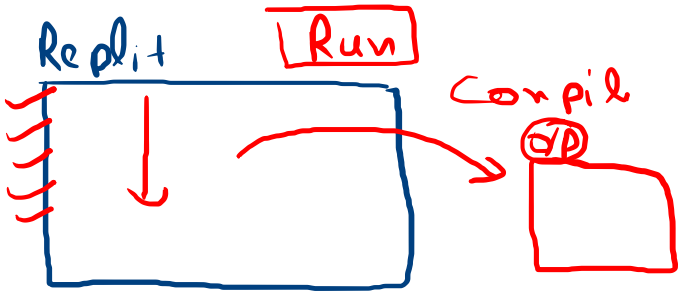
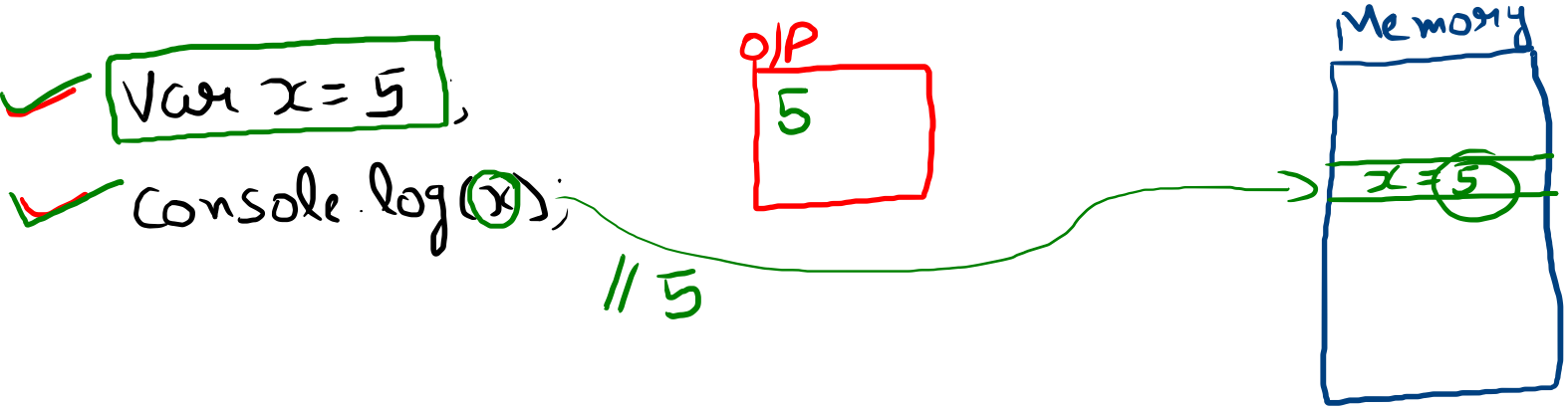
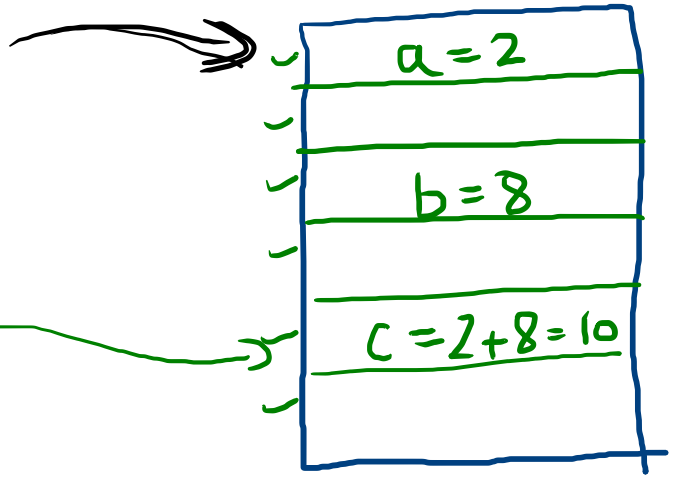
{Dry Run} →

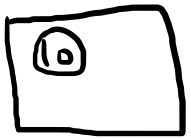
Manually Running the code



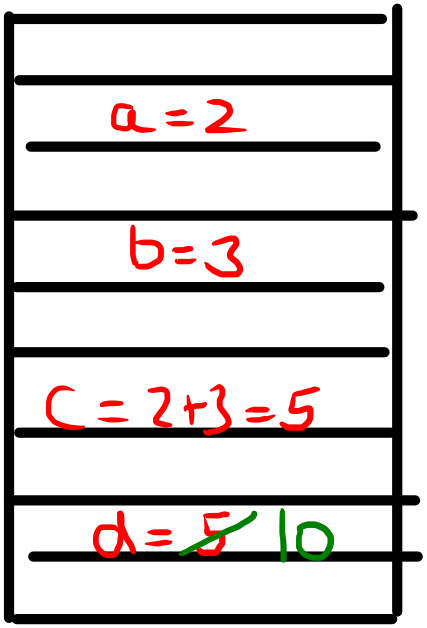

Var x = 5;
console.log(x);
// 5
O/P
5
Memory
x = 5

```
var a = 2;
var b = 8;
var c = a + b;
console.log(c);
```





```
var a = 2;
var b = 3;
var c = a+b;
var d = c;
d = d + a + b;
Console.log(d)
```



d = d + a + b
= 5 + 2 + 3 = 10

```
var a = 2;
var b = 3;
var c = a+b;
var d = c;
d = d+ a+b;
```

Console.log(d) → 10

a = 2

b = 3

c = 2+3 = 5

~~d = 5~~

↳ d = d + a + b = 5 + 2 + 3 = 10

Dry Run

Later topics ⇒ Dry Run

Runtime error →

Compile time error → Go to Internet

Syntax:

How are you

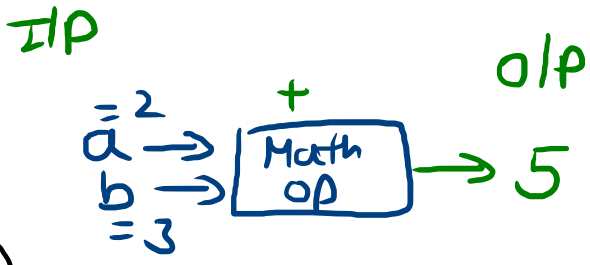
How a you

Partial accepted

→ 10 Testcase

2, 3, 4 Test ✓

4 Relational operators

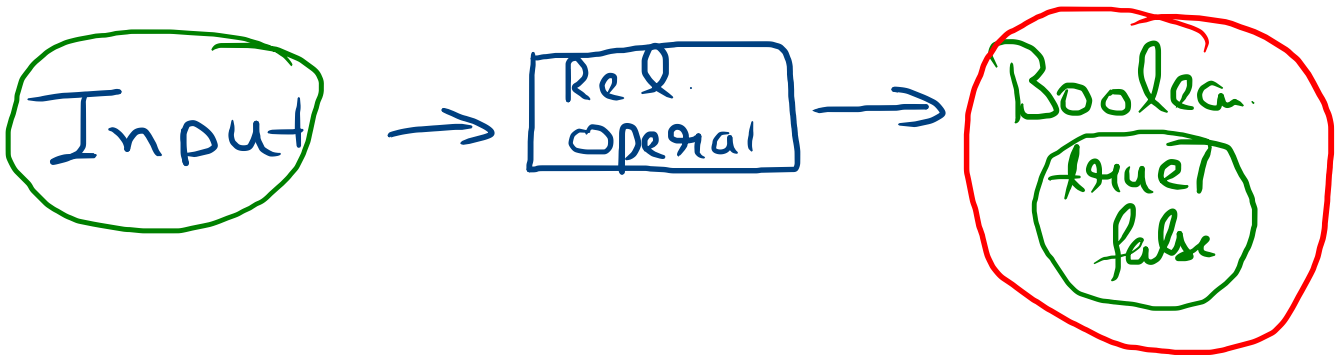


> (Greater than)

>= (Greater than equal to)

< (Less than)

<= (Less than equal to)



(1) > Greater than operator

True ⇒ first value is **strictly** greater than the second value.

```
console.log(5>4); True
```

False ⇒ first value is less than or equal to the second value.

```
console.log(4>5); // false
Consol.log(4>4); // false
```

(2) Greater than (>=)

True: First value is greater than or equal to second value.

Case 1: 5 >= 4
Case 2: 5 >= 5
] True

False: first value is less than second value.

Case 1: 5 >= 12 fails

$10 > 6$

$10 >= 6$

$10 >= 10$

$-9 > -8$

$10 > 10$

Less than operator

True: first value is strictly less than the second value

```
var a = 4;
var b = 5;
```

Case 1: console.log(a < b);

False: first value is greater than or equal to the second value.

Case 1: 5 < 3 false

Case 2: 5 < 5 false

<= (Less than or Equal to)

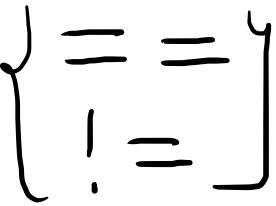
True => first value is strictly less than or equal to the second value.

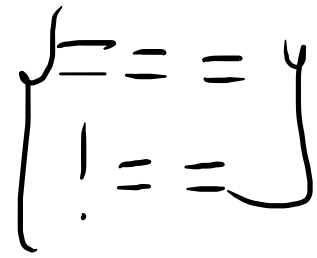
Case1: console.log(5 <= 7) ;

Case2: 5 <= 5

False => first value is greater than second value

Case 1: 5 <= 3 false





==

$$2 == 2$$

$$\Downarrow$$

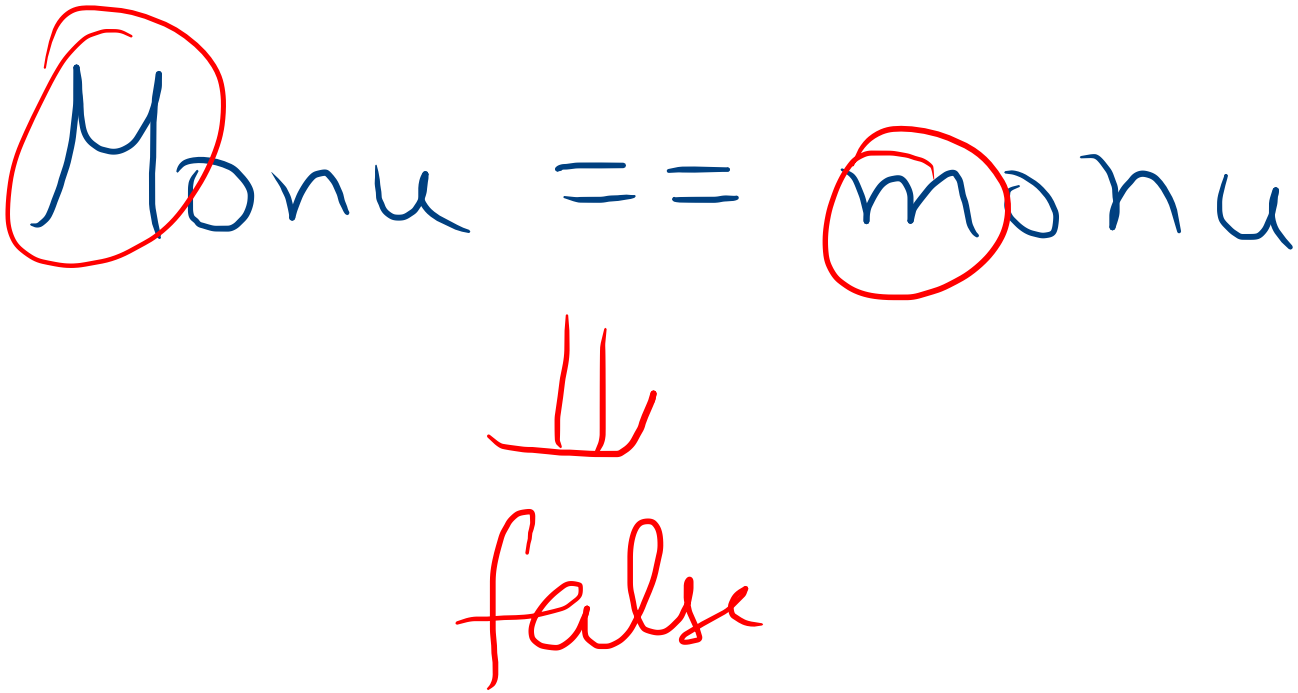
True

"Babli" == "Babli"

True

Dharam == Param

false


Monu == monu
false

I don't believe in caste

==

(value)

"2" == 2 ⇒ True

believ of caste

===

(value + datatype)

2 === "2"

↓ number ↓ String

false

!=

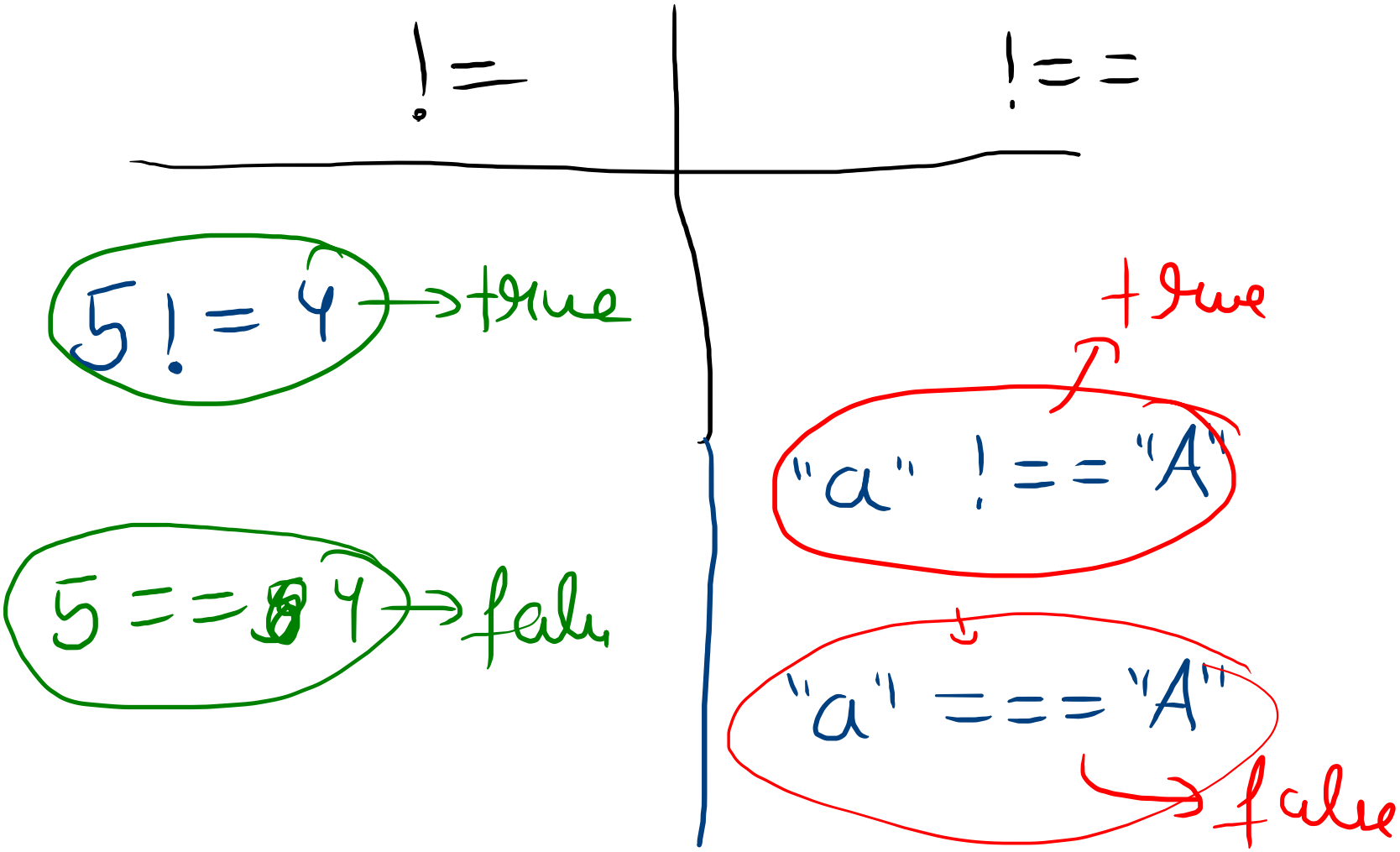
Opposite of ==

* Consider only the value

!==

Opposite of ===

* Consider value + datatype


!=
!==
5 != 4
true
5 == 4
false
"a" !== "A"
true
"a" === "A"
false

find o/p

Dry Run.

```
5 == 5      // true
5 != 5      // false
6 == "6"    // true
6 !== 7     // true
"a" == "A"  // true
"a" != "A"  // false
```

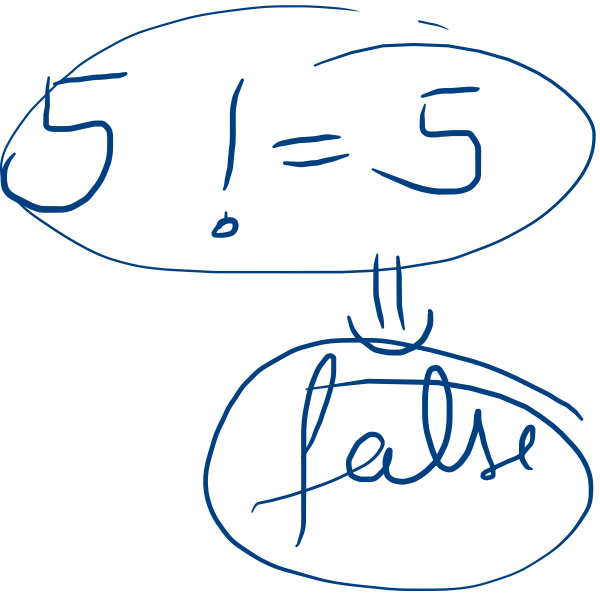
6 === 7 = false

"a" == "a" true

6 != "6"

6 == "6"

True

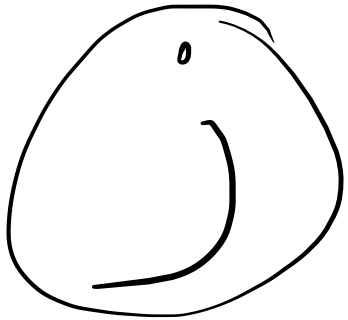


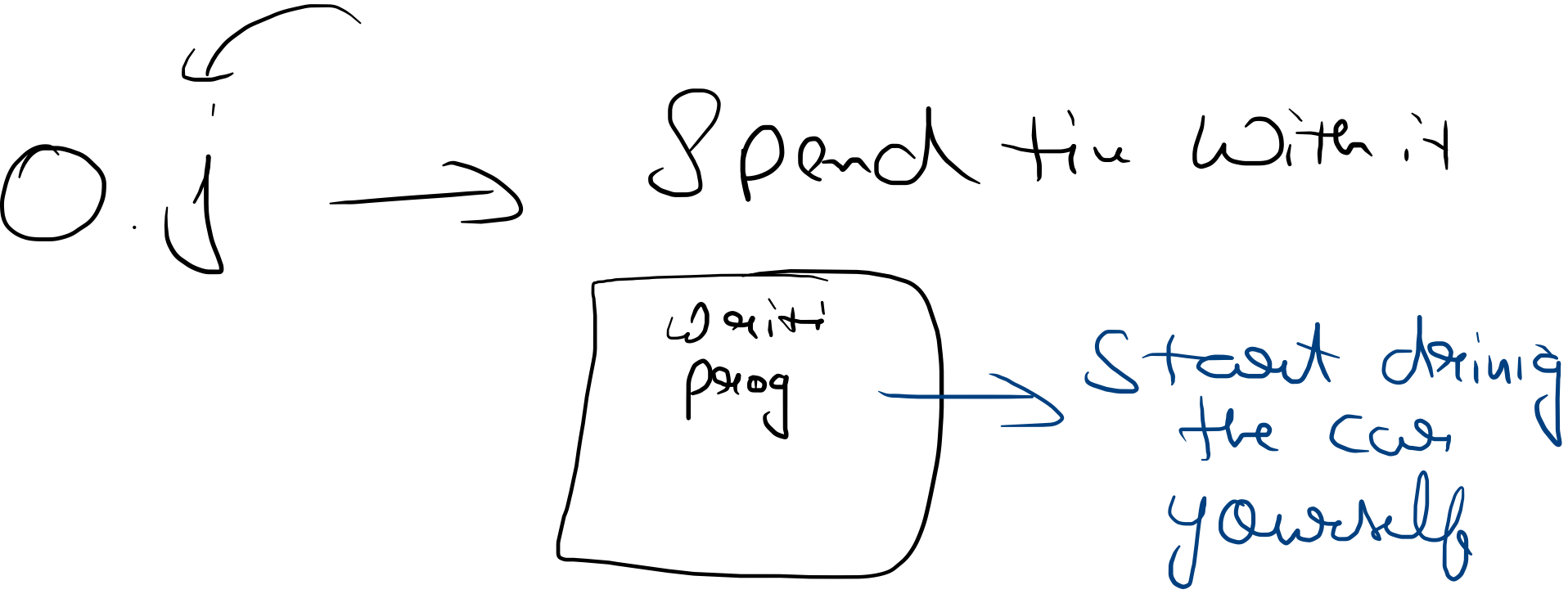
5 == 5

↓

True

Good practic

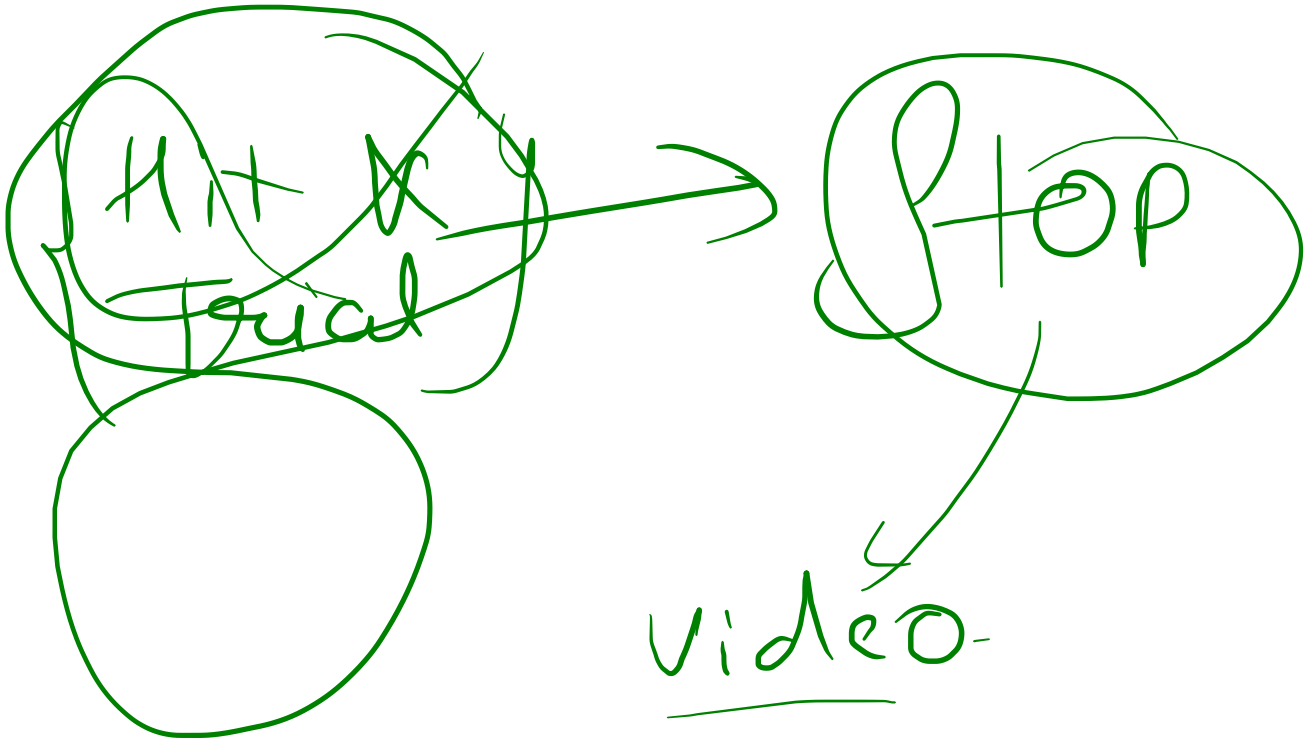


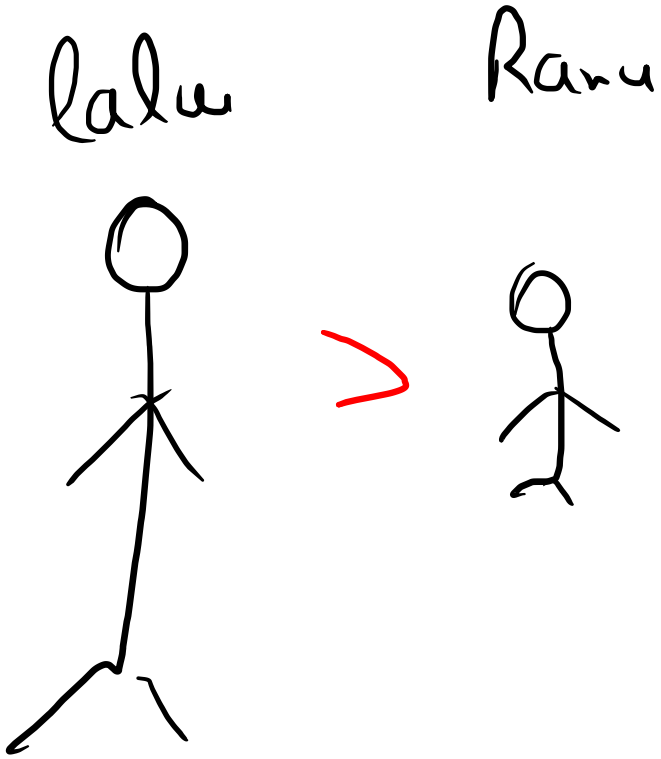

Spend time with it
Write prog
Start driving the car yourself

1000 → ಒಂದು ಸಾವಿರ

Don't go the Shortcuts

Not Leaving Proper Answer


Hit X
Twice
Stop
Video


lalu
Ramu