**Web Architecture**

**Web server**

* **Asp**
* **Jsp**
* **Php**
* **Phython**
* **C#**
* **java**

**Client**

* **Html**
* **Css**
* **Js,jq**
* **Angular**
* **Node js**

**http request**

**Database**

* **Ms sql**
* **My sql**
* **Oracle**
* **Mongo db**

**http response**

**Prensentation layer Business logic layer Data access layer**

**Three layers architecture**

**Client Side script**

The script which is running within the browser is called as client side scripting language.

Ex. Java script, vb script,jQuery angular js

**Server side script:**

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

Ex. Asp 🡪IIS(internet information system)

JSP🡪 tomcat/glassfish

Phython🡪apche

Php🡪 apache

**JAVA SCRIPTS:🡪**

Java script is the scripting language of html and the web.

* Java script is a client side scripting language.
* Js is a **dynamic,untyped** language.
* Js is **interpreted** scripting language.
* Js is a **case sensitive** language.

**JAVA SCRIPT is created by NETSCAPE CORPORATION(BRENDAN EICH) in may 1995.**

**Features of javaScript**

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

**WINDOW:🡪**

**Window is a primary object(tags) of java script providing some property and method.**

1. **Alert()**
2. **Confirm()**
3. **Prompt()**
4. **Open()**
5. **Location()**
6. **Set interval()**
7. **Clear** interval**()**
8. **Settimeout()**
9. **Ect….**

**Prompt()🡪** prompt() is used to take input on run time.

🡪return type is string

🡪prompt function is located in window object.

**Syntax🡪**

Var a=promt(“fieldname”, “placeholder”)

**Var🡪** var variasbles are globally scoped.

Var variable can be updated and redeclared.

**Let🡪** let variable can e updated but not re –declared.

**Const**🡪 const variables neither be updated not re-declared.

== **equality operator(** here value is important but not data type.) ex. 10== “10” result🡪 **true**

**=== strict equality operator(**here value and data type both important) Ex. 10=== “10” result 🡪**false**

**Type casting🡪**

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

1. **parseInt()**
2. parseFloat()

Location()🡪 **location** property is used to redirect location one page to another page.

**Syntax🡪**

**Define**  in window object

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

**Confirm()🡪 confirm**  box is used to a confirmation with user . it has two button **OK & CANCLE** .

If ok click on ok button it returns true output and if user click on cancle button it gives false as output.

**SYNTAX🡪**

**Window.confirm(“message”)**

**OPEN ()🡪** open function is used to open a website on a new tab or new window.

**Syntax🡪**

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

Ex🡪

Window.open(“<https://www.techpile.in/st>”,”\_blank”,”height=400px,width=40px”)

**getElementById()🡪getElementById function is used to select html element on thebasis of id.**

**Return type 🡪current object;**

**🡪getElementById function is located in document object.**

**🡪Document is an built-in object of DOM providing some property and methods to handle html tags.**

**🡪Syntax🡪**

**Document.getElementById(“id of html page”)**

**Ex🡪 document.getElementById(“div”)**

**Function is java script🡪**

**Funct**ion is a group of reusable code designed to perform particular task repeatedly.

* A java script function is revoke when it called.
* There are two types of function in js.

1.**Built-in function🡪**

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

Ex.-->

Alert(),

Prompt()

Open()

setInterval()

etc.

**2.User define Function🡪** A function which are developed by user according to business logic are known as UDF.

\*\*\*\*\*\*\*\*\*There are 2 keywords present to define Udf\*\*\*\*\*\*

1.Function(manadetory)

2.return(optional)

**Case 1.---🡪**

**function functionName()**

**{**

**//Line of code//**

**}**

Value🡪

Value  **pr**operty is used to get/set the value of selected form control.

**Syntax🡪**

**------**

**Get🡪 var a=document.getElementById(“txt”).value**

**Set🡪**

**Document.getElementById(“idofhtmlelement”)**

**X.value=”sdsd”**

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

Get🡪

<div id=”dv”>techpile</div>

document.getElementById(“dv”).innerText

var s=document.getElementById(“dv”)

d.innerText

set

---

Document.getElementById(“dv”).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=”<b>TECHPILE Tecchnology<b/>”

**Output🡪**

TECHPILE Tecchnology

Get🡪

**Synatx🡪**

**<div id= “dv”> <h2>** Java Script, HTML & CSS**</h2>>**</div>

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

**output🡪**

**a=**

**<h2>** Java Script, HTML & CSS**</h2>**

**textContent🡪**

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

Set🡪

**Syntax🡪**

document.getElementById(“dv”).textContent=”<b>TECHPILE Technology<b/>”

**Output🡪**

TECHPILE Technology

**Get**

**Syntax🡪**

<div id=”dv”>javaScript, HTML & CSS </div>

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

**output🡪**

javaScript, HTML & CSS

**setInterval()🡪**

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

🡪it is located in window object.

🡪it has 2 parameters(callback function, time periode).

**Syntax🡪**

**setInterval(“function\_name()”,1000) time in millisecond 1sec=100ms**

**or**

**setInterval(function\_name,1000)**

**EX🡪**

**--**

window.setInterval(demo,3000)

**setTimeout()🡪**

* setTimeout() is used to execute once a group of statement after given time period.
* **Window.setTimeout(callback\_function,time\_period)**

**Syntax🡪**

**setTimeout(“function\_name()”,1000) time in millisecond 1sec=100ms**

**or**

**setTimeout(function\_name,1000)**

**clearInterval()🡪**

clearInterval is used to stop functionality of setInterval.

**STOPWATCH:🡪** HH : MM : SS

**Date Object🡪**

**Date is a predefine object providing some methods related to current date and time;**

**Syntax->**

Var obj=new Date()

**1.gatDay (0-6)**

**2.getMonth(0-11)**

**3.getFullYear(2021)**

**4.getSecond(0-59)**

**5.getMinute(0-59)**

**6.getHour(0-23)**

**7.getDate(1-31)**

**8.getMilliSeconds(0-999)**

**Events:🡪**

**Onchange event🡪**

Array **in java Sccript🡪**

**In js Array is a collection of hetrogenius data types elements.**

**Indexing🡪0 to n-1**

**Syntax🡪**

**Var arrayName[item1,item2,item3,….itemN]**

**Ex.-->**

**Var arr[“ram”,10,2.34,true]**

**Arr[0]=ram**

**Alert(arr[0]);🡪 ram**

**Alert(arr[1]);🡪10**

**Alert(arr[2]);🡪23**

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

**Ex🡪**

**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 at window object.

<Syntax-->window.print()

**v.play()**

**v.pause()**

**function of array🡪**

* **Concat()🡪**

**Concat function is used to concat two or more array. And return new array.**

**Syntax🡪**

Var arr1=[item1,item2….itemn]

Var arr2=[item1,item2….itemn]

arr1.concat(arr2)

* **indexOf()🡪**

It is used to represent index of first occurrence element in given array.

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

Indexof function is used to return index of first occurrence character or string in given array

Ex -:

Var str=”Techpile Technology”

Str.indexof(“T”)//0

Case-2 ex-:

Str.indexof(“T”,3)//9

Str.indexof(“o”,4)[\\14](file:///\\14)

Str.lastindexOf(“l”,14)\\6

**Case 1🡪**

**Syntax🡪** --arr.indexOf(item)

Ex->   var arr = ["html", "css", "js", "html"]

Alert(arr.indexof(“JS”)// 🡪-1

Alert(arr.indexOf(js) ///🡪2

**Case2🡪**

**Syntax🡪**

Arr.indexOf(item,start index)

* **lastIndexof()🡪**

**lastIndexOf() is used to return last occurrence element of an array.**

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

**Case 1🡪**

**Syntax🡪** --arr.lastIndexOf(item)

Ex🡪

<script>

        var arr = ["html", "css", "js", "html", "python", "php"]

        alert(arr.lastIndexOf("js"))

        alert(arr.lastIndexOf("python"))

        alert(arr.lastIndexOf("html"))

    </script>

Case2🡪

Syntax🡪 arr.lastIndexOf(item,start index in backword direction)

Ex🡪

     alert(arr.lastIndexOf("python", 2))

        alert(arr.lastIndexOf("html", 4))

* **slice()🡪 slice** () is used to return a part of array on the basis of start index and end index;

Syntax🡪 arrName.slice(start\_index,end\_index)//it return values from start index to end -1.

 <h3>Slice function</h3>

    <script>

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

        alert(arr.slice(1, 3))

        alert(arr.slice(0));

    </script>

* **splice()🡪 splice() is used to add or remove element of given array.it change the real array**

**Syntax🡪**

<h2>Splice function</h2>

    <script>

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

            // alert(arr.slice(1, 3))

            // alert(arr.slice(0));

        alert(arr.splice(1, 1, "shyam"))

        alert(arr)

        alert(arr.splice(0, 0, "html", "php"))

        alert(arr)

    </script>

**-----**

**Arr\_name.splice(startIndex , no of etem to delete , items to add)**

**Pop()🡪**

**------ pop()** is used to delete element from last index. And return remove element

**Syntax🡪**

**--------- arr\_name.pop()/// last element**

Push()🡪

------- push() is used to add element at last index. It returns the length of updated array.

<script>

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

            //alert(arr.pop())

        alert(arr.push("ajay"))

        alert(arr)

        alert(arr.push("shyam", "dsd"))

    </script>

**Unshift and shift function🡪**

**Shift() is used to remove a element from starting point of an array.**

**unshift() is used to add a element from starting point of an array.**

  <script>

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

        alert(arr.shift())

        alert(arr.unshift("goel"))

        alert(arr)

    </script>

* **includes()🡪** include() is used to check given element is present in array or not.

**If** given element is not present in array then returns ture otherwise returns false.

Case1:🡪

<script>

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

        alert(arr.includes("Raju"))

    </script>

Case 2:🡪

Ex.

* **sort()🡪 sort() is used to sort element of given array.**

**Syntax🡪**

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

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

* **reverse()-🡪**

**reverse() is used to reverse element of given array.**

**Syntax🡪**

**arrName.reverse()**

* **join()🡪**

**----------- join() is used to separate the element of array on the basis of given seperator**

**for**Each()🡪

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

forEach() is used to call function for every element present in array.

Syntax🡪

----------.

**Arr\_name.forEach(current\_value , index , arrayName)**

**Ex.=>**    <script>

        var arr = ["ram", "mohan", "golu", "ajay", "sanjay"]

        arr.forEach(demo)

        function demo(value) {

            alert(value)

        }

    </script>

**Isarray()-:**

**Slice()-:**

**Slice function is used to return a part of string in given string on the basis of startIndex and endIndex;**

**Negative index is acceptable here**

  var str="techpile technology";

       // alert(str.slice(9,19))

       //alert(str.slice(-4,-2))

       alert(str.slice(-5))

**substring():-**

**substring(stratindex,indindex)//return part of string from start Index to endindex-1;**

**substr()-:**

substr function is used to return a part of string in given string on the basis of start index and length.

Syntax-:

Str.substr(startindex,length)

Ex-:

 var str="techpile technology"

        alert(str.substr(2,6))\\chpile

**Trim()-:**

**Trim function is used to remove** both side white spece of string.

Syntax-:

Var str =” techpile”;

Var str1=str.trim();

Alert(str1.lenght)//8

**Split()-:**

**Split function is used to** split a string into array of substring based on separator.

Syntax-:

Atr.split(separator);

Ex-:

  var a="techpile technology pvt ltd";

        alert(a.split(" "))//

charAt()-:

charAt function is function is used to return character at specified index in a given string

**syntax-:**

string\_name.charAt(index);

ex-

var  str="techpiletechnology"

        alert(str.charAt("2"))//output-c

charCodeAt()-:

charcodeAt function is used to return Unicode(asscii)value of given charecter at specified index.

Syntax-:

string\_name.charcodeAt(index);

ex-:

 var  str="techpiletechnology"

        alert(str.charAt("5"))

        alert(str.charCodeAt("i"))//116

includes()-:

includes function is used to check wheater 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”]

Alert(str.includes(“TECH”))//false

Var str=[“techpile”]

Alert(str.includes(“tech”))//true

Case-2:

Alert(str.includes(“tech”,start index,10)//false

**Replace()-:**

Replace function used to replace old string with new string.

By default replace function replaced first matched string.

**Note-:**

If all string change by replace function

Replace to globle :syntax-:

        document.write(str.replace(/tech/g,"TECH"))

**Syntax-:**

String\_name.replace(old string.newstring);

Ex-:

var str="techpile";

        alert(str.replace("tech","ompr"))

**startswith()-:**

**endsWith()-:**

**ex-:**

       /\* var name="Techpile technology pvt ltd"

        alert(name.startsWith("Tech"))

        alert(name.startsWith("techno"))

        alert(name.startsWith("pvt"))

        alert(name.endsWith("Tech"))

        alert(name.endsWith("pvt"))--\*/

**tostring()-:**

to string function is used to convert other datatype value in to string.

 var a=10

        alert(typeof(a))

        a=a.toString()

        alert(typeof(a))

Typeof()-:

Type of function is used to check type of data present in variable.

 var a=10

        alert(typeof(a))

uppercase()-:

lowercase()-:

**Advance selector-:**

1. getElementById();
2. getElementsByTagName()-:

getElementsByTagName function is used to return collection of all element with specified tag name as collection HTML object.this function also located in document object.

Syntax-:

Var a=document.getElementsByTagName(“tagname”)

Ex-:

<h1>HTML<h1/>

<h1>CSS<h1/>

<h1>JS<h1/>

<h1>JQUERY<h1/>

Var a=document.getElementsByTagName(“h1”)[0].innerHTML;

Var a=document.getElementsByTagName(“h1”)[1].innerHTML;

Var a=document.getElementsByTagName(“h1”)[2].innerHTML;

Var a=document.getElementsByTagName(“h1”)[3].innerHTML;

1. getElementsByClassName():-

getElementsByClassName() function is used to return collection of element with specified class name as object HTML collection.

Syntax-:

Var variable\_name=document.getElementsByClassName(classname)

Ex-:

<b class=”a”>summer tranning</b>

<b class=”a”>winter tranning</b>

<b class=”a”>apprenticeship tranning</b>

Var x= ocument.getElementsByClassName(“a)[0].textcontent;//summer tranning

Var x= ocument.getElementsByClassName(“a)[1].textcontent;//winter tranning

Var x= ocument.getElementsByClassName(“a)[2].textcontent;//apprenticeship tranning

1. querySelector()
2. querySelectorAll()

**blur**event()-:

focus event()-:

**math function()-:**

**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(not a number).

**Syntax-:**

Math.sqrt(number);

Ex-:

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

Document.write(Math(-16))//NaN

**Math.pi()-:**

**Syntax-:**

Document.write(Math.PI)//3.14678;

**Cbrt()-:**

Cbrt function is used to return cube root value of given number.

Negative value acceptable here.

**Syntax-:**

Math.sqrt(number);

Example-:

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

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

**Pow()-:**

Pow function is used to calculate power value with wrt x power y.

X=base value.

Y=pow value.

**Syntax-:**

Math.pow(x,y)

**Ex-:**

Document.write(Math.pow(x,y))//

**Min()-:**

Min function is used to return minimum value of given number.

**Syntax-:**

Math.min(num1,num2,….,numN)

**Ex-:**

Document.write(Math.min(10,20,30,40,50))//10

**Max()-:**

Max function is used maximum value is given numbers

**Syntax-:**

Math.max(num1,num2,….,numN)

**Ex-:**

Document.write(Math.max(10,20,30,40,50))//50

**Log()-:**

Log function is used to return logarithm value of given number wrt to base e.

**Syntax-:**

Math.log(number)

**Ex-:**

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

**Example of parameterized function-:**

 <script>

        var obj={

     name:"ompraksh",

     college:"cicst",

     msg:function demo(x,y){

        //alert("ok");

        //return "omprakash"

        return x+y;

     },

     age:30,

        };

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

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

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

        document.write("Result is parametarized function :"+obj.msg(10,20));

    </script>

**Note-:**

//{``}//it is called iterpolation || template string

**Arrow function(🡪)-:**

Arrow function is use like unnamed function

Syntax-:

Ex-:

Object-:

Object is collection of property(variable) and method(function).

Haw to access property and method.

Syntax-:

Var obj={property name,:value1,property name2:value2,………….,property value n:value n}

Or

Var object name={ property name,:value1/method name :method definition ,………….,method name n:method definition n }

Ex-:

Var student={name:”omprakash”,college:”cicst”,age:19,address:”cktd”}

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

Syntax- for property access-:

Object name .property name;

Object name .method name//function name();

Ex-:

 <script>

        var  student={

name:"omprakash",

        college:"cicst",

        age:19,

        address:"cktd",

    demo:function (){

document.write("i am from fun 1"+"<br/>")

    },

    demo1:function(){

        document.write("i am from fun2"+"<br/>")

    },

    branch:"cs",

}

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

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

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

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

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

student.demo()

student.demo1()

    </script>

Syntax-:

Var obj={

Property name1:value;,

Function name()

};objname.function name;

**Array of object-:**

It is multiple collection of object.

Syntax-:

Var arrobj=[{},{},{},{},{}];

Ex-:

Var studentinfo=[

{name:”ram”,college:”cicst chitrakoot”,age:25,branch:”cs”}, {name:”ravi”,college:”cicst chitrakoot”,age:20,branch:”it”}, {name:”rohit”,college:”cic chitrakoot”,age:21,branch:”mba”},//talling comma];

Haw to access element of array of object-:

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

Studentinfo[0].name//ram

Studentinfo[0].college//cicst chitrakoot

Studentinfo[0].branch//cs

Studebtinfo[0].age//25

Studentinfo[2].name//rohit

Studentinfo[1].college//cicst chitrakoot

**AddevEventlistenter()-:**

This function is used to add event to the selected html element.

We can add more than one event at a time selected html element.

Syntax-:

Object of html element.addeventListenter(“eventname”,”function name)

Ex-:

Document.queryselector(“button”).addeventlistenter(“click”,demo1());

Document.queryselector(“button”).addeventlistenter(“click”,demo2());

Document.queryselector(“button”).addeventlistenter(“click”,demo3());

]

Note-:

For making rest operator

Syntax-:

Function(…a,b){}

Ecma= European computer manufacturing association

Enternet explore-java script

Netscape-vb(visual basic).

Ecma virsion(ES1)--script first (1997) first addition

Ecma script 2 virsion(ES2)-- (1998) editorial changes

Ecma script 3 virsion(ES3)-- (1999) added switch

Added do-while

Ecma script5 virsion(ES5)—(2009)—

String.trim(),array,isarray(),foreach(),multiline string object,array of object,resorve word as property ,tralling commos, Etc.

Ecma script 6 virsion(ES6)—(2015)---

Let and const arrow function default parameter destructuring array itarble ,object ,entries itrable object

Ecma script (virsion(ES)—(2016)—

Java script exponentation(\*\*) javascript exponentiation(\*\*=)

virsion(ES)—2017—

object.entries()-:

object.values()-: