JavaScript

* JS => Hight Level Programming Language, interpreted Language(line by line Execution) and Dynamic Content.
* It runs inside the browser engines called V8 Engine
* Firefox=> Spider Money Engine.
* Safari => Nitro
* Math and Functional =>20%
* DOM=>50%

Variables:

* Named Container and holds the data in memory (Temporary)
* Var, let, const =>Keywords
* Phase1=> Declaration Phase
* Phase2 => Execution Phase

Declaration Phase:

* Variable Keywords variable name => Memory allocation Creation.
* Var => old ES1-ES5
* Let =>ES6
* Const =>ES6 => No declaration

Camel Case:

One variable First word first letter small and Second word capital letter.

Types of JS:

1. Internal JS => Using Tag
2. External JS => separate script file

Printing Statement:

Console.log()

Document.write()

Alert()

Primitive Data Types:

1. String
2. Number
3. Boolean
4. Undefined
5. Null
6. Big int
7. Symbol

Non-Primitive Data Types:

1. Array
2. Object
3. Array of object
4. Function

Operators:

1. Arithmetic Operator => +,-,/,%,\*
2. Assignment Operator => =,==,===,+=,-=
3. Comparison Operator => ==,<=,>=,<,>,===,!=
4. Logical Operator => &&, ||, !

Redeclare and Reassign:

VAR:

* We can do declaration
* We can do Re-declare
* We can do Re-assign
* Global and Functional Scope

LET:

* We can do declaration
* No re-declare
* No re-assign
* Global and block Scope

CONST:

* NO Declaration
* No re-declare
* No re-Assign
* Block and global Scope

Conditional Statement:

* If-Condition => Used to check whether the statement or expression is true or false.
* If true execute one block of code
* Expression=> Valid piece of code inside the block and produce the values.

If -else Syntax:

If(condition){

}

Else{}.

Truthy Value:

1. Any thing have a string
2. Without O and others are true.
3. Empty Array[]
4. Empty Object
5. Function
6. True and infinite Numbers.

Flasy Value:

1. Zero Number
2. Negative Zero
3. Big int Zero
4. Empty String or single quotation
5. Null
6. No value assign, undefined
7. Not a number

Keywords:

* Typeof => it is used to find the data type

Looping Statement:

* Loop is used to repeat a block of code multiple times.
* Until a certain condition becomes false.
* Automatic Repeat Task
* Used for mostly in Array and used for calculation.

TYPES:

1. For-Loop
2. While Loop
3. Do-While

* For-Loop Syntax:

For (initialization, condition, Increment/Decrement) {

}

* While-Loop Syntax:

While(condition) {

Increment/decrement

}

* Do-While Syntax:

Do {

Increment/decrement

}

While(Condition){

}

Non-Primitive Data Types:

1. Array
2. Object
3. Array of object
4. Function

Array =>Collection of data types or values stored in single variable.

* You want to access the array values must know the index
* Starts from 0 Index
* One dimensional Array
* Two Dimensional Array
* Multi Dimensional Array

Object => collection of structured data and it must have a key value pair

* If we use object we have to give classification of own data.
* We can easily access the data using keys or values

Array of Object => Collection of objects, Structured collection of data.

Function:

1. Normal Function Declaration:

SYNTAX:

Function name () {

}

* Function declaration or definition
* Function stored in memory in creation page.
* Function invoked=> Called and Start Execution
* Returns values give back to call

1. Anonymous Function: => function without Name.

* Stored in variable

SYNTAX:

Const add =function(){

}

1. Arrow Function => Both Function and Anonymous Function.

SYNTAX:

Const add=(()=>{

})

1. Immediate invoked Function Expression:

* Function runs as soon as it created automatically.

Parameters and Arguments => Actual Values of part when calling the function.

SYNTAX:

Const add=(a,b)=>{

Console.log(a)

Console.log(b)

}

Array Looping => Using length only we can do Looping

SYNTAX:

For(let index=0, index<array.length; index++) {

Console.log(index) }

Array Methods Prototypes:

* Every Array in JS is created using the array prototype.
* Array Prototype gives access to built in methods.
* Built in function => Already write a function that performs specific task based on arguments.

Repeated Used Prototype:

1. Push() => Add elements in the last
2. Pop() => Remove the element in the last
3. Unshift() => Add element in beginning
4. Shift() => Remove the element in beginning
5. Join() => Convert into String
6. Concat() => Add 2 or more Array
7. Includes() =>if check the values exist in the array wheather check true or false
8. Slice() => Piece of array slice(Start, End)
9. Splice() => Add or remove element in array splice(start, delete,new)
10. Index0f() =>check the index of a value, input will be normal number.
11. Valuesof() => Check the value and input will bw index value.