

## Introduction to Javascript ES 6

24/09/2023

### Javascript :-

- The language that powers the web.
- There was a browser called Mosaic in the beginning.
- Netscape navigator dominated most of the website in ~~earlier~~ earlier days [25-years] ago.
- Netscape is evolved into Modern day firefox.
- Brenda Eich created java script in just 10 days.
- Live script → Jscript [Microsoft] → ECMAScript [European]
- Java script.
- ES stands for ECMA Script

### Script :-

(After Gwyneth appears on stage for 1 second  
Brad appears.)

Gwyneth : "Hello".

(Delay 1s)

Brad : "World".

↑ This is the working of script in stage.

Gwyneth & Brad ~~are~~ are actors.

when it comes to javascript Gwyneth and Brad are known as elements. which is related to `<hi></hi>`

`<p></p>`

Example:-

```
unhide 'hi'  
is delay  
unhide 'p'  
hi change text to "Hello"  
p change text to "world"
```

→ Java is different from Java script.

→ Java is a compiler whereas Java script is a interpreter.

- JS supports all of the major browsers.
- Chrome → view → developer options → source tab, → New snippet → "to edit and play as like".
- Explore using MDN website for Javascript.

### Syntax:

```
alert ( "Hello" );
function do something {
  Message end.
```

### Data types:

String → "level"

numbers → 1 2 3

integers

Boolean → True / False.

⇒ `typeof (2)` ⇒ gives the data type of the data  
Output ⇒ "number".

### Javascript Variables:

→ store something.

Eg:-

```
var a = prompt("What is your name ?");
    ↓
    ↗ variable
    ↗ Name
```

Keyword

→ 'var' says the computer to construct a new box to save something.

→ 'a' → variable name assigns Name to a box.

→ everything after the '=' is stored inside the box.

→ We can change the content in variables while we are changing, we don't need that Keyword.

Example :-

```
var gameLevel = 1;  
gameLevel = 2;  
gameLevel = 3;  
alert(gameLevel);
```

- give your variables a Meaningful Name.
- don't keep Keyword as variable Name, it is reserved.
- don't start your variables by numbers but we can add numbers to variable names.
- don't use space between variable names
- Variable Name contains only alphabets, numbers, - and \$ symbols, not any other.
- use camelCasing → camelCasing.

String Concatenation:-

→ we can combine strings  $\Rightarrow "a" + "b" = "ab"$   
 $"a" + " " + "b" \Rightarrow "a b"$

length:-

→ we can easily calculate the number of characters in a variable using length function.

length ( )

$\Rightarrow$  Variable-name.length.

Slicing & Extracting part of String:-

Slice (x, y)

Example :-

var name = "arwin"  
name.slice(0, 5)  $\rightarrow$  to 1  
                     $\hookrightarrow$  from 0

a r w i n  
0 1 2 3 4  
-5 -4 -3 -2 -1

Output :-

"a"

toUpperCase() :-

→ changes any word to uppercase.

Eg:-

```
var name = 'arwin';
name = name.toUpperCase();
```

toLowerCase() :-

→ changes any character to Lowercase.

Eg:-

```
var name = "A swIN";
name = name.toLowerCase();
```

Basic arithmetic & Modulo operator in Javascript:

Numbers:-

→ straight forward.

Eg:-

```
var add = 2 + 3;
var sub = 3 - 2;
var mul = 5 * 2;
var div = 6 / 2;
```

Modulo:-

```
var Mod = 6 % 4;
```

→ used Pemdas rule for large expression.

Example challenge:-

Dog age to Human age formula:-

$$\text{HumanAge} = (\text{Dog Age} - 2) \times 4 + 21$$

solution.

```
var dogAge = prompt("dog age?");
```

```
var humanAge = ((dogAge - 2) * 4) + 21;
```

```
alert(humanAge);
```

## Increment & Decrement Expressions:-

Example :-

$$\boxed{\text{Var } x = 5;}$$
$$x = x + 1;$$

instead

$\times$ 

+	+
---	---

 ;

increment  
expression

if  $x \rightarrow$  decrement expression.

`++` increment the value by 1.

$\rightarrow \text{--}$  decrement the value by 1.

$\rightarrow ' + = n'$  @ increment the value by  $n'$ .

$\rightarrow \text{`-}' = n'$  decrement the value by 'n'.

Functions :-

→ a set of instructions.

→ Syntax :-

function getMilk () {  
 a set of codes;  
}  
→ Keyword

→ To call a function just by function name.

Proper syntax:-

~~syntax~~  
~~gross~~ // creating a function.

```
// Creating a function  
function functionName () {
```

```
    alert ("This is a odd function");
```

```
var a = prompt("a=?");
```

```
var b = prompt("b = ?");
```

`alert(a + b);`

3 // calling a function

functionName();

set of  
code

`→ "console.log('...');"` displays something in console which is not seen by user.

## Functions with parameters & Arguments :-

→ functions which takes inputs:-

⇒ Example:-

```
function getMilk(bottles) {
```

```
    var cost = bottles * 10;
```

```
}
```

```
getMilk(2);
```

Here the output will be based on the input what we are giving.

```
if bottles=1 => O/P => 10,
```

```
" " =2 => O/P => 20,
```

## Functions with Outputs & Return Values:-

```
function getMilk(b) {
```

```
//do something.
```

```
var remaining = b % 10;
```

```
[return] remaining ;
```

```
y
```

→ Now when we call the function, it will output that says the remaining cost.

Example:-

```
function getMilk(bottles, rupees) {
```

```
    var cost = bottles * 10 ;
```

```
    var remaining = rupees % 10 ;
```

```
    return remaining ;
```

```
3
```

```
getMilk(2, 25);
```

Output:-

5