***

***Alert(str.include(“TECH”))//true***

**Case-2**

***Syntax:***

***Alert(str.include(“TECH”,10))//false***

**Replace:-** replace function is used to replace old string with new string.

By Default replace function first matches string.

***Syntax:***

**Case-1**

***stringName.replace(“oldstring”,”newstring”);***

**Case-2**

**Replace to globle**

***stringName.replace(/changeString/g,”newstring”);***

**startsWith and endsWith:-**

The **endsWith()** method determines whether a string ends with the characters of a specified string, returning true or false as appropriate.

***Syntax:***

***endsWith(searchString)***

***endsWith(searchString, length)***

The **startsWith()** method determines whether a string begins with the characters of a specified string, returning true or false as appropriate.

***Syntax;***

***startsWith(searchString)***

***startsWith(searchString, position)***

**toString:-** toString function is used to convert other datatype value into string.

***Syntax:***

***toString()***

Example:

const stringObj = new String('foo');

console.log(stringObj);

console.log(stringObj.toString());

> String { "foo" }

> "foo"

**typeOf:-**typeof function is used to check type of data presents variable.

***Syntax:***

***typeOf(variablename)***

**Advance Selecter in javascript**

**getElementsByTagName:-**getElementByTagName is used to return collection of all element with specified tag name as collection HTML object.

It is located in document object.

***Syntax:***

***documentByTagName(“TagName”)[indexvalue]***

**getElementByClassName:-**getElementByTagName function is used to return collection of element with specified className as objectHTML

***syntax:***

***document.getElementByClassName(“className”)***

example:

<b class=”a”>Summer traning</b>

<b class=”a”>Winter traning</b>

<b class=”a”>aps traning</b>

Var x=document.getElementByClassName(“a”)[0].textContent

Var x=document.getElementByClassName(“a”)[1].textContent

Var x=document.getElementByClassName(“a”)[2].textContent

**Output:**

Summer traning

Winter traning

aps traning

**querySelector:-**query Selector is used to return

***syntax:***

***var variableName=document.querySelector(“selector”);***

example:

**querySelectorAll:-**querySelectorall function is used to return collection of all matches element with specified selector name as NodeList Object.

Example:

<h1 class=”a”>HTML</h1>

<h1 class=”b”>CSS</h1>

<h1 class=”a”>JS</h1>

<h1 class=”b”>JQUERY</h1>

<script>

Var h=document.querySelectorAll(“h1”)[0].innerText//HTML

Var h=document.querySelectorAll(“h1”)[3].innerText//JQOUERY

Var h=document.querySelectorAll(“.a”)[1].innerText//JS

</script>

**classList:-**classList is a object providing some method related to css class .

add()

remove()

toggle()

contains()

**add():-**add function is used to add css class to select html element.

.a

{

Background:red;

Color:white;

Font-size:100px;

}

Document.getElementById(“box”).classList.add(‘a’)

**Remove();-**remove function is used to remove css class from selected html element.

**Toggle():-**toggle function is used to provide toggle between add and remove function .

**Contains();-**contains function is used to check weather given css class is added or not.

It return true if given css class is added otherwise return false.

**getAttribute():-**getAttribute function is used to get value of attribute of any selected html object.

***Syntax:***

***<img src=”abd.jpg” id=”iml” class=”a”/>***

Document.querySelector(“img”).getAttribute(“src”)//abc.jpg

Document.querySelector(“img”).getAttribute(“id”)//iml

Document.querySelector(“img”).getAttribute(“class”)//a

**setAttribute():-**setAttribute function is used to set attribute value to selected html element.

***Syntax:***

Document.querySelector(“div”).setAttribute(“attributeName”,”value”);

***<img src=”abd.jpg” id=”iml” />***

Document.querySelector(“img”).setAttribute(“class”,”abc”)//

Document.querySelector(“img”).setAttribute(“src”,”abc.jpg”)//

**Math Function In Javascript**

**Math:-** math function is used to object designed to provide some method related to mathmetical operation.

**Random():-**random function id used to return random number between 0(inclusive) and 1(exclusive).

***Syntax:***

***Math.random()***

**Ceil():-**ceil function is used to return largest integer value of given number.

***Syntax:***

***Math.ceil(number)***

**Floor():-** floor function is used to return lowest integer value of given number.

***Syntax:***

***Math.floor(number)***

**Round():-**round function is used to return closest integer value of given number.

***Syntax:***

***Math.round(number)***

**Sqrt():-**sqrt function is used to return square root value of given number.

Negative value is not acceptable here.

If we try to provide negative value then it return NaN.

***Syntax;***

***Math.sqrt(number)***

Example:

Document.write(Math.sqrt(16))//4

**Cbrt():-** cbrt function is used to return cube root value of given number.

Negative value is acceptable here.

***Syntax:***

***Math.cbrt(number);***

Example:

Document.write(Math.cbrt(27))//3

Document.write(Math.cbrt(-27))//3

**Pow():-**pow function is used to calculate power value wrt x power y.

X=base,y=power

***Syntax:***

***Math.pow(x,y);***

Example:

Document.write(Math.pow(2,4))//16

**Min():-**min function is used to return minimum value of given number.

***Syntax:***

***Math.min(number1,number2,……..)***

**Max():-** max function is used to return maximum value of given number.

***Syntax:***

***Math.max(number1,number2,……….)***

**Log():-**log function is used to logarithm value of give number wrt to base e.

***Syntax:***

***Math.log(number);***

Example:

Document.write(math.log(10))//1

**Advance javascript**

**Object:-** object is a collection of property(variable) and methods(funcrion).

***Syntax:***

***Var objectName={propertyName1:value1, propertyName2:value2,………………………. propertyNameN:valueN}***

***Or***

***Var objectName={ propertyName1:value1,methodName1:Methoddefinition1…………}***

Example:

Var stu={name:”shivam”,collage:”csjm”,age:50,

Demo1:function(){

//alert(“hello”)

return "hello"

},

Demo2:function(){

alert(“hello”)

return "hello1"

},

,address:”alld”}

**How to access property and methods from object:-**

***Syntax:***

***objectName.propertyName***

***objectName.methodName();***

Example:

document.write(`Name:${stu.name} and collage:${stu.collage} and age:${stu.age} and ${stu.Demo1()} and ${stu.Demo2()}`);

alert(stu.name)

alert(stu.collage)

alert(stu.address)

**Array of object:-**it is collection multiple object.

***Syntax:***

***Var arrobj=[{},{},{},………{}]***

Example :

Var studentInfo=[{name:”shivam”,collage:”csjm”},{ name:”shivam”,collage:”csjm”},{ name:”shivam”,collage:”csjm”},{ name:”shivam”,collage:”csjm”},]

**How to access element of array of object:-**

studentInfo[0].name//shivam

studentInfo[0].collage//csjm

studentInfo[2].name//shivam

**addEventListener():-** this function is used to add even to the selected HTML element .

we can add more than one event at a time on selected HTML element.

***Syntax:***

***objOfHTMLElement.addEventListener(“Event name”,function Name)***

**(…)Spread operator:-** this function is used to

**(…)Rest operator:-**

**(\*\*)Exponent Operator:-**

**Destructuring of Array:-**

Var arr=[]

**Version of ECMA Script**

**Bootstrap**

Bootstrap is used to create responsive website and application.

Bootstrap was developed by Mark Otto and Jakon Thornton at Twitter.

Bootstrap was released in 2011(latest version is bootstrap 5).

Bootstrap provides predefine classes.

Bootstrap

**Bootstrap Color:-**

Primary

Secondary

Danger

Success

Info

Warning

Dark

Light

**Background:-**

Bg-ColorName

Example:

Bg-success

**Text Color:-**

Text-Colorname

Example:

Text-success

**Spacing:-**

**Padding:** [0-5] p=> top left right bottom, Use=p-4

Pt->padding top

ps->padding left

pe->padding right

pb->padding bottom

px->padding left and right

py->padding top and bottom

**Margin**: [0-5] m=> top left right bottom, Use=m-4

mt->margin top

ms->margin left

me->margin right

mb->margin bottom

mx->margin left and right

my->margin top and bottom

**Width:**

W-25 || width 25%

W-50 || width 50%

W-75 || width 75%

W-100 || width 100%

**Font:-**

Fs || font size(1-6)

Fw || font weight [bold ,bolder ,light ,lighter ,normal]

Fst || font style [italic ,normal]

Text-uppercase

Text-lowercase

Text-capitalize

Text-center

Text-start

Text-end

**Border:-**

Border || top left right bottom

Border-top

Border-bottom

Border-start

Border-end

Border-colorName

Border-(0-5)

**BorderRadius:-**

Rounded||top left right bottom

Rounded-top

Rounded-left

Rounded-right

Rounded-bottom

Rounded-pill

Rounded-circle

**Button:-**

Btn btn-colorName

Btn btn-outline-colorName

**Responsiv of image:-**

img-fluid

Img-thumbnail

**Breaking point:**

Sm || small screen(<=572 px)

Md || medium screen(<=768 px)

Lg || larg screen(<=992 px)

Xl || extra larger(<=1200px)

Xxl || extra extra larger(<=1400px)

**Justify:**

Justify-content-start

Justify-content-end

Justify-content-center

Justify-content-between

Justify-content-around

Justify-content-evenly

Alert alert-color\_Name

**DropdownList:**

<ul class=”list-group”>

<li class=”list-group-item”>html</li>

<li class=”list-group-item”>html</li>

</ul>

**For Background Opacity:**

Bg-opacity-10

Bg-opacity-25

Bg-opacity-50

Bg-opacity-75