"" empty string

Short Circuiting

↓

applied to logical operators

↪ they only operate on boolean values.

first ← x (AND) y → second

x	y	
F	F	→ F
F	T	→ F
T	F	→ F
T	T	→ T

if you give non boolean then JS treats them as boolean by converting them.

→ In the and (&&) operator if the first operand is false, then it doesn't check 2nd operand & returns the first operand.

→ if the 1st operand is true, then it returns the 2nd operand.

truthy and falsy value in JS

→ those values which convert to true in the ToBoolean operation are truthy value , rest are falsy values.

falsy values → null, undefined, +0, -0, nan, "", false

↳ normal 0

Examples →

10 → true

100 → true

-0 → false

"Sanket" → true

"false" → true

10 && 30 → 30

true

logical operator

true

Because the first operand is true

automatically return 2nd operand

x	OR ||	y	
F		F	→ F
F		T	→ T
T		F	→ T
T		T	→ T

→ if the first operand is false then return the 2nd operand

→ if first operand is true then don't look on 2nd operand, return the first.

alway give boolean result ← Comparison Operator

↳ relational

10 < 20 → is 10 less than 20 ??

↳ true

10 > 20 → is 10 greater than 20 ??

↳ false

true ← 5 <= 6 → is 5 less than or equal to 6

8 >=10 → is 8 greater than or equal to 10.

a == b → abstract equality operator

↳ checks if a and b are equal or if we can convert a or b into some other type & make them equal

a === b → Strict equality operator

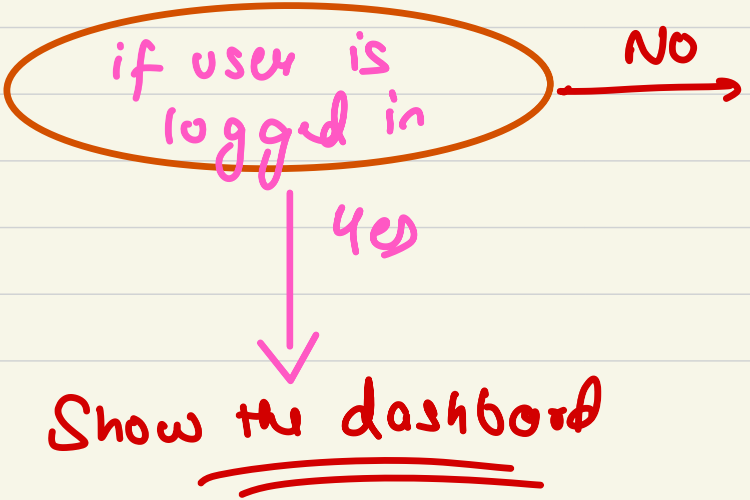
↳ it checks if a and b are equal and donot convert anything.

→ a != b → abstract not equals

a !== b → strict not equals.

Decision Making

in programming languages we can take decision to do a particular task based on conditions



Conditionals

↓

give JS decision making **capability**

(if, else, else if)

whenever you write an if JS automatically know a new conditional **statement** is gonna come.

↓

One valid JS **instruction**

evaluates to boolean

keyword

```
if ( condition ) {
        ____
        ____
        ____
}
else {

}
```

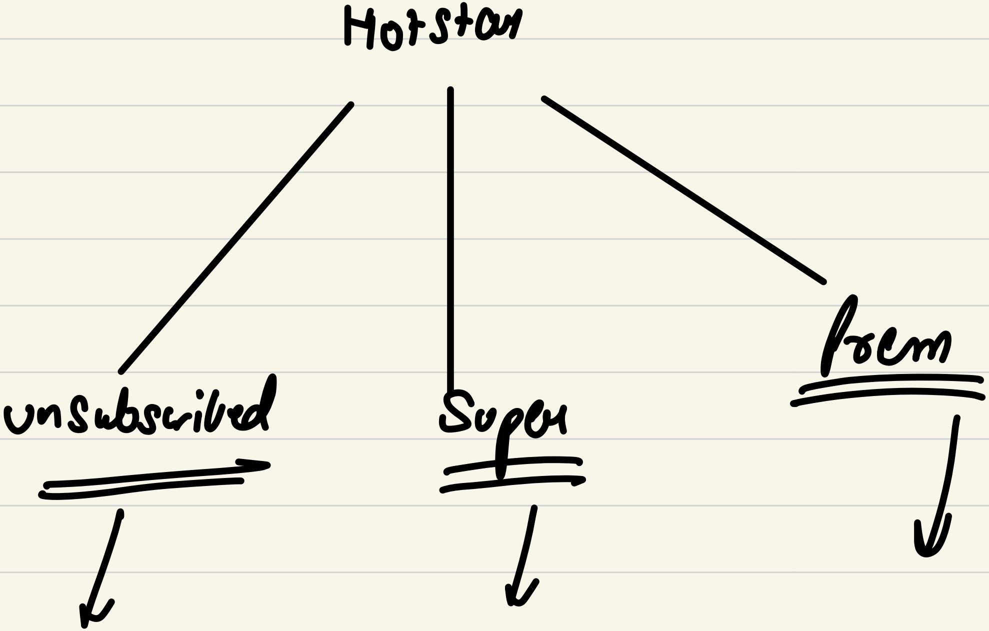
if block

block is a collection of statement

keyword

cannot exist without if

executes when condition is false


Hotstar
unsubscribed
Super
Prem

```
if (userIsUnsubscribed) {

} else if (userIsSuper) {

} else {

        →

}
```

Cannot exist without if

Can exist without else

```
if (c1) {

} else if (c2) {

} else if (c3) {

} else if (c4) {

}...
```

the moment any if or else if block executes remain blocks of the if, else, else-if conds

<u>execute</u>

nested conditional

```
if (userIsAPrimeCustomer) {
          if (userhasLimoCreditflag) {

          } else {
           }
}
```