**Introduction to JavaScript:**

JavaScript is a dynamic programming language that's used for web development, in web applications, for game development, and lots more.

It allows you to implement dynamic features on web pages that cannot be done with only HTML and CSS

JavaScript is a popular programming language that has a wide range of applications.

JavaScript was previously used mainly for making web pages interactive such as form validation, animation, etc.

Nowadays,

JavaScript is also used in many other areas such as server-side development, mobile app development, and so on.

Because of its wide range of applications, you can run JavaScript in several ways:

\*1. Using console tab of web browsers

\*2. Using Node.js

\*3. By creating web pages

**JavaScript Can Change HTML Content**

One of many JavaScript HTML methods are:

1. **getElementById( ).**
2. **console.log( )**
3. **write( )**
4. **alert( )**
5. **prompt()**

**JavaScript Alerts – Adding behavior to websites**

**Using console tab of web browsers**

Open your favorite browser and right-click on an empty screen

And right-clicking on an empty area and selecting Inspect

On that page, you have to see live in that browser type "Console.log("write something");"

And now see our output displayed in the browser.

**Using Node.js**

Node is a back-end run-time environment for executing JavaScript code.

To run JS using Node.js, follow these steps:

Install the latest version of Node.js.

Install an IDE/Text Editor like Visual Studio Code.

In VS code, create a file > write JS code > save it with .js extension.

Open up the terminal/command prompt > navigate to the file location > type node hello.js >enter.

You will get output on the terminal.

**how to use JavaScript in HTML**

**1.Inline JavaScript**

<button onclick="alert('You just clicked a button')">Click me!</button>

**2.Internal JavaScript**, with the script tag

<script>

function(){

alert("I am inside a script tag")

}

</script>

**3.External JavaScript**

<script src="./script.js">alert("I am inside an external file");</script>

**Creating web pages:**

New File > Save it with sample.html extension.

For example,

<!DOCTYPE html>

<html lang="en">

<head>

<title>RCS</title>

</head>

<body>

**<script src="sample.js"></script>**

</body>

</html>

**JavaScript Can Change HTML Content**

One of many JavaScript HTML methods is

**getElementById().**

<script>

document.getElementById("demo").innerHTML = 5 + 6;

</script>

**console.log():-**

console.log(“hello”);

**write():-**

<script>

document.write(5 + 6);

</script>

**alert():-**

<script>

alert("pls enter the name");

</script>

**prompt():-**

<script>

prompt("pls enter the name");

</script>

**Statements in JavaScript**

A computer program is a list of “instructions” to be “executed” by a computer.

In a programming language, these programming instructions are called statements.

JavaScript program is a list of programming statements.

Semicolons (;)

Semicolons separate JavaScript statements.

var a;

a = 5;

**JavaScript Comments:-**

Code is easier to read and understand.

They are completely ignored.

There are two ways to add comments to code:

//- Single Line Comments

/\* content \*/ - Multi-line Comments

**JavaScript outline**

**JavaScript Variables**

* + - Var
    - Let
    - const

**Data types**

* + - String
    - Numbers
    - Boolean
    - Arrays
    - Objects
    - Undefined
    - Null

**Operators**

* + - Arthimatic
    - Assignment
    - Comparision
    - Logical
    - Bitwise

**Control statements:**

* + - If
    - else
    - if-else conditionals
    - switch

**Loops**

* + - while Loop
    - for Loop
    - do-while

**Functions**

* + - With parameter
    - Without parameter
    - return

**Types of Variables**

Variables are containers for storing data (values).

In JavaScript we have 3 types of variables

* + Var
  + Let
  + Const

|  |  |  |
| --- | --- | --- |
| **Var** | **Let** | **Const** |
| **var** can be reassigned | **let** can be reassigned | **const** can’t be reassigned |
| All type of browsers are not accept | All type of browsers are accept | All type of browsers are accept |
| **var** can be redeclared | **let** can’t be redeclared | **const** can’t be redeclared |
| Ex:-  Var a = 3;  Var a =9;  Console.log(a); | Ex:-  Let a=’rcs’;  Let a =7;  Console.log(a); | Ex:-  Const a=’rcs’;  Const a=3;  Console.log(a); |

**Variables**

**Variables are containers for storing data (values).**

example,

x, y, and z, are variables, declared with the var keyword:

* var a = 5; var b = 6; var c = a + b;
* Example:-
  + a stores the value 5
  + b stores the value 6
  + c stores the value 11

**JavaScript Identifiers**

* All JavaScript **variables** must be **identified** with **unique names**.
* These unique names are called **identifiers**.
* Identifiers can be short names (like x and y) or more descriptive names

(age, sum, totalVolume).

* The general rules for constructing names for variables (unique identifiers) are:
* Names can contain

letters, digits, underscores, and dollar signs.

* Names must begin with a letter
* Names are case sensitive

(y and Y are different variables)

* Reserved words

(like JavaScript keywords) cannot be used as names

**Variable Scope in JavaScript**

* Scope of a variable is the part of the program from which the variable may directly be accessible.
* In JavaScript, there are two types of scopes:
* **Global Scope** – Scope outside the outermost function
* **Local Scope** – Inside the function being executed.

**Variable let:**

Variables defined with let cannot be Redeclared.

Variables defined with let must be Declared before use.

Variables defined with let have Block Scope

let x = "John Doe";  
let x = 0;  
note: SyntaxError: 'x' has already been declared

**Variable Const:**Variables defined with const cannot be Redeclared.Variables defined with const have Block Scope.

Data types in JavaScript

Primitive types

String

Number

Boolean

null

Undefined

Composite types

Objects

Arrays

**String**

* strings are used for storing and manipulating text, Is used to store text represents
* A JavaScript string is zero or more characters written **inside quotes**.

Example:

let FirstName = "Nirosha";  // Double quotes  
 let LastName = "Rathod";  // Single quotes

**1. String**:- Is used to store text represents

\* Single quotes: ‘Hello',

\* Double quotes: “Hello !"

\* Backticks: `Hello`

const y = "Hello"; //“Hello" is a string data.

console.log(typeof(y)); //its checking the **type of datatype**

**Properties In String:-**

**Length():-**

<script>

let text = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";

document.getElementById("demo").innerHTML = text.length;

</script>

**2.Number:-**

Number represents integer and floating numbers (decimals and exponentials).

const number1 = 3;

const number2 = 3.433;

const x = 4; //4 is an integer data.

Console.log(typeof(a)); //its checking the type of datatype

**3.Boolean:-**

This data type represents logical entities.

Boolean represents one of two values: true or false.

It is easier to think of it as a yes/no switch.

\* true,

\* false

var a = 10, b = 20;

Console.log(Boolean(a>b));

ex:-

const data2 = true;

if (data2 == true ){

console.log("Yes");

}else {

console.log("No");

}

**Undefined:-**

data type represents value that is not assigned

\* let a;

\* console.log(a); // undefined

**null:**

null is a special value that represents empty or unknown value

\* let a = null;

Console.log(a);

**Array:-**

* Arrays are the collection of similar types of data stored at contiguous memory locations.
* It is the simplest data structure where the data element can be accessed randomly just by using its index number.
* An array is a special variable, which can be hold multiple values:
* A ={8,6,8,9,2,99}

8 6 8 9 2 99

Arrays are those data structures that span across one dimension.

const car = ["Saab", "Volvo", "BMW"]; //single dimensional array

console.log(car);

let arr=[5,6,7,8,9,1,2,3,4];

console.log(arr[5]);

arr[4] =0;

console.log(arr[4]);

console.log(arr);

* Arrays are those data structures that span across more than one dimension.

const data = [[1, 2, 3], [1, 3, 4], [4, 5, 6]]; //multidimensional array

console.log(data);

**ex:-**

const data3 = [name: "rcs", index: 5];

if (data3.name == "rcs"){

console.log("Yes");

}else {

console.log("No");

}

**SessionPrograms:**

let a = 20;

let b = 10;

let result = a + b;

console.log("result is :" + result);

var a = 3;

var b = 10;

console.log("a value is : " + a + " \n " + "b value is :" + b);

a=19;

console.log(a);

var a=20;

console.log(a);

let a=3;

a=19;

console.log(a);

let a=20;

console.log(a);

const a=3;

console.log(a);

a=19;

console.log(a);

const a =20 ;

console.log(a);

var a=30;

a++; //a=a+1 ->a= 30+1

console.log(a);

let b=20;

b++;

console.log(b);

const c=40;

console.log(c);

c++; //c=c+1 -> 40+1

console.log(c);

{

var a=30;

a++; //a=a+1 ->a= 30+1

console.log(a);

}

console.log(a);

{

let b=20;

b++;

console.log(b);

}

console.log(b);

let e=89;

{

const c=40;

console.log(c);

let d=30;

console.log(d);

}

console.log(e);

let names = "Vijaya";

console.log("My name is :" + names);

let msg ="Hello"

msg += " Vijaya!!";

console.log(msg);

let firstname="Vijaya";

let lastname="p";

let fullname = `${firstname} ${lastname}`;

console.log(fullname);

let a=0;

let b=22.7;

let c=7878847348.00;

let d=535332.999;

console.log(a);

console.log(b);

console.log(c);

console.log(d);

let age=17;

if(age>=18){

console.log("he is elible for voting");

}

else{

console.log("he not is elible for voting");

}

let g="ff";

console.log(g);

let arr=[5,6,7,8,9,1,2,3,4];

console.log(arr[5]);

arr[4] =0;

console.log(arr[4]);

console.log(arr);