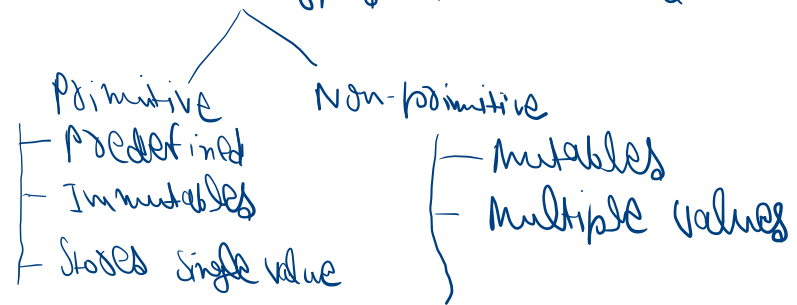


Datatypes & Variables



• Primitive (Numbers, Strings, boolean, symbol, Null, BigInt, Undefined)

in js (number) — 7-5 (Numbers)
float covers float, Integers

Strings - group of characters

boolean - { true / false }

Null - Intentionally want value to be absent

Undefined - when value is not assigned to variable but it is declared.

• Non primitives

Non-primitive

- objects (in detail)
- Arrays (respective functions)
- functions (same)

Everything inside JS is object
(Prototypes - objects)

Variable - Containers that store some data

→ Declaratives - let, const, var

Reassign	}	Redeclare
- var		- var
- let		

* const → only constant variable
can't be change or
reassigned.

Not-defined (any variable that is not declared yet)

Undefined - (variable declared but value is not assigned.)

→ console.log (type of undeclared var)

output - undefined

// // (undeclared var)

Reference error - (not defined)

Operators & Loops

ops are symbols & they are universally accepted

- we perform operations

- Arithmetic (+, -, *, /, %, ++, --)
- Assignment (=, +=, -=, *=, /=, %=)
- Comparison (==, !=, >, <, >=, <=)
- Logical (&&, ||, !)
- Ternary (?:)

• Arithmetic - mathematical operations.

Increment - increase the value by 1

Decrement - -1

• Assignment

• Comparison

Q → Conditional statements

→ If - else

→ Switch - Case

if (true) {

 sum
} → else if () {
else {
 sum
}

let marks = 25 >= A
 = 18 >= B
 = C

Loop \rightarrow (Repetitive task)

- while
- Do-while
- for
- for in (objects)
- for of (Arrays)

eg 1, 2, 3, 4 \rightarrow 10

~~loop~~ \rightarrow ~~for~~

\rightarrow for \Rightarrow (fixed no of repetitions or iterations)

\rightarrow while (certain condition to be true)

\rightarrow variable (control the loop)

\rightarrow In condition increment / decrement statement.

for (let i = 1; i <= 10; i++) {

 i = 10 - 1

 i = i + 1 = 0

 i++ 3--

Switch Case \Rightarrow when a single variable can have multiple

fixed values. egs = 40-50

let grade = A, B, C, D

It-OR / Switch Case \rightarrow