**JavaScript:**

JavaScript was first known as Live Script, but Netscape changed its name to JavaScript can be implemented using JavaScript statements that are placed within the <script>... </script> HTML tags in a web page. You can place the <script> tags, containing your JavaScript, anywhere within your web page, but it is normally recommended that you should keep it within the <head> tags

E.g.

<script ...>

JavaScript code

</script>

The script tag takes two important attributes −

* Language
* Type

e.g.:

<script language = "javascript" type = "text/javascript">

**JavaScript code**

</script>

Program1:

<html>

<body>

<script language = "javascript" type = "text/javascript">

<!--

document. Write ("Hello World!") /\*Document.write is a keyword used to print the statement\*/

//-->

</script>

</body>

</html>

**Jscript used**: There is a flexibility given to include JavaScript code anywhere in an HTML document. However, the most preferred ways to include JavaScript in an HTML file are as follows −

* Script in <head>...</head> section.
* Script in <body>...</body> section.
* Script in <body>...</body> and <head>...</head> sections.
* Script in an external file and then include in <head>...</head> section

**Java script used in head section using Function:**

e.g:

<html>

<head>

<script language = "javascript" type = "text/javascript">

function hello()

{

alert ("Hello world")

}

</script>

</head>

<body>

<input type="Button" onclick="hello()" value="Button">

</body>

</body>

</html>

// **alert**- it is a set of key word.

**Hello-** is function name used in the program.

**onclick**- It is the event used to call the function used in program.

**Value**- It is set of word or context used inside the button icon.

**JavaScript Datatypes**:

Numbers, eg. 123, 120.50 etc.

Strings of text e.g. "This text string" etc.

Boolean e.g. true or false.

E.g.:

<script type = "text/javascript">

<!--

var money;

var name;

Or

var money, name;

//-->

Storing a value in a variable is called variable initialization.

**JavaScript Variable** **Scope**:

**Global Variables** − A global variable has global scope which means it can be defined anywhere in your JavaScript code.

**Local Variables** − A local variable will be visible only within a function where it is defined. Function parameters are always local to that function.

**Program 2**:

<html>

<body onload = checkscope();>

<script type = "text/javascript">

<!--

var myVar = "global"; // Declare a global variable

function checkscope( ) {

var myVar = "local"; // Declare a local variable

document.write(myVar);

}

//-->

</script>

</body>

</html>

**Onload**-onload is most often used within the <body> element to execute a script once a web page has completely loaded all content

**JavaScript - Operators**

Let us take a simple expression 4 + 5 is equal to 9. Here 4 and 5 are called operands and ‘+’ is called the operator

JavaScript supports the following types of operators.

* Arithmetic Operators
* Comparison Operators
* Logical (or Relational) Operators
* Assignment Operators
* Conditional (or ternary) Operators program

Program 3:

Live Demo

<html>

<body>

<script type = "text/javascript">

<!--

var a = 33;

var b = 10;

var c = "Test";

var linebreak = "<br />";

document.write("a + b = ");

result = a + b;

document.write(result);

document.write(linebreak); // Move to the next line.

document.write("a - b = ");

result = a - b;

document.write(result);

document.write(linebreak);

document.write("a / b = ");

result = a / b;

document.write(result);

document.write(linebreak);

document.write("a % b = ");

result = a % b;

document.write(result);

document.write(linebreak);

document.write("a + b + c = ");

result = a + b + c;

document.write(result);

document.write(linebreak);

a = ++a;

document.write("++a = ");

result = ++a;

document.write(result);

document.write(linebreak);

b = --b;

document.write("--b = ");

result = --b;

document.write(result);

document.write(linebreak);

//-->

</script>

Set the variables to different values and then try...

</body>

</html>