* Data types :

1. String — any kind of textual Information anything in double quotes

let firstName = "Anurag";

2. Number — Any mathematical real number

let a = 123;
let b = 12.45 ;
let c = 0.125 ;
let d = -6.23 ;

3. Boolean — true/false   let areYouDrunk = true;
let x = false;

4. undefined — no value / empty Box

* To know the data type of the value,

typeof value

eg: typeof 1234 → number
typeof "Anurag" → string
typeof year
↳ typeof 2001 → number

* typeof "2001" → string
typeof 2001 → number
typeof false → boolean
typeof "false" → string

✳ JS is a dynamically typed language,
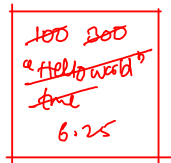
→ data types are for values,
not for variables

```
let   a = 100;
a = 300;
a = "Hello world";
a = true;
a = 6.25;
```

~~100 300~~
~~"Hello world"~~
~~true~~
6.25

a

⌗ other ways :

```
let name = "Anurag";

const year = 2001;

var Job = "teacher";
```

"Anrg"
name

2001
year

"Teach"
var

**✳ Const variables cannot
be changed, they are
immutable.**

✳ const day;
console·log (day);

→ Const cannot be undefined.

→ For now we say var and let are similar.
(discussed during functions).

**✳ var is old JS,
we will only use
let, const.**

★ Operators :

Arithmetic operators : +, -, *, /, **, %

* Ignore decimal part

    eg: 152 · 678

    op: 152

parseInt (152·678)

→ 152

* round off to 3 decimal points

let ar = 12·19823

ans. toFixed(3)

→ 12·199

* a % b = ans

range of ans [0, b-1]

---

① $a ** b \Rightarrow a^b$

$9^5 \Rightarrow 2 ** 5$

$8^{10} \Rightarrow 8 ** 10$

$\sqrt{16} \Rightarrow (16)^{1/2} \Rightarrow 16 ** (1/2)$

$\sqrt[3]{27} \Rightarrow (27)^{1/3} \Rightarrow 27 ** (1/3)$

② 11 % 3 ⇒

= 2

$$\begin{array}{r} 3 \\ 3\overline{)11} \\ -9 \\ \hline 2 \end{array}$$

% gives rem → rem

* odd % 2 , 13 % 2 = 1

even % 2 , 26 % 2 = 0

$X \% 3 = ans$

$1 \% 3 = 1$

$2 \% 3 = 2$

$3 \% 3 = 0$

$4 \% 3 = 1$

$5 \% 3 = 2$

$6 \% 3 = 0$

$7 \% 3 = 1$

$8 \% 3 = 2$

$9 \% 3 = 0$

range of

ans $= [0, 2]$

$\{0, 1, 2\}$

$$3 \overline{)1} \quad \begin{array}{c} 0 \\ \hline \end{array}$$
$$\underline{-0}$$
$$(1)$$

$$9 \overline{)2} \quad \begin{array}{c} 0 \\ \hline \end{array}$$
$$\underline{-0}$$
$$(2)$$

$$3 \overline{)3} \quad \begin{array}{c} 1 \\ \hline \end{array}$$
$$\underline{-3}$$
$$(0)$$

$$3 \overline{)4} \quad \begin{array}{c} 1 \\ \hline \end{array}$$
$$\underline{-3}$$
$$(1)$$

★ $a \% b = ans$

$ans = [0, b-1]$

$\{0, 1, 2, 3, \ldots, b-1\}$

eg: $a \% 4 = ans$

$ans = [0, 4-1] = [0, 3]$

$\{0, 1, 2, 3\}$

Addition with strings : (+) Joins / Concatenates two strings

```
const  firstName = "Ranbir"
const  lastName = "kapoor"
console.log ( firstName + lastName )
console.log ( firstName + " " + lastName );
```

"Ranbir" + "kapoor"
⇓
"Ranbirkapoor"

"Ranbir" + " " + "kapoor"

"Ranbir " + "kapoor"

"Ranbir_kapoor"

* Dynamic strings:

```
145  const name = "Soham";
146  const birthYear = 2002;
147  const job = "Mechanical Engineer";
148  const age = 2024 - birthYear;
149
150  console.log("Hi I am Anurag, 23 years old, working as a software engineer");
```

"Hi I am Soham, 22 years old, working as a Software engineer"

"Hi I am " + name + ", " + age + " years old, working as a " + job

↳ template literals

1. Identify dynamic parts change to $ { variable }

2. wrap in backticks (`)

```
`Hi I am ${name}, ${age} years old, working as a ${job}`
```

\# Assignment operators :   = , += , * = , /= , -= , % = , ** = , ++ , --

\# Comparision operators :   == , > , < , >= , <= , (!=)

not equals

* 22 < 22 → F   == → Comp

   22 <= 22 → T   = → assign

   23 > 23 → F

   23 >= 23 → T   * 22 != 22 → F

                    23 != 25 → T

* Comparision operator ↗ T
                       ↘ F