

# Let's Learn JavaScript

- ① What is JavaScript? ✓
- ② Java v/s JavaScript. ✓
- ③ The story of JavaScript. ✓
- ④ Applications of JavaScript. ✓
- ⑤ Installation of NodeJS & VS Code. ✓
- ⑥ JavaScript first Program. ✓
- ⑦ Introduction to Variables ✓

ECMA

## ⑧ Operators & Operands.

- Assignment Operators : = ✓
- Arithmetic Operators : +, -, \*, /, %, \*\*, ++, -- ✓
- Relational Operators : ==, ===, !=, !==, >, <, >=, <= ✓
- Logical Operators : &&, ||, ! ✓
- Bitwise Operators: &, |, 1, ~, <<, >> ✓

## ⑨ Data types

- String ✓ → Objects
- Number ✓ → Arrays
- Boolean ✓
- null ✓
- undefined ✓
- Symbol ✓
- BigInt 'n' ✓

## What is JavaScript?

- Programming language.  
↓ Why?
- Communicate with Computers  
↓ Why?

Framework  
What? Why? How?

2 mins → < 1 sec

- ① To do tasks
  - faster
  - error-free
  - multiple
  - repetition

Java v/s JS.

↳ Programming language

Car X Carpet

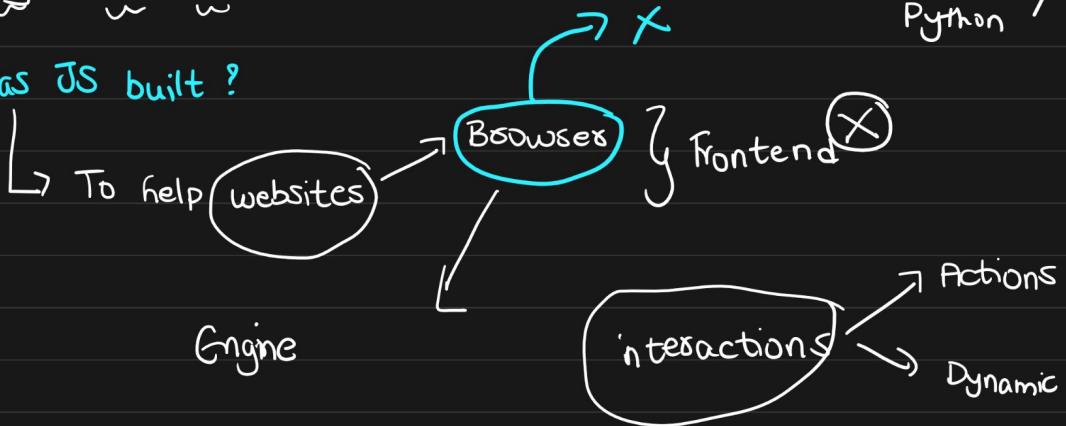
Programming language

Java X JavaScript

Car X Carpet

The Story of JS

① Why was JS built?



\* In the beginning only Browsers used to understand JS.

↳ engine

V8 Chrome

client-side JS

Full Stack

Frontend

Backend

Java

Python

server-side

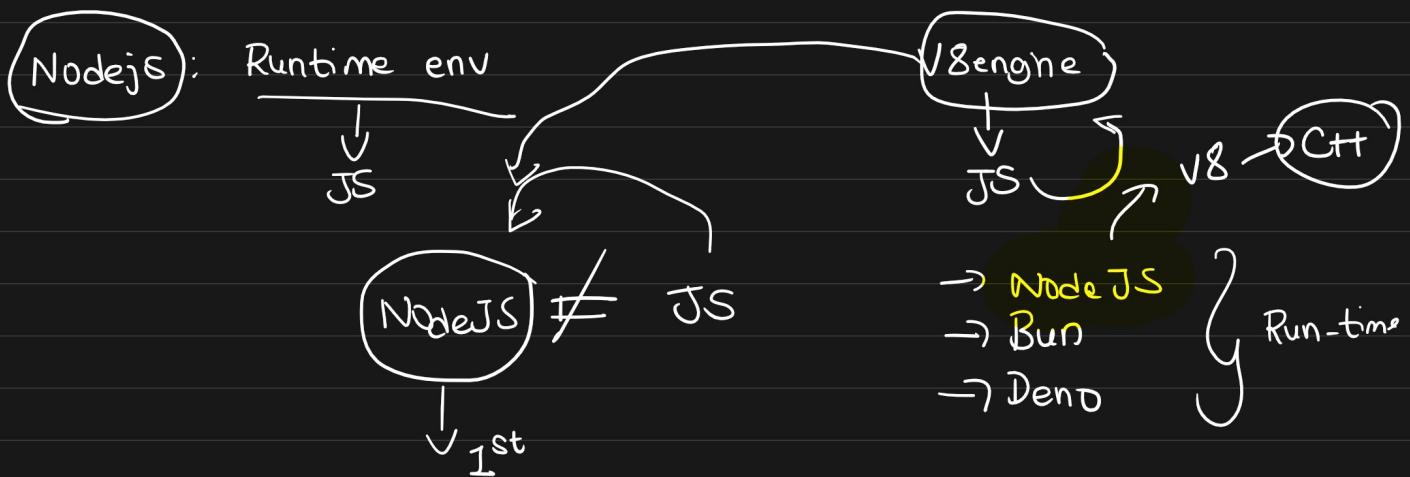
\* Frontend : Popular

V8 : Chrome



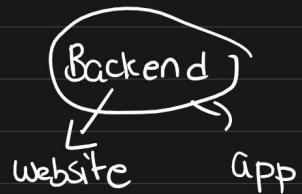
Google

↳ v8 → server → Node JS.



## JavaScript Applications

- Web Apps.
- Server-side applications: backend.
- Mobile Apps
- Game development
- Extensions
- Desktop Apps.



NodeJS

LTS → long term Support

Current → latest version

terminal > node -v

## Variables



Storage container.



$$2 + 3 + 5$$

5

10

How to create  
Variable ?  
in JS

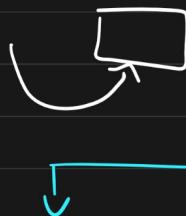
var      old  
let      latest  
const    Constants

why ?

Scope

## DataTypes

→ definition: what data is supported?



JS

Primitive

- ↳ direct memory
- ↳ immutable

Non-Primitive  
↳ ref → \*

↳ mutable

- string ✓
- Number ✓
- Boolean ✓
- undefined ✓
- null ✓
- Symbol
- BigInt

- Object ✓
- ↳ Array ✓

## String:

What? ✓ Why? ✓ How? ✓

→ set of characters.

→ representation of characters

- ↳ a-z
- ↳ A-Z
- ↳ -@-\$%
- ↳ Special char

Mithun said "JS is cool".

3 ways

- ‘ ‘ , , ✓
- “ “ ” ” ✓
- ‘ ‘ ‘ ‘ ✓

; → end of the line

JavaScript is a Dynamically-Typed language.

You are not expected to specify the datatype explicitly

typeof - Datatype.

Numbers → Infinity  
 Numbers → NaN → not a number.

→ Numerical values → To represent numeric values

- ↳ integers
- ↳ float/decimals
- ↳ Positive
- ↳ negative
- 0

Booleans

true or false

To help in decision-making

Object

Undefined    &    null  
empty → let discount = null.

→ i don't what is inside.  
↓  
let age = undefined

12

BigInt  
→ 16 position

9\_8\_7\_6\_5\_4\_3\_2\_1\_0\_1\_12

Symbol.  
↓  
→ unique value

String Interpolation. → ` ` \${Variable}

Object, Arrays.

- ↓
- Non-Primitive Datatype
  - key - value
  - key & value will be separated by :
  - each key-value pair will be separated by ,.
  - we create using {} .
- \* Memory  
\* Searching
- dot  
→ bracket

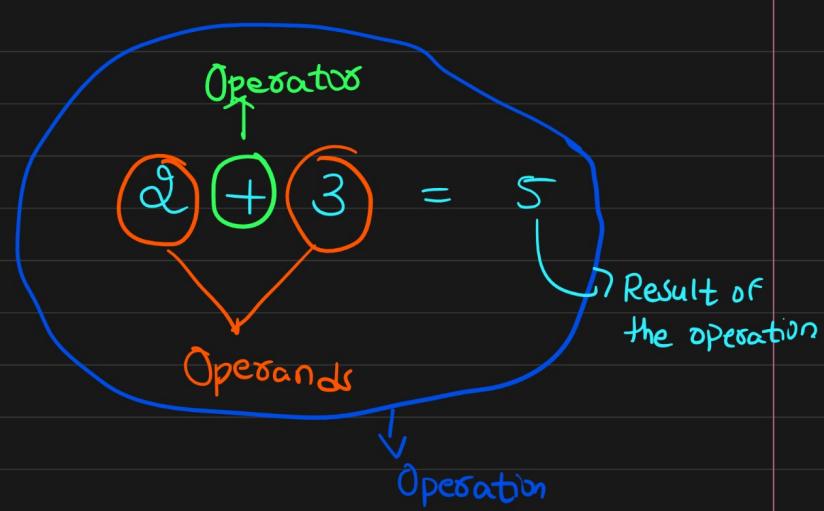
Arrays

- store multiple values of various datatypes
- each value is separated by comma.
- Index.

→ Indexing starts from 0 *\*HW*

## Operators

- Operation
  - Operators
  - Operands
- Maths



### ① Assignment Operators : "="

let num1 = 5  
 ↑  
 assign      ← Operator

### ② Arithmetic Operators.

- + → adds
- → subtract
- \* → multiply
- / → Result
- % → Remainder
- \*\* → power

++  
 --

2 to power 2 = 4

2 \*\* 2 = 4.

$$\begin{array}{r} 4 \\ \times 2 \\ \hline 4 \\ 0 \end{array}$$

Quotient  
Result  
→ Remainder

$$4 \div 2 = 2$$

$$4 \% 2 = 0$$

### ③ Relational Operators → Boolean.

→ It defines relation / Comparison

- ==
- !=
- ===
- !==
- >
- <
- >=
- <=

Comments  
 ↗ notes  
 //  
 ↗ single-line //  
 ↗ multi-line /\* \*/

## (4) Logical Operators $\rightarrow$ Truthy & falsy

AND ( && )

OR ( || )

NOT ( ! )

if both conditions are true the result will be true

if any 1 condition is true the result will be true

if the condition is true the result will be false.

## Bitwise Operators

$\rightarrow (a-z)$ ,  $\&$ ,  $\$$  : Bit notations  
 $a \Rightarrow$  0  
 $b \Rightarrow$  1  
 ASC II

$$1 + 2 = 3$$

Binary

1  $\rightarrow$  1

2  $\rightarrow$  10

3  $\rightarrow$  11

\$

|

&

!

1's

2's

10

11

2      \$      3.  
 Binary  
 ↓  
 10      ↓      Binary  
 11

= =  
 True

human = Tom  
 Dog = Tom

= = =  
 False

## Conditionals

- It is used to check for conditions
- Decision-making

→ Movie  
→ Class-end

① Write a program to check if number is even or not.

→ Even : Numbers which are completely divisible by 2.  
→ Odd :

→ remainder is 0

→ 0/0

Modulus = 0

```
let num = 5
if (num % 2 == 0) {
    console.log('The number is even');
} else {
    console.log('The number is odd');
}
```

if ( )  
↓  
Keywo .

)  
↳ condition → Boolean.

==

num % 2 == 0  
↓  
Boolean.

Edgecase ↗ 0

"0 is neither even nor odd".

## Ternary Operators

→ if-else ka easy way

if (condition 1 is true) {

execute this

} else {

execute this

}

Condition 2 ? execute this : execute this