Operator : JavaScript provided lot of different type of operator which help to maths operations.

Arithmetic operator : +, -, *, /, %(modules)

```
var a=10;

var b=3;

var sum,sub,mul,div,mod;

 sum = 13

sub = 7

mul = 30

div = 3

mod = 1
```

conditional operator :

```
>

>=

<

<=

==

!=
```

Result can be true or false.

assignment operator

```
=

a=10;

sum = 10+30;
```

10=20; error

10==20 compare

a+b=c+d; error

a+b==c+d; compare

===

=== vs ==

== it check only value doesn't matter their data type

=== it check value as well as their data types.

typeof operator or functions : it is use to check the variable data type.

Increment and decrement

var a=10;

a=a+1;

a++; it will increment by 1

a=a-1;

a-- it will decrement by 1

conditional statement ie if statements

it is use to execute set of statement base upon condition.

1. normal if
   if(condition) {

   }
2. If else
   if(condition) {

   }else {

   }
3. If else if

   If(condition) {

   }else if(condition) {

   }else if(condition) {

   }else {

   }

If_statement.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <script type="text/JavaScript">
        // var a=10;
        // var b=50;
        // if(a>b){
        //         document.write(" Yes a is largest ")
        // }else {
        //     document.write(" Yes b is largest ")
        // }
        var m1=60;
        var m2=90
        var m3 = 87
        var total = m1+m2+m3;
```

```
        var avg = total/3;
        // if avg > 90 A++ else if avg > 80 A if avg > 65 B else C grade
        document.write("<br/> Finished...")
    </script>
</body>
</html>
```

Looping : looping is use to execute set of statement again and again till condition become false.

Initialization : start and end var i=1, n=10

Condition : must be true i<=10 or i>=10

Do the task

Increment or decrement i++ or n--

While loop initialization entry loop

while(condition) {

do the task

increment or decrement

}

Do while loop exit loop

initialization

do {

do the task

increment or decrement

}while(condition)

For loop

1st 2nd 4th

for(initialization;condition;increment/decrement) {

for body 3rd

}

1*1=1

1*2=2

1*3=3

Loop_demo.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <script type="text/JavaScript">
        // while loop
        // var i=1,n=10;
        // while(i<=n){
        //     document.write("<br/> i = "+i)
        //     i++;
        // }

         // do while loop
        //  var i=1,n=10;
        // do{
        //     document.write("<br/> i = "+i)
        //     i++;
        // }while(i<=n);
        // for loop

        // for(var i=0,n=10;i<=n;i++){
        //     document.write("<br/>"+i)
        // }
        // document.write("<ul type=circle>")

        // for(var i=0,n=20;i<=n;i++){
        //     document.write("<li>"+i+"</li>")
        // }

        // document.write("</ul>")

       document.write("<table border=1>")
        var table =4
        for(var start=1,end=10;start<=end;start++){
            document.write("<tr><td>"+(start*table)+"</td></tr>")
```

```
        }
        document.write("</table>")
    </script>
</body>
</html>
```

JavaScript functions

Function : function is use to write set of instruction to perform a specific task.

Using function we can do re-usability of the code.

Syntax

Normal Functions

function functionName(parameterList) {

body of the function

}

1. Function no passing parameter and no return type
2. Function passing parameter no return type
3. Function passing parameter and return type

Pre defined function

1. alert(“Msg”); it is use to display pop message
2. prompt(“”) it is use to take the value through keyboard.
3. eval() : it use to convert string to number.

pre-defined-fun.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <h1>Welcome</h1>
    <script type="text/JavaScript">
        document.write("Welcome")
        //alert("Welcome to JS");
        // var name = prompt("Enter your name")
        // alert("Welcome user "+name)
        var a=prompt("Enter the value of a")
```

```
        var b=prompt("Enter the value of b")
        var sum =eval(a)+eval(b);
        alert("sum is "+sum);
    </script>
</body>
</html>
```

User defined function or custom function

User-defined-fun.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <script type="text/JavaScript">
        function sayHello() {
                alert("Welcome to user defined function")
        }
        function add(a,b){
            var sum = a+b;
            alert("Sum in add function is "+sum);
        }
        function checkUser(emailid,password){
            if(emailid=="akash@gmail.com" && password=="123"){
                return "success";
            }else {
                return "failure";
            }
        }
    </script>
</head>
<body>
    <script type="text/JavaScript">
        // sayHello(); // no passing parameter and no return type
        // add(10,20); // function passing parameter.
        // add(100,200);
        // add(1,2);
        var emailid = prompt("Enter your emailid")
        var password = prompt("Enter your password")
```

```
        var result1 = checkUser(emailid,password);
        alert("Main script "+result1)
    </script>
</body>
</html>
```

JavaScript events

Event provide bridge between html and JavaScript code.

DOM elements (Document object model) any html tags.

All dom element generate different type of events.

Types of events

In JavaScript all event start with pre defined on followed by evet name.

Like

onClick	button click
onDblClick	
onMouseOver	
onMouseOut	
onKeyUp	
onKeyDown	textfield
onSubmit	
onChange	
onBlur	
onFocus	

if we want to generate the event we need to register that event on dom element or tags

base upon type of events.

Once the event generate we need to provide listener. In JavaScript we can use

Function which help to listen generated events. After capture we can do the task.

event-demo.html

<!DOCTYPE html>

```
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <script type="text/JavaScript">
        function sayHello() {
            alert("Welcome")
        }
    </script>
</head>
<body>
    <input type="button" value="Click Here"
    onclick="sayHello()"/>
</body>
</html>
```