

# Javascript class # 1

- Javascript was invented by Brenden Eich in 1995.
- too it was developed for Netscape 2 and became the ECMA-262 Standard in 1997.
- European Computer manufacturers association Ecma International formerly European Computer manufacturer Association Organization that develops standards in Computer and technology.
- ES1 to ES9 (1997 to 2009)  
After that in 2015 major changes to follow the rules and regulation that is called Ecma script | ES2015 | ES6.
- ES6 is standard for Javascript after that every year new changes come. ES7, ES8, ES10.

Js is light weight oriented object oriented programming language.

- Use in form submits.
- in client side validation.
- PopUp / events on click uses.
- Client side execute | browser / Js query , Read js , angular js ,
- website server side (Nodejs , Expressjs)
- Mobile development | Hybrid App (framework for mobile app react native , phone gap etc)
- Software Development | Electronjs , Ex - VSCode , framework etc.
- we use <script> for Js  
<script src="index.js"></script> (for file link)
- we use alert for popUp.  
alert("Hello")

## Class 2 (Variable and datatypes).

- variable: Variable is just like a container.  
variable is used to store information.  
• It reserves space in memory. its  
data can vary but memory location will  
always remain same.

### Variable Rules:-

- variable name. Can't be any keyword. e.g  
(Alert, prompt) etc.
- Variable is ~~case sensitivity~~. Case sensitive.  
Same name in Capital and small letters are  
different. e.g Name or Name (both are 2 diff  
variables).
  - Variable can be consist of alphabet numbers,  
dollar sign and underscore.
  - Variable name can't be start with digit  
(no) in first letter.
  - no space allowed.

### As A Good programmer:-

your variable name should match with its  
concept.

- When you want 2 words js in variable name, so first was start with small letter and 2nd letter start with Capital.  
e.g fullName, rollNumber etc.  
 ↴ it is Camel Case.

### Types of variables.

- Var (Used before Ecmascript this type of variable can be define again in again in JS).
- After ESG in modern or advance JavaScript these 2 keywords use for declaration variables.
  - let:- (its value can be change any time in programming language and can declare and assign in 2 steps) e.g `let name = (de`
  - Const .- (it use for Constant value e.g pi its cant be changed . it value must

assigned at the time of declaration.

e.g Const name = "Alma"; (declare and assign in same sentence).

## • Variable Scope

1) Block scope Variable:- If variable declared in block of code, in curly bracket {}, it will alive only in block and will not be accessible after curly bracket.

2) Global scope Variable:-

These variable used Globally in whole program.

• Comments in JS:

Single line: // let name = "Alma";

multi line: /\* \*/

3) Print / Display in JS

On Browser:

window . document . write ("Alma");

In console:

console . log ("Alma");

• PopUp -  
window.alert ("Ama");

Taking Input from User in  
Prompt - in Javascript we use in the  
Prompt() function to ask the user for input

As a parameter, we input the text and  
want to display to the user. Once  
the user presses "Ok" the input  
value is returned. We typically  
store user input in variable so  
that we can use the information  
in our program.

Let answer = prompt

## class # 3 Data types Primitive and Non-primitive

### primitive Data type.

- 1) Number 2) String 3) Boolean 4) Undefined
- 5) Object (Null)

i) Numbers :- aisa data jo no mein wrte kia jaye use no data type kehlte He.

e.g. let rollNo = 23;

2) Strings: whose data written in text called string.

e.g let name = "Aman"

3) Boolean:- koi aisa data jise hm true ya false me wrte krya ya jis ki value true ya false me ho boolean data type kehly He.

e.g let isPass = true;  
document.wrte (isPass);

4) Undefined :- jis mein koi value defined no ho is Undefined data type kehly He.

let class;

let person;

## Null Object:-

e.g let abc = null

document.write(null);

## Non-Primitive | Composite | Data types

### i) Array.

- Store multiple value in single variable.
- Value written in square bracket [ ]

e.g

let info = [5, "hina", "Asma"],

console.log(info);

## Object:-

- Store multiple value in single value
- Values written in curly brackets {} in pairs with keys.

e.g

let ~~student~~ student = { name: "Asma",  
roll no: 63,  
Sub: "Computer" }

## class 4 "Operators"

operators:- perform some task called operators

e.g  $a+b$ ,  $4+5$   
Expression      Operator

### i) Arithmetic Operation (Addition)

Let  $a = 9$

$b = 5$

`document.write(a+b)`      (direct point)

### 2) Subtraction

let  $a = 10$

$b = 4$

`document.write(a-b)`

### 3) Multiplication

`document.write(a * b)`

### 4) Division

`document.write(a / b)`

ج. Post mortem - بَلْيَاتِي بَعْدِ الْمَوْتِ  
Or Pre increment or deceased  
ج. Pre mortem - بَلْيَاتِي

# (Modular) Remainders

```
document.write(a, "b")
```

document.write(a + b)

## (Exponentiation)

let a=5

let b = 4

document.write( $a^{**b}$ ),

## Unity Operators

post increment a++

اس کا مطلب ہے  $1 + \text{مود}$

post decrement a --

اے نامیں جو ۱-نمرہ

pre increment ++A

pre-decement -- a

(Post increment) / Post decrement

Uranium Operated

$a++$ ,  $a = a + 1$

$$a = \dots, a = c_1 - 1$$

b--

document.write(b)

(pos decrement) - first

document.write(--a);

document.write(a);

(pos increment) + first

document.write(++a);

document.write(a);

## 2) Assignment Operators / Assign Value

let a=8

a+=4

document.write(a); 12

a-=3

document.write(a); 5

a\*=3

document.write(a); 24

=

+ = = +

- = = -

\* =

/ =

%

a/=3

document.write(a) = 2.66

a\*\*=3 (power)

1 ... 1-6. 18x3+8

Compare two values  
result in boolean

## Class 5 Comparison Operators (Conditional Operator)

اُس فارزک میتواند (false) یا (true) باشد

e.g. let  $a = 5$

-  $b =$  ?

$==$  (equal to)       $==$  (equal to + same data type)

$!=$  (not equal to)       $\neq$  (not equal to & different data type)

$>$        $<$        $>=$        $\leq$

let  $a = 2$ ; (number)

let  $b = "2"$ ; (string)

$a == b$  (True)       $a != b$  (False)

$a == b$  (False)       $a != b$  (True)

## Logical Operators (Comparison b/w two values)

Logical AND  $\&$

Logical OR  $\|$

Logical NOT  $!$

## Conditional Operators

if statement

if - else statement

if - else if statement

## Class 6 String

- string is a sequence of characters used to represent a text.
- It is primitive data type.
- we can create string by using template literal and in single and double quotation.

String creation and manipulation:-

```
let str = "I am learning JS"; double  
= ' i am learning JS'; single  
let str = ' i am learning JS'; // template  
literal (adjacent to 1 key in key)
```

template literal = ' ' back ticks

Template literal:-

Template literal are a feature in Java that were introduced in FSG. They give you a more flexible and maintainable way of working with string in Java.

- How to Use template literal in Java

- for tab (space) it
  - for print 1 in string
  - for write variable in string (variable name).
- ns. For double quotation hello | "Hello"  
let str = "Hello In How are you?"  
let num = 9  
document.write ("Hello \$(num)");

## Some String Properties and method.

- let str = " i am learning JS";  
" " = " CSS"  
" " = " HTML"
- position start in 0 in string
- To find length (str.length)
- To join string  
document.write (str1 + " " + str2)
- by concat {}  
let str4 = str1.concat(str2) / str2.concat(str3)  
document.write (str4)

length

let str = "hello"; how are you.

let l = str.length

document.write(l); length

Concat لطفاً این کار را در میان فایل های اخیر انجام دهید

let str2 = "I am learning JS";

document.write(str1 + str2)

trim

str.trim() // to remove spare start and end.

str.trimStart()

str.trimEnd()

str.toUpperCase() Change in upper case

str.toLowerCase() Change in lower case

str.replace("Java", "CSS")

String manipulation

\t (for space)

\\ (for quotation)

\" (for Single quotation)

In for next line

## String manipulation loops.

- To execute a piece of code again and again
  - Finite loop and infinite loop
  - Finite loop (ending point)
- infinite (not end) memory full / computer hang

for loop

```
for(let i=1, i<=5, i++)
```

```
{
```

```
document.write("Hello");
```

```
}
```

i is block scope variable.

use for iteration / counting.

• first step initialization

Condition true block of code

execute -

• 3rd step updatons

```
for(let i=1, i<=5, i++)
```

```
{
```

```
document.write(i, <br>)
```

```
let tabno = (from let tabno = parseInt(prompt("Enter your choice")))
```

~~for~~ for i=1, i<=12, i++

```
document.write(`$ ${tabno} x ${i} + ${tabno * i}`);
```

```
document.write("<br>");
```

## Class 9 (Array methods and for of loop)

Array :- (Store multiple value in single variable)

- Value written in square bracket [ ].

Values separated by Comma , .

each position is called index .

index no start with 0.

```
let arry = ["Hina", "Kazachi", "Islamabad"]
```

document.write (arry.length) Find length of array

```
for (let i=0, i<=1, i++)
```

{

```
    document.write (arry[i], "<br>") .
```

for of loop

```
for (let value of arry)
```

{

```
    document.write (value);
```

}

```
arry.push (html) (Insert item in end)
```

```
document.write (arry)
```

```
arry.unshift (item) (add item in start)
```

```
arry.shift (remove item in start).
```

```
arry.pop (remove item in end).
```

## class 9 (Array Methods and for of loop)

arr.tostring (Concat two array)

arr.indexOf (give index no)

~~arr.slice~~

arr.slice (Is mein Jo index no degy woh Se start hoker end take dega)

last 10 (for in loop / while loop do while loop)

Object:-

- Store multiple value in single variable
- Value written in curly brackets {} in pairs with  
let student = {  
 name: "Arma"

roll no: 25

class: Chemistry

}

for (let key in student)

{

document.write(key) (key": student[key], "<br>

}

for of loop

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

for (let i of arr)

{

document.write(i, "<br>")

}

while loop?

let i = 1

while (i <= 10)

{

  document.write(i, "Alma")

  i++

}

do while loop?

let i = 20

do

{ document.write(i, "Alma") }

  i++

} while (i <= 10); false

## Parithesh

### class 10 functions

- A Javascript function is a block of code designed to perform a particular task.
- A javascript function is executed when "Something" invokes it (call).

```
function abc(){  
    document.write("Alma");  
}  
abc();
```

function  
definition(define  
    function fname()  
        {  
            block of codes  
        }  
    }

function invoke  
(call)  
• fname()