Array : Array is collection of elements. Javascript is loosely typed script so we can store any type of values in arrays.

Array is collection of elements each element is combination of index and value. Bydefault first element index starts with 0 and last element index is total number of elements-1

```
<script>
var arr=[10,20,30,40,'scott']
alert(arr)
alert(arr[4])
</script>
```

length : using this property we can get total number of elements

```
<script>
var arr=[10,20,30,40,'scott']
alert(arr.length)
</script>
```

---

push() : To add an element at the end of array and returns total number of elements

```
<script>
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
var arr=[10,20,30,40,50]

var rv=arr.push(60)

alert(arr)

alert(rv)

</script>
```

pop() : Removes the last eleemnt of array and returns value of that element

```
<script>

var arr=[10,20,30,40,50]

var rv=arr.pop()

alert(arr)

alert(rv)

</script>
```

shift() : To remove the first element of array and returns value of that element.

```
<script>

var arr=[10,20,30,40,50]

var rv=arr.shift()

alert(arr)

alert(rv)

</script>
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

unshift() : To add an element at the begining of array and returns total number of elements.

```
<script>
var arr=[10,20,30,40,50]
var rv=arr.unshift(2)
alert(arr)
alert(rv)
</script>
```

slice() : To get some part of array.

```
<script>
var arr=[10,20,30,40,50]
var rv=arr.slice(1,4)
alert(rv)
</script>
```

splice() : To add/remove the elements from array based on specified index.

```
<script>
var arr=[10,20,30,40,50]
arr.splice(3,2,111,222,333)
alert(arr)
</script>
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

for..in : To run a loop through the elements of array and get index of every element

```
<script>

var arr=[10,20,30,40,50]

for(var x in arr){

        alert(x)

        }

</script>
```

for..of : same as for..in but holds value of elements.

```
<script>

var arr=[10,20,30,40,50]

for(var x of arr){

        alert(x)

        }

</script>
```

---

forEach() : It is combination of for..in and for..of to run a loop throught the elements of array.

we can also execute a function for every element.

```
<script>

var arr=[10,20,30,40,50]
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
arr.forEach(function(x,y){
        alert(x)
        alert(y)
})
</script>
```

map() : Same as forEach but it also returns values. Total number of return values are equal to total number of elements.

```
<script>
var arr=[10,20,30,40,50]
var narr=arr.map(function(x,y){
        alert(x)
        alert(y)
        return x+y
})
alert(narr)
</script>
```

filter() : Runs a loop throught the elements of array and returns truth values.

```
<script>
var arr=[10,20,30,40,50]
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
var narr=arr.filter(function(x,y){
        return x > 30
})
alert(narr)
</script>
```

---

split() : using this function we can split a string as array elements based on input delimiter

```
<script>
var str="welcome to scott"
var arr=str.split("c")
alert(arr)
alert(arr[0])
</script>
```

join(): To join the elements of array as string based on input value we can use join

```
<script>
var arr=[10,20,30,40,50]
var str=arr.join("/")
alert(str)
</script>
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

function : function is set of executable statements to perform a task. functions we can use for reusable. functions are 2 types

1) pre-deefined functions : These functions are available with javascript

Ex: alert,prompt,...

2) user-defined functions : The functions which we are creating at the time of application devp comes under user-defined functions.

using function keyword we can create functions

```
<script>
function fun1(){
        var x=100;
        var y=200
        var z=x+y
        alert(z)
        }
fun1()
</script>
```

We can pass arguments into function.

```
<script>
function fun1(x,y){
        var z=x+y
        alert(z)
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
        }
fun1(10,20)
fun1(111,222)
</script>
```

we can return a value from a function. we can call it as function with return type

```
<script>
function fun1(x,y){
        var z=x+y
        return(z)
        }
var rv=fun1(10,20)
alert(rv)
var rvv=fun1(100,200)
alert(rvv)
</script>
```

event : event is an action performing on control. everhy control contains events(click, mousemove,mouseover,...)

we can execute functions(listeners) for events

```
<script>
function fun1(){
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
        alert("Func exec..")
        }
</script>
<body>
 <input type="button" value="Click" onclick="fun1()"/>
</body>
```

```
<body>
 <img src="mango.jpg" width="100px" onclick="alert('hiiii');alert('john')"/>
</body>
```

this : this refers current control object

```
<body>
 <img src="mango.jpg" width="100px"
 onclick="this.src='orange.jfif'" onmouseover="this.src='smile.png'"
 onmouseout="src='mango.jpg'"/>
</body>
```

Id : using this property we can provide identity to a control.

```
<body>
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
 <img src="mango.jpg" width="100px" onclick="i1.src='all.jpg'"/>

 <img src="smile.png" width="100px" height="80px" id="i1"/>

</body>
```

```
<body>

 <img src="all.jpg" width="600px" height="400px" id="i1"/>

 <br />

 <img src="mango.jpg" width="200px" height="150px"
onclick="i1.src=this.src" />

 <img src="all.jpg" width="200px" height="150px" onclick="i1.src=this.src"/>

 <img src="smile.png" width="200px" height="150px"
onclick="i1.src=this.src"/>

</body>
```

---

string : string is collection of characters. Every character in string contains index number.

length : using this property we can get total number of characters in a string

```
<script>

var str="welcome"
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
alert(str.length)

alert(str[0])

</script>
```

---

toUpperCase() : Using this we can convert the characters into uppercase

toLowerCase() : To convert the characters into lower case

```
<script>

var str="Hello"

alert(str.toUpperCase())

alert(str.toLowerCase())

</script>
```

charAt() : To get the character of specified index from string

```
<script>

var str="Hello"

alert(str.charAt(1))

</script>
```

charCodeAt() : To get ASCII value of input character from a string

A - 65

B - 66

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

C - 67

D - 68

Z - 90

a - 97

b - 98

z - 122

0-48

1-49

9-57

space - 32

enter - 13

bksp - 8

tab - 9

esc - 27

```
<script>
var str="hello"
alert(str.charCodeAt(0))
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
</script>
```

---

indexOf() : To get the index number of input chanracter from a string.

```
<script>
var str="welcome sctt hdhdhhdhdhd jdjdjhdhdh kskksojdjdj"
for(var i=0;i<str.length;i++){
        alert("i is "+i)
        var ind=str.indexOf("o",i)
                alert("index is "+ind)
        if(ind == -1)
        break;
        i=ind

        }
</script>
```

---

replace: To replace a string with new string

replaceAll : It is same as replace but replaces all the matches

```
<script>
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
var str="welcome scott amith scott"

alert(str.replaceAll("scott","rajesh"))

</script>
```

---

substr() : To get substring of input string

substring() : To get substring of input string

```
<script>

var str="welcome"

alert(str.substring(2,4))

</script>
```

---

match() : Check the matches of string and returns those matches. If no match is found it returns null

```
<script>

var str="welcome scott"

alert(str.match("scott"))

</script>
```

```
<body>
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
 <input type="text" onblur="fun1(this)"/>
</body>
<script>
function fun1(t){
        t.value=(t.value.split(" ").join(""))
        }
</script>
```

Conversion Functions : using these function we can conver the data from one format to another format.

parseInt() : using this function we can convert input value as integer value.

```
<script>
var x=100
var y="200"
alert(x+parseInt(y))
</script>
```

```
<script>
var x="1scott00"
alert(parseInt(x))
</script>
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

parseFloat() :

```
<script>
var x="100.12"
alert(parseFloat(x))
</script>
```

---

eval() : Using this function we can evaluate a string as arthemetic expression

```
<script>
var x="100+50*2"
alert(eval(x))
</script>
```

---

isNaN() : using this function we can check input value is number or not.

```
<script>
var x="scott"
alert(isNaN(x))
</script>
```

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

Date() : Date is an object to get the information of current date and time. It provides many methods

getDate()

getTime()

getHours()

getMinutes()

getSeconds()

getMonth()

getYear()

....

Ex:

```
<script>
var dt=new Date()
alert(dt.getDate())
alert(dt.getMonth())
alert(dt.getYear())
alert(dt.getFullYear())
alert(dt.getHours())
alert(dt.getMinutes())
alert(dt.getSeconds())
</script>
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

Math : Using this object we can work with mathematical functions.

Math.random() : To get the random number between 0 and 1

```
<script>
alert(Math.random())
</script>
```

Math.round() : Rounds input value as to its nearest integer value.

```
<script>
alert(Math.round(10.3))
</script>
```

Math.floor() : converts input value as nearest lowest integer value

```
<script>
alert(Math.floor(9.89))
</script>
```

Math.ceil() : convert input value as nearest highest integer value

```
<script>
alert(Math.ceil(9.23))
</script>
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

Math.sin() : To get sin value of input value

Math.con() : To get cosin value of input value

Math.tan() : To get tangent value of input value

Object: object is collection of properties and methods.

property is name and value pair. method is a function inside the object.

Object we can create in many ways

1) Using object literals {}

BOM (Browser Object Model) : It provides many functions to work with applications.

window

document

history

navigator

location

screen

Note : window object members we can access witt/without object name

<script>

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
var obj={
        uname:"scott",
        city:"hyd",
        fun1:function(){
                alert("Func exec...")
                }
        }
        alert(obj)
        alert(obj.uname)
        obj.fun1()
</script>
```

---

uname - scott city -hyd

uname - john city - chennai

```
<script>
var obj=[{uname:"scott",city:"hyd"},
{uname:"john",city:"chennai"}]
alert(obj[0].uname)
alert(obj[1].uname)
</script>
```

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

uname - scott

wife sw

child

sons - s1,s2

Ex:

```
<script>
var obj={
        uname:"scott",
        city:"Hyd",
        child:["s1","s2"]
        }
        alert(obj)
        alert(obj.uname)
        alert(obj.child[0])
</script>
```

---

uname - scott

wives -

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

name - w1

child - sons - s1,s2

name - w2

child - sons -w2c1,w2c2

Ex:

```
<script>
var obj={
        uname:"scott",
        wives:[{
                name:"w1",child:{sons:["s1","s2"]}
        },{
                name:"w2",child:{sons:["w2s1","w2s2"]}
        }]
        }
        alert(obj.wives[0].name)
        alert(obj.wives[0].child.sons[0])
</script>
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

---

cmp - hyd - pro - hp1,hp2 ad - had1,had2

bang - prog - bp1,bp2 hr - bhr1,bhr2

```
<script>
var cmp={hyd:{
        prog:["p1","p2"],admin:["a1","a2"]
},bang:{
        prog:["bp1","bp2"],hr:["bhr1","bhr2"]
}}
alert(cmp)
alert(cmp.bang.prog[0])
</script>
```

We can create object using Object.create() function.

```
<script>
var obj=Object.create({
        uname:"scott",
        city:"hyd"
})
alert(obj)
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
alert(obj.uname)

alert(obj.city)

</script>
```

---

we can also create object using class

---

window : It is root object of browser provides many properties and methods.

alert(): using this ,ethod we can display message box on browser

```
<script>

window.alert("welcome")

</script>
```

prompt() : using this function we can display input dialog box

```
<script>

var rv=prompt("Enter Name")

alert(rv)

</script>
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

confirm() : To display confirmation dialog box

```
<script>
var rv=confirm("You want to close?")
alert(rv)
</script>
```

print() : To display print properties dialog box

```
<script>
print()
</script>
```

Ex:

```
<script>
function fun1(){
        var rv=confirm("You want to print?")
        if(rv)
         print()
        }
</script>
<body>
<input type="button" value="Click" onclick="fun1()"/>
</body>
```

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

location: using this property we can redirect from current page to another page

Ex:

```
<script>
function fun1(){
        location="http://google.com"
        }
</script>
<body>
<input type="button" value="Click" onclick="fun1()"/>
</body>
```

---

open(): using this function we can redirect from one page to another page

Ex:

```
<script>
function fun1(){
        window.open("http://google.com","_self")
        window.open("http://fb.com","_blank")
        window.open("http://durgasoft.com","_blank","width=300,height=300")
        }
</script>
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
<body>

<input type="button" value="Click" onclick="fun1()"/>

</body>
```

---

setTimeout() : using this function we can call another function after specified time

```
<script>

function fun1(){

        alert("Exec...")

        }

        setTimeout("fun1()",5000)

</script>
```

setInterval() : using this function we can call another function for every regular intervals of time

```
<script>

function fun1(){

        alert("Exec...")

        }

        setInterval("fun1()",3000)

</script>
```

clearInterval() : To stop the functionality of setInterval

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

Ex:

```
<script>
function fun1(){
        alert("Exec...")
        }
        var t=setInterval("fun1()",3000)
</script>
<body>
 <input type="button" value="Stop" onclick="clearInterval(t)"/>
</body>
```

document : Using this object we can work with current document

document.write() : Using this function we can write some content on current document

Ex:

```
<script>
window.document.write("welcome scott")
</script>
```

---

document.createElement() : Using this function we can create ne element in current document.

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

document.body.appendChild(): To append an element as child element of current document.

```
<script>
function fun1(){
        var con=document.createElement("input")
        con.type="file"
        document.body.appendChild(con)
        var con1=document.createElement("div")
        con1.innerText="Welcome"
        document.body.appendChild(con1)
        var con2=document.createElement("img")
        con2.src="mango.jpg"
        document.body.appendChild(con2)
        }
</script>
<body>
 <input type="button" value="Click" onclick="fun1()" />
</body>
```

---

document.getElementById() : Using this function we can get an element from the current document based on id of the element.

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

Ex:

```
<script>
function fun1(){
        var txt1=document.getElementById("t1")
        txt1.value="Enter text"
        alert(txt1.value)
        }
</script>
<body>
 <input type="button" value="Click" onclick="fun1()"/>
 <input type="text" id="t1"/>
</body>
```

Ex:

```
<script>
function fun1(ch){
        var txt1=document.getElementById("t1").value
        var txt2=document.getElementById("t2").value
        if(ch=='a')
        document.getElementById("t3").value=parseInt(txt1)+parseInt(txt2)
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
        else if(ch=='s')
        document.getElementById("t3").value=parseInt(txt1)-parseInt(txt2)
        }
</script>
<body>
 <input type="text" placeholder="No1" id="t1"/>
 <br />
 <input type="text" placeholder="No2" id="t2"/>
 <br />
 <input type="text" placeholder="res" id="t3"/>
 <br />
 <input type="button" value="+" onclick="fun1('a')"/>
 <input type="button" value="-" onclick="fun1('s')"/>
</body>
```

```
<body>
<div id="div1">Hi</div>
<input type="button" value="Stop" onclick="funStop()" />
</body>
<script>
function fun1(){
        var dt=new Date()
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
        document.getElementById("div1").innerText=dt.getHours()+":"+dt.get
Minutes()+":"+dt.getSeconds()
        }
        var t=setInterval("fun1()",1000)
        function funStop(){
                clearInterval(t)
                }
</script>
```

---

document.title : To get/set title of current document

```
<script>
function fun1(){
        document.title="Mysite"
        alert(document.title)
        }
</script>
<body>
 <input type="button" value="Click" onclick="fun1()" />
</body>
```

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

navigator : using this object we can get browser information.

history : using this object we can get browser history

pro.html

```
<script>


</script>
<body>
<h1>This is pro</h1>
 <a href="pro2.html">Pro2</a>
</body>
```

pro2.html

```
<body>
This is pro2
 <a href="pro3.html">Pro3</a>
<input type="button" value="Prev" onclick="fun1()"/>
<input type="button" value="Next" onclick="fun2()" />
</body>
<script>
function fun1(){
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
        history.back()
        }
function fun2(){
        history.forward()
        }
</script>
```

pro3.html

```
<body>
<h2>This is pro3</h2>
</body>
<body>
Pro3
<input type="button" value="Click" onclick="funGo()" />
</body>
<script>
function funGo(){
        history.go(-2)
        }
</script>
```

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

Types of functions :

Anonymous Function : If we create a function without any name comes under anonymous function.

Ex:

```
<script>
var fun1=function(){
        alert("From anonymous function")
        }
        //alert(fun1)
        fun1()
</script>
```

callback function: If we pass a function as the argument of another function we can use this concept.

```
<script>
function fun1(ref){
        alert("fun1 exec...")
        ref()
        }
function fun2(){
        alert("fun2 exec...")
        }
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
function fun3(){
        alert("Fun3 exec..")
        }
fun1(fun2)
fun1(fun3)
</script>
```

Ex2:

```
<script>
function validateValues(no1,no2,ope){
        if(isNaN(no1)){
                return "No1 is not valid"
                }
        else if(isNaN(no2)){
                return "No2 is not valid";
                }
        else{
                return ope(no1,no2)
                }
        }
function sum(n1,n2){
        return(n1+n2)
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
        }
function mul(n1,n2){
        return(n1*n2)
        }
var rv=validateValues('a',200,sum)
alert(rv)
var rv=validateValues(100,200,mul)
alert(rv)
</script>
```

Nested Function : A function inside another function comes under this concept.

Ex:

```
<script>
function fun1(){
        alert("Fun1 exec...")
        function fun2(){
                alert("Fun2 exec...")
                }
                fun2()
        }
        fun1()
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
</script>
```

Ex:

```
<script>
function fun1(){
        alert("Fun1 exec...")
        return function fun2(){
                alert("Fun2 exec...")
                }
        }
        var rv=fun1()
        rv()
</script>
```

Arrow Function : It is the short format of a function. We can use arrow functions to place nested functions inside the class.

```
()=>{}
```

Ex:

```
<script>
fun1=()=>{
        alert("Fun1 exec...")
        }
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
fun1()
</script>
```

Ex:

```
<script>
var arr=[10,20,30,40,50]
var rv=arr.map(x=>x*10)
        alert(rv)
</script>
```

Types of variables:

Local Variable : Variable declaration inside the function comes under local variable. Local variable of a function we can not access from another function.

Ex:

```
<script>
function fun1(){
        var x=100;
        alert(x)
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
        }
function fun2(){
        alert("From fun2")
        alert(x) // local of fun1
        }
        fun1()
        fun2()
</script>
```

Global Variable : Variable outside all functions. we can access this variable from any function.

```
<script>
var x=333
function fun1(){

        alert(x)
        }
function fun2(){
        alert("From fun2")
        alert(x)
        }
        fun1()
        fun2()
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
</script>
```

Block Scope variable : Variable declaration inside a block comes under this concept. We can not access this variable from outside the block. block scope variables we can declare using let keyword

var : Function scope

let : Block scope

```
<script>
function fun1(){
        {
        var x=100
        let y=200
        }
        alert(x)
        alert(y)
        }
fun1()
</script>
```

Lexical Scope : It is outer scope variable of inner function.

```
<script>
function funouter(){
        var x=100
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
        function funinner(){
                alert("Inner function")
                alert(x)
                }
                funinner()
        }
        funouter()
</script>
```

Hoisting : It is a concept of declaration of the variable before starting the execution of script.

Var contains undefined but not let

Ex:

```
<script>
alert(x)
var x=100
alert(x)
</script>
```

Ex:

```
<script>
fun1()
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
function fun1(){
        alert("Func exec...")
        }


</script>
```

OOPS : Javascript supports OOPs concepts to provide reusable and security to application.

class: class is collection of members those are properties and methods

Property : Variable declaration inside the class we can call as property

Method : Function declaration inside the class we can call as method.

new : using this keyword we can allocate memory to class.

object : It is memory allocated to class. We can access class members using object

```
<script>
class cls1
{
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
        sno=100;
        fun1(){
                alert("hi")
                }
}
var obj=new cls1()
alert(obj.sno)
obj.fun1()
</script>
```

super(): to get the members of parent class from derived class.

```
<script>
class cls1
{
        fun1(){
                alert("fun1 from class1")
                }
}
class cls2 extends cls1
{
        fun1(){
                alert("Fun1 from der class")
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
        }
    fun2(){
        super.fun1()
        this.fun1()
        alert("fun2 from class2")
        }
}
var obj=new cls2()
obj.fun2()
</script>
```

```
<script>
class cls1
{
    sno=100
    fun1(){
        alert(this.sno)
        let funinner=()=>{
            alert("From inner function")
            alert(this.sno)
            }
            funinner()
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
        }
}
var obj=new cls1()
obj.fun1()
</script>
```

---

rest : we can use to get the rest of array elements into variables.

```
<script>
var arr=[10,20,30,40,50,60,70,80]
var [x,y,...z]=arr
alert(x)
alert(y)
alert(z)
</script>
```

spread : To spread the elements of array into variables

```
<script>
var arr=[10,20,30,40,50,60]
function fun1(x,y,z,a,b,c){
    alert(x)
    alert(y)
    alert(z)
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
        alert(a)
        alert(b)
        alert(c)
        }
fun1(...arr)
</script>
```

default : using this we can set a default value to argument of function.

```
<script>
function fun1(x,y=222){
        alert(x+y)
        }
        fun1(10,1)
</script>
```

Object Destructure : To create variables based on the properties of object. variable names are property name and values are property values

p1.html

```
<script>
localStorage.setItem("sno",100)
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
alert(localStorage.getItem("sno"))
</script>
<body>
 <a href="p2.html">Go next</a>
</body>
```

p2.html

```
<script>
alert("hi")
alert(localStorage.getItem("sno"))
localStorage.setItem("uname","scott")
</script>
<body>
This is p2
</body>
```

Ajax : Ajax is web technology to send a request to server with out submit the current web page. Sometimes according to the requirement we need to send a request to server with out submit web page, then we can take the support of ajax.

Ajax stands for Asynchronous Javascript and XML.

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

Asynchronous : It is a process of sending a request to server with irrespective of previous response.

Javascript : using javascript we can create ajax object and send that object to server.

xml : Data between browser and server transfers in the form of xml

AJAX we can use to get informatin from another page of current site or from another site.

We have many methods to send a request to server

get: This method we can use to send a request without any data

post : To send a request to server with some data

put : To send a request to update data in server

delete : To send a request to delete the data in server

----

axios() : using this function we can send a request through ajax

fetch() : We can also use this function to work with AJAX

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

---

then() : using this function we can execute some statements when the previous function execution is completed.

we can link then with ajax object, promise object.

Promise : In javascript promise is an object to apply Asynchronous functionality on html control.

Bydefault in javascript functions are synchronous to apply asynchronous functionality we can use promise object.

promise can return two values

resolve : To resolve promise with success result

reject : To reject promise with failure result

then() : This function executes when the promise object execution is completed.

we need to pass 2 arguments

1) A function to execute when promise returns success value

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

2) A function to execute when the promise returns failure value

```
<script>
function fun1(){
return new Promise(function(resolve){
  return resolve("fun1")
})
}
function fun2(){
console.log("fun2 exec...")
}

fun1().then(function(dt){
console.log(dt)})

fun2()
</script>
<body>
Welcome
</body>
```

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
<script>
function fun1(no){
return new Promise(function(resolve,reject){
if(isNaN(no))
return reject("Invalid")
else{
 no++
 return resolve(no)
}
})
}

fun1(100).then(function(dt){
        alert(dt)
        fun1('abcd').then(function(dtt){
        alert(dtt)
                fun1(dtt).then(function(dttt){
                alert(dttt)
                },function(error){
                alert(error)
                })
        },function(err){
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
        alert(err)
        })
},function(er){
        alert(er)
})
</script>
```

clientX : using this property of event object we can get x position of mouse

clientY : using this property we can get y position of mouse

Ex:

```
<script>
function fun1(e){
document.getElementById("div1").innerText=" X pos is "+e.clientX+" Y pos is:"+e.clientY
}
</script>
<body onmousemove="fun1(event)">
<div id="div1"></div>
</body>
```

//////////////////////

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

type : using this property we can get the type of event which is executed.

```
<script>
function fun1(e){
alert(e.type)
}
</script>
<body onclick="fun1(event)">
<div id="div1"></div>
</body>
```

//////////////////

button : using this property we can get the type of button which is clicked by mouse.

If user clicked on left button value is 0 if center value is 1 if right button value is

2

onmousedown event executes for left,center and right buttons.

```
<body >
<img src="mango.jpg" width="100px" onmousedown="fun1(event)"/>
</body>
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
<script>
function fun1(e){
if(e.button == 2){
alert("Can not copy image")
}
}
</script>
```

key : using this property we can get the key information pressed by user.

onkeydown

onkeyup

onkeypress

Ex:

```
<script>
function fun1(e){
alert(e.key)
}
</script>
<body>
 <input type="text" onkeyup="fun1(event)"/>
</body>
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

keyCode : using this property we can get keycode value. keycode is same as ascii but

it is common for both capital and small alphabets.

A - 65

B - 66

C - 67

Z - 90

a-97

b-98

z-122

0-48

1-49

9-57

```
<script>
function fun1(e){
alert(e.keyCode)
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
}
</script>
<body>
 <input type="text" onkeyup="fun1(event)"/>
</body>
```

preventDefault() : using this function we can stop the functionality of current event

```
<script>
function fun1(e){
if(e.keyCode < 48 || e.keyCode > 57){
alert("Invalid value")
e.preventDefault()
}
}
</script>
<body>
 <input type="text" onkeydown="fun1(event)"/>
</body>
```

Event Trickling/Bubbling : It is a concept of execute parent element event we we perform

an action on child element.

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

Trickling is the concept of calling events from up to down.(parent to child)

Bubbling is the concept of calling events from down to up(child to parent)

For every event 2 flows execute and calling the functions in the flow which we set.

First flow is tricking and next is bubbling

By default events are set in bubbling flow

---

stopPropagation() : using this function we can stop the flow of events in trickling/bubbling chain

```
<body>
  <div id="container" class="div1">
    <img src="close.png" width="20px" style="float:right;margin-top:10px;margin-right:10px" id="imgClose"/>
  </div>
</body>
<script>
</script>
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
<style>
  .div1{
  width:300px;
  height:300px;
  background-color:yellow;
  border-radius:20px;
  position:absolute;
  top:150px;
  left:-250px;
  }
</style>
<script>
document.getElementById("container").addEventListener("click",function(){
        document.getElementById("container").style.left="-20px"
})
document.getElementById("imgClose").addEventListener("click",function(e){
        document.getElementById("container").style.left="-250px"
        e.stopPropagation()
})
</script>
```

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

call() : Using this function we can pass an object into the function of another object as owner of that function

```
<script>
var obj={
        uname:"scott",
        city:"hyd",
        fun1:function(x,y){
                alert(this.uname)
                alert(this.city)
                alert(x)
                alert(y)
                }
        }
        var newobj={
                uname:"amith",
                city:"chennai"
                }

        obj.fun1.call(newobj,10,20)
</script>
```

apply() : It is same as call but second argument should be array

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
<script>
var obj={
        uname:"scott",
        city:"hyd",
        fun1:function(x,y){
                alert(this.uname)
                alert(this.city)
                alert(x)
                alert(y)
                }
        }
        var newobj={
                uname:"amith",
                city:"chennai"
                }

        obj.fun1.apply(newobj,[10,20])
</script>
```

bind : It is same as call, but it returns a function instead of execute it.

```
<script>
var obj={
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
        uname:"scott",
        city:"hyd",
        fun1:function(x,y){
                alert(this.uname)
                alert(this.city)
                alert(x)
                alert(y)
                }
        }
        var newobj={
                uname:"amith",
                city:"chennai"
                }

        var rv=obj.fun1.bind(newobj,10,20)
        rv()
</script>
```

Ex:

```
<body>
<img src="all.jpg" width="100px" onclick="fun1()"/>
<br />
```


DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

```
<input type="button" value="Click" id="but1"/>
</body>

<script>
function fun1(){
        alert("Fun1 exec...")
        document.getElementById("but1").onclick=fun2.bind({},10,20)
        }
function fun2(x,y){
        alert("Fun2 exec...")
        alert(x+y)
        }
</script>
```

React : React is library of Javascript to implement dynamic web pages. React functionslities are available in the form of node packages.

Basic required packages are

react

react-dom

react-scripts

react : This package provides all the functionalities of react.

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

react-scripts : This package provides commands to run,build and test react project

react-dom : Creates dom object based on the controls of react component

----

Repository : It is a location provides packages of various technologies like react,angular,node,....

Available repositories are node repository, yarn repository....

node repository : It is a location provides the packages of various technologies like react,angular,....

we can download the required packages from node repository using npm program.

npm install react

npm install react-dom

npm install react-scripts

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

npm (node package manager) : It is a program to install required packages from node repository. We need to install this program using node installation.

nodejs : It is server side environment of javascript. After installation of nodejs we can get npm program and node server.

node server we can use to run nodejs script

npm program we can use to download the required packages from node repository

----

public : index.html - root

component : It is collection of logics and design script. We have 2 types

1) functional component : If we create a component using function comes under functional component

2) class components : component with class comes under class component.

src : We can use this folder to maintain components of project. index.js is a file available to load the components of project. It is start-up javascript file of application

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

By default every new project provides a component with name app

----

index.css : It is stylesheet of entire application.

npm start : using this command we can run the project.

First it loads index.js and compiles the components

it starts server in port 3000 and loads index.html file

in index.html we have div with id root displays the content of component

----

export : using this keyword we can export a component from current location to another location

import : To import the component in to current file

Microsoft Visual Studio Code : It is an open source IDE to work with react,angular and other applications.

It provides many options like color codes, integrated terminals, error messages,....

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

Binding : Using this concept we can bind a property or function with html control.

Data Binding : In this concept we can bind data(variable) with html control

Event Binding : To bind an event with html control

we can use {} brackets to bind data/events with html controls.

---

bootstrap : It is framework of css provides many pre-defined classes. we need to install bootstrap using

npm install bootstrap

We can use bootstrap to apply pre-defined classes to the html elements. we can also create responsive web pages using bootstrap.

font-awesome : using this package we can display font-awesome icons on browser.

we need to install this package.

fa fa-home

fa fa-shopping-cart

fa fa-print

props : props is an object to read the properties of the component.

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

React functional component lifecycle hooks

React provides many hooks to execute some statements when the data of component is updated.

useState() : using this method we can update the state of component. Means we can re-render the view when the variable is updating....

useEffect() : This hook executes whenever the state data changes.

useReducer() : Usingthis hook we can update the view when data is modified. We can execute a function before updating the data.

useRef() : Using this lifecyclehook we can refer input control of current document. We can access a dom element using this function.

Routing : using this concept we can work with single page applications.

react-router-dom : using this package we can work with single page applications.

BrowserRouter : using this we can create router object.

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

Route : To create route path to component. path is a property to provide route path

Link : To provide link to the component. to is the property to specify path to the link

---

Outlet : using this class we can display the content of routing page

---

useParams() : Using this lifecycle hook we can read the parameters of router object. We can read query string information of url address.

Query string : It is some piece of data passing through url address from one component to another component.

Router Guard : using this concept we can provide restrictions to access the components from user.

useNavigate() : Using this hook we can redirect from one component to another component through script.

useLocation() : Using this hook we can read the state data of component.

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

Lazy Loading : using this concept we can load the componentes based on demand.

nodejs : It is server side environment of javascript. Nodejs we can use to create API services.

expressjs : It is framework of nodejs

----

node : using this program we can run nodejs file

nodemon : It is same as node command but it compiles the program every time when we save file.

nodejs file extension should be .js

require() : using this function we can load external package into current node program.

listen() : using this function we can specify the port number to run node server

use() : using this function we can use external package which is installed in current program.

API service : Stands for application programming interface. It is a type of method to share the data with different platforms.

Syntax :

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com

appref.http method(<service name>,callback function)

In callback function of API service we need pass 2 arguments one holds the request data another one holds the response data.

cors : stands for cross origin resource sharing to send the api data from one domain to another domain

DURGASOFT, # 202, 2nd Floor, HUDA Maitrivanam, Ameerpet, Hyderabad - 500038,
☎ 88 85 25 26 27, 72 07 21 24 27/28 | www.durgasoftonline.com
Maii: durgasoftonline@gmail.com