JAVA SCRIPT

**Client side script:**

The script is running within the browser is called as clinet side script.

Exa:

Javascript,vbscript,jquery,angular js.

Server side script:

* The script which is running within the web server
* Is called as server side script
* Exa:
* Asp:iis(interent information system)
* Jsp :tomcate/glass
* Php:apache
* Python :apache

Introduction to javascript:

1. Javascript is the script language of html and the web.
2. Javascript is a client side scripting language.
3. Javascript is a dynamic,untyped language.
4. Javascript is interpreted scripting language.
5. Javascript is case sensitive language.
6. Javascript is created by Netscape.
7. Created in may 1995
8. Inveted by Brendan Eich .

Fetures of javascript

1. Js provides client side validation.
2. Js is simple
3. Dom(document object model) 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 .

Alert()

Confirm()

Prompt()

Open()

Location()

setInterval()

clearInterval()

setTimeout()

etc.

exa:

window.alert()

window.confirm()

prompt(): prompt function is used to take input on runtime.

Return type of prompt is string.

Prompt function us lacated in window object.

Syntax: var a=prompt(“filename”,”placeholder”)

**Var keyword:**

Var variavle are globally scoped.

Var varables can be updated re-declared.

**Let keyword:**

Let variables can be updated but not re-decalred.

**Const keyword:**

Const variable neither be updated nor re-declare.

**== vs ===:**

**== equality operatior**

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 conversing

The process of converting one type value to another type is known as type casting or type conversion.

1.parseInt()

Exa;

<script>

var num1=parsInt (prompt("enter first number : "))

var num2=parsInt (prompt("enter second number : "))

alert("result is : "+(num1+num2))

</script>

Or

<script>

var num1=number (prompt("enter first number : "))

var num2=number (prompt("enter second number : "))

alert("result is : "+(num1+num2))

</script>

2.parseFloat()

**Location:** location property is used to redirect location one page to another page

Syntax:

Window.location=”<https://www.techpile.in>”

Exa:

<html>

<head>

</head>

<body>

<script>

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

if (a==true)

{

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

}

</script>

</body>

</html>

**Confirm():** confirm box us used to take confirmation with user.it two buttons.(ok and cancel)

=>if user click on ok button the it returns **true** and if user click on cancel button return **false**.

Syntax:

Window.confirm(“message”)

**Open():** open () is used to open a website on new tab or new window.

Syntax:

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

Exa:

Window.open(“<https://www.techpile.in>”,”\_blank”,”height=400px,width=400px”)

**getElementById():** getElementById function used to select HTML elements on the basis of Id.

Returntype of getElementById function us current object.

=>getelementById function is located in document object.

=>document is the built-in object of DOM providing some property and methods to handle html tagas.

Syntax:

Document.getElementById(“html\_element\_Id”)

Exa:

Document.getElementById(“outer”)

**Function in javascript :** function is a group of reusable code designed to perform particular task repeadly.

* A javascript function is execute when calls it.
* There are two type of function in js
* -----------------------------------------------------------

**1.built in function :**

**=> the** function which are coming along with system interpreter are known as built-in function.

1. Alert()
2. Prompt()
3. Open()
4. setInterval()

etc.

**2 user define function(udf):** the function which are developed by user according to business logic are known as pdf.

There are two keyword prsents to define udf.

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

Case1.

Function without parameter

Syntax;

Function function\_name()

{

Line-1

Line-2

Line-3

------

Line-n

}

Value:

Value property is used to get/set the value of selected form control

Syntax;

Get

-----

Var a=document.getElementById(“txt1”).value

Set

-----------

Var s=Document.getElementById(“IDOfhTMLElement”)

s.value=”xyz”

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

**Syntax:**

Get

<div id=”dv”>TECHPILE</div>

Document.getelementById(“dv”).innerText

Var d=document.getelementById(“dv”)

d.innerText

set

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

Document.getElementById(“dv”).innerText=”TECHPILE TECHNOLOGY PVT .LTD.”

Or

Var a=getElementByID(“dv”)

a.innerText=”TECHPILE TECHNOLOGY PVT LTD.”;

**innerHTML:** innerHTML property is used to set/get content with tag and extra space of selected HTML element.

Set

**----**

**Syntax:**

Document.getElementById(“dv”).innerHTML=”TECHPILE technology”

Get

-----------

<div id=”dv”></h1>javascript, html &css</h1</div>

Var a=document.getElementById(“dv”).innerHTML

**textContent:** textContent prpperty is used to set/get normal text with extra with space of selected HTML element.

Get

<div id=”dv”></h1>javascript, html &css</h1</div>

Var a=document.getElementById(“dv”).textContent

**Output:=> javascript, html &css**

**setInterval():** setinterval function is used to execute a group of statement for every given time period.

=>it is located in window object.

=>is has two parameter(callback function, time period)

Syntax:

Window.setInterval(functionNme,time In MILISECOND)

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

Syntax:

**Window.setTimeout(function\_name,timeperiod);**

Exa:

**Window.setTimeout(“demo()”,2000);**

**Or**

**Window.setTimeout(demo,2000);**

**clearInterval():**

**function with parameter**

**parameter:** parameter inputs to the functions ,if a function contains parameter,that at the time of calling compulsory we have to provided value to that parameter.

**Syntax:**

----------

Function functionName(paramer1,parameter2……..parameterN)

{

Line1

Line2

-

-

}

Exa:

Function add(x,y)

{

alert (x+y)

}

Add(10,30) //calling funcinot

**Function with return statement**

Return

---------

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

Exa:

Function exp(x,y)

{

Res=x\*\*y

Resturn res

}

A=Exp(3,2)

Alert(a)

**clearInterval():** clearInterval function is used to stop functionality of setInterval.

**Date object:** date is a predefined object providing some methods 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)

exa:

x.getDate()//29

x.getMonth()//10

x.getFullYear ()2021

**ARRAY in js :** in js array is collection of heterogeneous datatypes elements.

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

Syntax:

**Var arrayName=[item,item2,item3,……itemN]**

Exa:

Var arr=[“ram”,10,23.5,true]

Printing of array element:

Alert( arr[0]) //ram

Alert(arr[1]) //10

Alert(arr[2]) //23.5

Alert(arr[3]) //true

**Length:**

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

**Exa:**

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

**Alert(arr[arr.length-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()**

**Function of array**

Concat : cancat function is used to concat two or more array and return

Syntaxt:

Var arr1=[item1,item2,item3]

Var arr2=[item3,item4,item6]

Var arr3=[item3,item4,item6]

Arr1.concat(arr2)

Arr1.concat(arr2,arr3)

**indexOf:** indexOf function is used to return index of first occurrence element in given array.

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

**case1 :**

-------

Syntax:

----------  
arr.indexOf(item)

Exa:

Var arr=[“HTML”,”CSS”,”JS”,”HTML”]

Alert(arr.indexOf(“js”)) //-1

Alert(arr.indexOf(“JS”)) //2

Alert(arr.indexOf(“HTML”)) //0

**Case2:**

**Syntax:**

**Arr.indexOf(item,startIndex)**

**Exa:**

Alert(arr.indexOf(“HTML”,2)) //3

Alert(arr.indexOf(“CSS”)) // -1

**lastIndexOf :** lastindexOf function is used to return index of last occurrence element in given array.

If given element is not present

**Case1 :**

Arr.lastIndexOf(item)

Exa:

<Script>

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

<script>

**Case2 :**

**Syntax:**

Arr.lastIndexOf(item,startIndex in backword direction)

<script>

Var arr[“HTML”,””]

</script>

**Reverse:** reverse function is used to reverse element of give array.

Syntax:

arrName.reverse()

exa:

<script>

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

Alert(arr)

Alert(arr.reverse())

</script>

**Sort : sort function is used** sort element of given array

**Syntax:**

**<script>**

**Var arr=[“ram”,”mohan”,”golu”,”ravan”]**

**Var arr1=[10,50,60,20,5]**

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

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

**</script>**

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

Negative index is ecceptable here.

Slice method is does not change the original element

Syntax:

arrayName.slice(startIndext,endIndex) // it returns of array from startindex to endindex -1.

Exa :

<script>

Var arr=[“ram”,”mohan”,”golu”,”ravan”]

Arr.slice(1,3) // mohan,golu

Arr.slice(2) // mohan,ravan

Arr.slice(-2) //golu ravan

Arr.slice(-3,-1) mohan golu

</script>

**Splice () : splice is used** to add/remove element of given array.

Syntax;

arrayName.splice(SatrtIndex,no of Item to delete,items to add)

exa:

Var arr[“ram”,”mohan”,”golu”,”ravan”]

Arr.splice(1,1,”shyam”) // ram , shyam ,golu,ravan

Arr.splice(0,3,”HTML”,”PHP”)

Arr.splice(0,0,”HTML”,”PHP”) // html, php, ram, golu , ravan

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

**Syntax;**

Arrayname.pop() last element

Exa:

Var arr=[“ram”,”shyam”,”mohan”]

Arr.pop() //mohan

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

Syntax;

arrayName.push(item)

exa:

var arr=[“ram”,”mohan”,”golu”,”ravan”]

arr.push(“sohan”)

**shift ():** shift function is used to remove element from first postion of array and return remove element.

**Syntax :** arrName.shift()

Exa:

var arr=["ram","mohan","golu","ravan"]

document.write(arr.shift()) //ram

**Unshift ():** unsift function is used to add element from first postion and return update length of array.

Synax :

arrName.unshift()

exa:

var arr=["ram","mohan","golu","ravan"]

document.write(arr.unshift(“umesh”,”dinesh”)) //6

**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 returns false.

Syntax:

**Case 1:**

Arrayname.includes(item)

Exa;

<script>

var arr=["ramesh","suresh","mohan","sohan"]

document.write(arr.includes("ram")) //false

document.write(arr.includes("mohan")) //true

</script>

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

Syntax:

arrayName.join(separator)

exa:

<script>

var arr=["umesh","mahesg","ramesh","rajesh"]

document.write(arr.join("\*"))

</script>

**ForEach() :** for each method is used to call functioonn for every element present in array.

Syntax:

arrayName.forEach(currentValue,index,arrayName)

index and arrayName is optional

exa:

<script>

var arr=["ram","umesh","shuklla","omp"]

arr.forEach(demo)

function demo(value)

{

alert(value)

}

</script>

OR

<script>

var arr=["ram","umesh","shuklla","omp"]

<!-- arr.forEach(demo) -->

<!-- function demo(value) -->

<!-- { -->

<!-- alert(value) -->

<!-- } -->

arr.forEach(function(currentValue,index,arr)

{

alert(index+""+currentValue);

}

)

</script>

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

Syntax:

Var arr=["HTMl","CSS","JS","JQUERY","BOOTSTRAP"]

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

for(var x of i)

{

alert(x)

}

OUTPUT -> 0,HTML

1,CSS

2,Js

3,JQUERY

4.BOOTSTRAP

**Values():** values function is used to return array object with array items.

Syntax:

Var arr=["HTMl","CSS","JS","JQUERY","BOOTSTRAP"]

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

for(var x of i)

{

alert(x)

}

OUTPUT -> HTML

CSS

Js

JQUERY

BOOTSTRAP

First occuren=

**indexOf() :** indexOf function is used to return index of occurrence character in given string.

Syntax:

Var str =”Techpile Technology”

Str.indexOf(“t”) // -1

Str.indexOf(“T”) //0

Str.indexOf(“T”,3) //9

Str.indexOf(“0”,3) //0

Str.indexOf(“o”,3) //14

**lastIndexOf() :**

exa:

Str.lastIndexOf(“o”) //16

Str.lastIndexOf(“T”,6) //0

chartAt()

charCodeAt()

split()

trim() : trim function is used to remove both side white space of string.

Syntax:

<script>

Var str=” TECHPILE ”

Alert(str.length) //10

Var str=str.trim()

Alert(str.length) //8

</script>

toUppercase()

toLowerCase()

endsWith()

startsWith()

includes()

**slice() :** slice function is used to return a part of string in given string on the basis of startIndex and end inde.

Negative index is acceptable here.

Exa:

Var str=’’Techpile Technology”

Str.slice(3,7) //hpile

Str.slice(9) // techpile

subStr() :

syntax:

str.subString(startIndex,endIndex) // return part of string from startindex to endindex -1.

Exa:

<script>

var str="Techpile Technology"

alert(str.substring(9,12)) // tec

</script>

substring() : substr function is used to retun a part of string in given strin on the basis of startIndex and length.

Syntax:

Str.substr(StartIndex,length)

<script>

EXA:

Var str=”TECHPiLE”

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

</script>

**Split() :** split function is used to split a string inot array of substring based on separator.

Syntax:

Str=split(“separator”)

Exa:

<script>

var str="TECHPILE TECHNOLOGY PVT LTD"

var x=str.split("T") //,ECHPILE ,ECHNOLOGY PV, L,D

document.write(x)

</script>

conCat()

join()

**querySelector :** queryselector is used to return first matches element with specified selector name.

syntax:

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

exa:

<div id=”box”>HTML</div>

<div class=”a”>CSS </div>

<div classs=”a”>JS<h1>ABC</h1> </div>

<div>JQUERY </div>

Var dv=document.querySelector(“div”).innerText //HTML

Var dv=document.querySelector(“#box”).innerText //HTML

Var dv=document.querySelector(“.a ”).innerText //HTML

Var dv=document.querySelector(“.a,#box”).innerText //HTML

Var dv=document.querySelector(“.a h1”).innerText //HTML

**querySelectorAll() :** querySelectorAll function is used to return collection of all matches element with specified selector name as NodeList object.

Exa:

<h1 class=”a”>HTML</h1>

<h1 class=”b”>CSS</h1>

<h1 class=”a”>JS</h1>

<h1 class=”b”>JQ</h1>

<script>

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

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

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

</script>

**classList :** classlist is a object some methods related to css class.

Add()

Remove()

Toggle()

Contains()

**Add() :** add function is used to add class to selected HTML element .

.a

{

Background:red;

Color:white;

Font-size:40px;

Height:100px;

Text-align:center;

Line-height:40px;

}

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 funtionns .

**Contains() :** contains function is used to check wheter given css class is added or not.

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

Exa:

.a

{

Background:red;

Color:white

Font-size:40px

}

<h1 class=”a”>HTMLS CSS AND JS</h1>

**getAttribute () :** getAttribute function is used to get value of attribute of any selected HTMl element.

Syntax:

<img stc=”image/24.jpg” id=”im” class=”a”>

Document.querySelector(“img”) .getAttribute(“src”) // image/24.jpg

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

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

Exa:

<html>

<head>

<style>

</style>

<head>

<body>

<img src="image/24.jpg" height="500px">

<button onclick="demo()">GET</button>

<script>

function demo()

{

var a=document.querySelector("img")

alert(a.getAttribute("height"))

alert(a.getAttribute("src"))

}

</script>

</body>

</html>

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

Syntax:

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

<img src=”image/24.jpg” id=”im” >

Document.querySelector(“im”).setAttribute(“src”,”image/32.jpg”)

**Math :** math is a object designed to provide some methods related to mathematical operation.

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

**Syntax:**

Math.random() //0

Math.random() //

Math.random() //

Math.random() //

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

Math.ceil(5.3) //6

Math.ceil(5.254) //6

Math.ceil(5.844664) //6

Math.ceil(Math.random()\*10)

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

Math.floor(4.446846) //4  
Math.floor(4.55) //4

Math.floor(0.446846) //0

**Round () :** round function is used to return closest integer value.

Math.round (4.55) //5

Math.ceil(5.844664) // 6

Math.round (4.45)//4

**Min ():** min function is used to return minimium value of given numbers.

Syntax:

Math.min(num1,num2,num2,……)

Exa:

Document.write(Math.min(10,50,60,90)) // 10

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

Syntax:

Math.max(num1,num2,num2,……..)

Exa:

Document.write(Math.min(10,50,60,90)) // 90

Pow

**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 returns NaN(not a number) .

**Syntax:**

Match.sqrt(number);

Exa:

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

Document.write(Match.sqrt(-16)) //NaN

Document.write(Match.PI)) //3.14678

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

Negative value is acceptable here.

syntax:

Math.cbrt(numer)

Exa:

Document.write(Match.sqrt(27)) //3

Document.write(Match.sqrt(-27)) //-3

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

X=base value

Y=power value

Syntax:

Math.pow(x,y)

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

Log(): log function is used to return

**Object :** object is a collection of property (variables) and method(functions).

Syntax:

Var obJectName={propertyName:value1,property2:value2,propertyN:valueN}

OR

Var objName={propertyName1:value,methodName:methodDefiniton,……}

EXA:

var student={name:"rohit",college:"CICIST",age:50,address:"chitrakoot",

demo:function (){

document.write("welcome to javvascript 1")},

demo1:function(){document.write("wELCOME TO HTML ")},

branch:"BCA",}

How to access property and method object

Syntax:

objecetName.propertyName

objecetName.Metohd Name

Exa:

document.write(student.name+"<br>");

document.write(student.college+"<br/>");

document.write(student.age+"<br/>");

document.write(student.address+"<br/>");

document.write(student.demo()+"<br/>");

document.write(student.demo1()+"<br/>");

document.write(student.branch+"<br/>");

**PROGRAM:**

**<script>**

**var student={name:"rohit",college:"CICIST",age:50,address:"chitrakoot",**

**demo:function (){**

**document.write("welcome to javvascript 1")},**

**demo1:function(){document.write("wELCOME TO HTML "+"<br/>")},**

**branch:"BCA",**

**}**

**document.write(student.name+"<br>");**

**document.write(student.college+"<br/>");**

**document.write(student.age+"<br/>");**

**document.write(student.address+"<br/>");**

**student.demo()+document.write("<br/>")**

**student.demo1()+"<br>"**

**document.write(student.branch+"<br/>");**

**</script>**

**Arrya of object:** it is collection multiple object.

**Syntax:**

**Var arrobj=[{},{},{},{},{}….{}]**

**Exa;**

**Var studenInfo=[**

**{name:”ram”,college:”CICST”,branch:”CS”},**

**{name:”umesh”,college:”CIC”,branch:”IT”},**

**{name:”OMPRAKSH”,college:”JIC”,branch:”MACHNICAL”}, //trailing comma.**

**]**

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

sudentInfo[0].name //ram

sudentInfo[0].colleg //CICST

sudentInfo[0].branch //CS

studentInfo[1].name //UMESH

studentInfo[1].college //CIC

studentInfo[1].branch //IT

**addEventListenter():**  this function is used to add events to the selected HTML element. We can add morethan one event at a time on selected HTML element.

Syntax:

objOfHTMLelement.addEventListener()

exa:

<html>

<head>

</head>

<body>

<button onclick="fun1()">CLICK ME</button>

<script>

var fun1=()=>

{

alert("i am from fun 1");

}

var fun2=()=>

{

alert("i am from fun 2");

}

var fun3=()=>

{

alert("i am from fun 3");

}

document.querySelector("button").addEventListener("click",fun1)

document.querySelector("button").addEventListener("click",fun2)

document.querySelector("button").addEventListener("click",fun3)

</script>

</body>

</html>

**Rest operator :**

Function demo(…y) // rest operator

{

Alert(y)

}

Demo(10)

Demo(10,20,50)

Demo(10,50,80,”ram”)

Demo(10,85,96,78,.23,21)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Function demo (x,y,…z) // z rest operator

{

}

**Spread operator :**

Function demo(name,…x)

{

}

Var arr=[50,20,30,40,50]

Demo(“ram”,…arr) // spread operator

**Version of ECMA script:**

**Version official name description**

Es1 ECMAScript 1(1977) first edition

ES2 ECMAScript 2(1998) editorial change

Es 3 ECMASCRIPT(1999) added switch

ES 5 ECMAScript(2009)

ES6 ECMAScript 2015 let and const ,

Arrow function,

Default parameter,

Destructuring arrary,

ES2016 javascript exponentiation(\*\*)

**Exponent operator :**

<html>

<head>

</head>

<body>

<script>

var x=3;

var y=3;

//x+=y -> x=x+y

// x\*=y -> x=x\*y;

x\*\*=y // x=x\*\*y

alert(x)

</script>

</body>

</html>

**Destructuring of array:**

<html>

<head>

</head>

<body>

<script>

var demo=(...x)=>

{

return x

}

var arr=["ram","prasad","pandey"]

var [a,b,c]=demo(...arr)

document.write(a+b+c)

</script>

</body>

</html>

**Object.entries():**

**<html>**

**<head>**

**</head>**

**<body>**

**<script>**

**var obj1={name:"uemsh",college:"CICST",age:20,}**

**var a=Object.entries(obj1)**

**for(var [a1,a2]of a)**

**{**

**document.write(`key is ${a1} value is ${a2}<br>`)**

**}**

**</script>**

**</body>**

**</html>**

**Object.values ():**

**<html>**

**<head>**

**</head>**

**<body>**

**<script>**

**var obj1={name:"uemsh",college:"CICST",age:20,}**

**var a=Object.values(obj1)**

**for(var x of a)**

**{**

**document.write(`value is ${x}<br>`)**

**}**

**</script>**

**</body>**

**</html>**