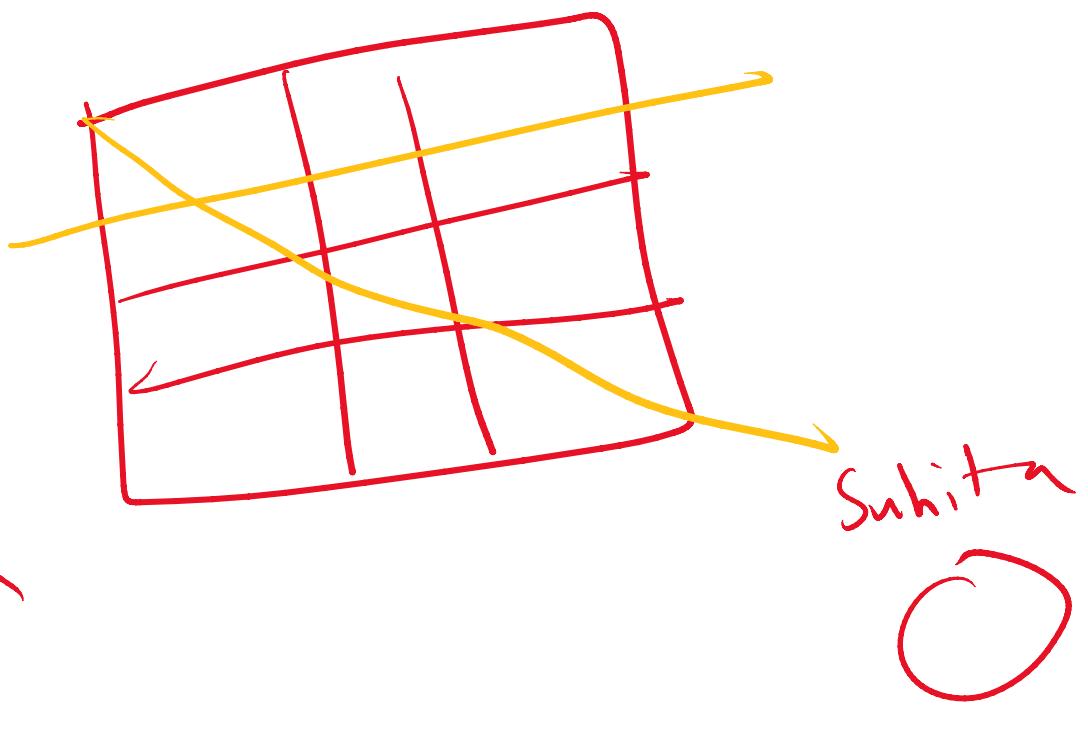


Let car = {
 "color": "white",
 "seats": 2,
 "nitros": false}

3
const msg = "My car have " + car.seats
+ " seats and it is of color "
+ car.color *

+ car. color #
or
car {"color"}]





Objects Continue

17 August 2023 21:11



let example

```
Student = {  
    name: "bhaveesh",  
    Standard: 9,  
    mobNumber: 1234  
}
```

Student.name =

Student.Standard = Student["Standard"] = 9

Student.mobNumber = Student["mobNumber"] = 1234

console.log(Student.name);
↓
Property/Key Object

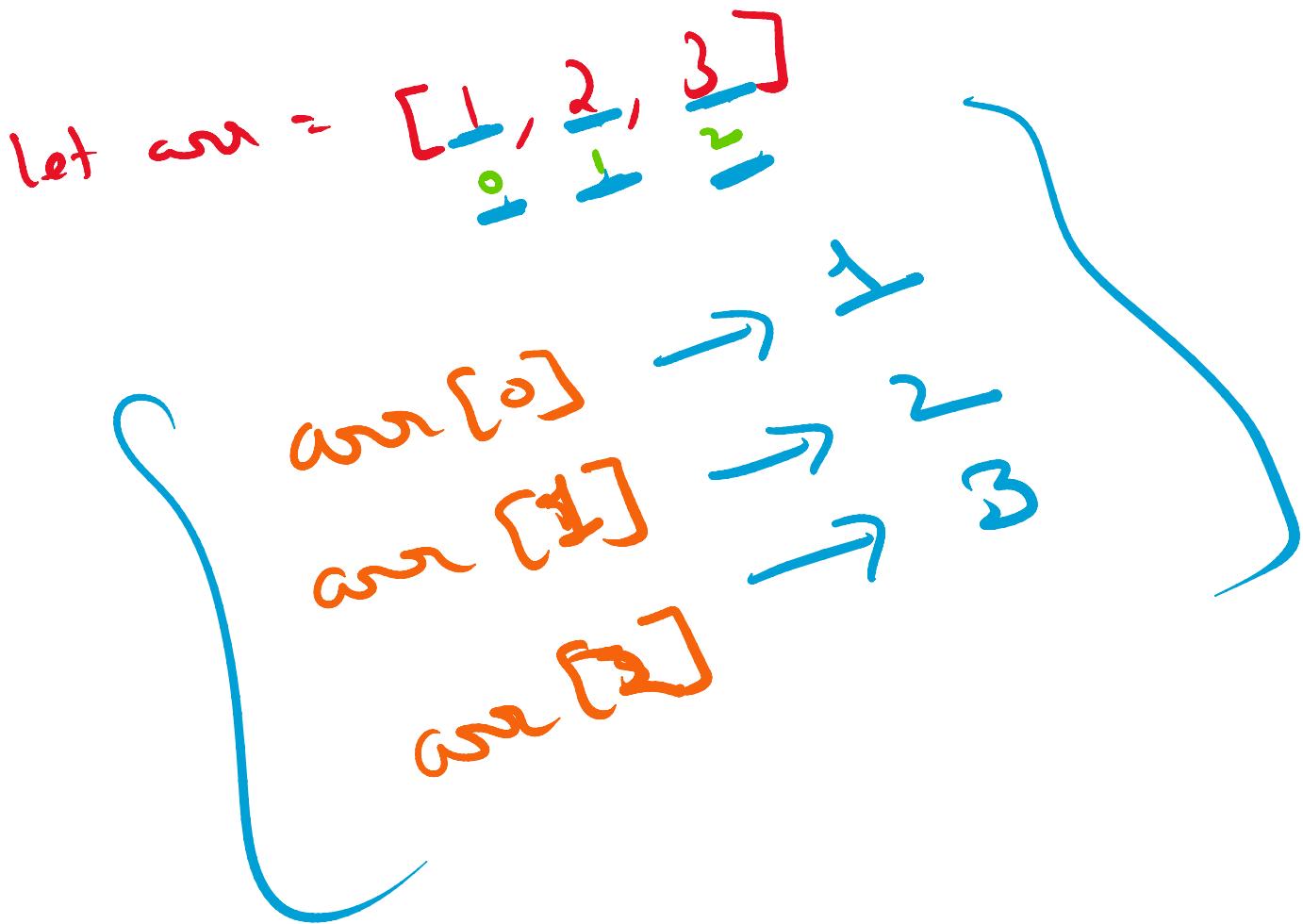
console.log(Student["name"]);
↓
NOO PK

1234

9

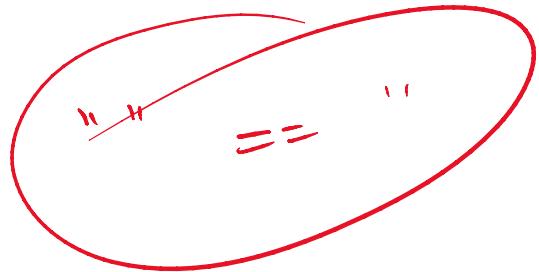
Arrays Continue

17 August 2023 21:14



String

17 August 2023 21:39



const msg = "My car have " + car.seats + " seats and it is of color " + car.color + ""

3 times
↓
concatenation

" + car.color + "
or
car["color"]

" + car["color"] + "

"My car have \$ {car.seats} seats and
it is of color \$ {car.color}"

Functions

08 August 2023 20:51

```
var x = 5;  
console.log(x);  
var x = 10;  
console.log(x);
```

In Java from which function execution
of code starts
↓
main

~~function~~ **name of function** **Arguments / Parameters** {
 Business logic }

JS reserved keyword for declaring functions }
 function keyword

print **argument** **(value)** {
 console.log(value)
 } **B.L**

... + (" hello ")

print ("hello")

Single Thread vs Multithread

08 August 2023 20:52



DSA → faster solution

1 operation at a time

①

$$2+2$$

②

$$3+3$$

$$JS \approx \underline{2\text{ sec}}$$

$$\frac{2+2 + 3+3}{2+2} \rightarrow 1 \text{ sec}$$

Stable System
Perfect

Java
C++

Perfect
Accompanier

C++

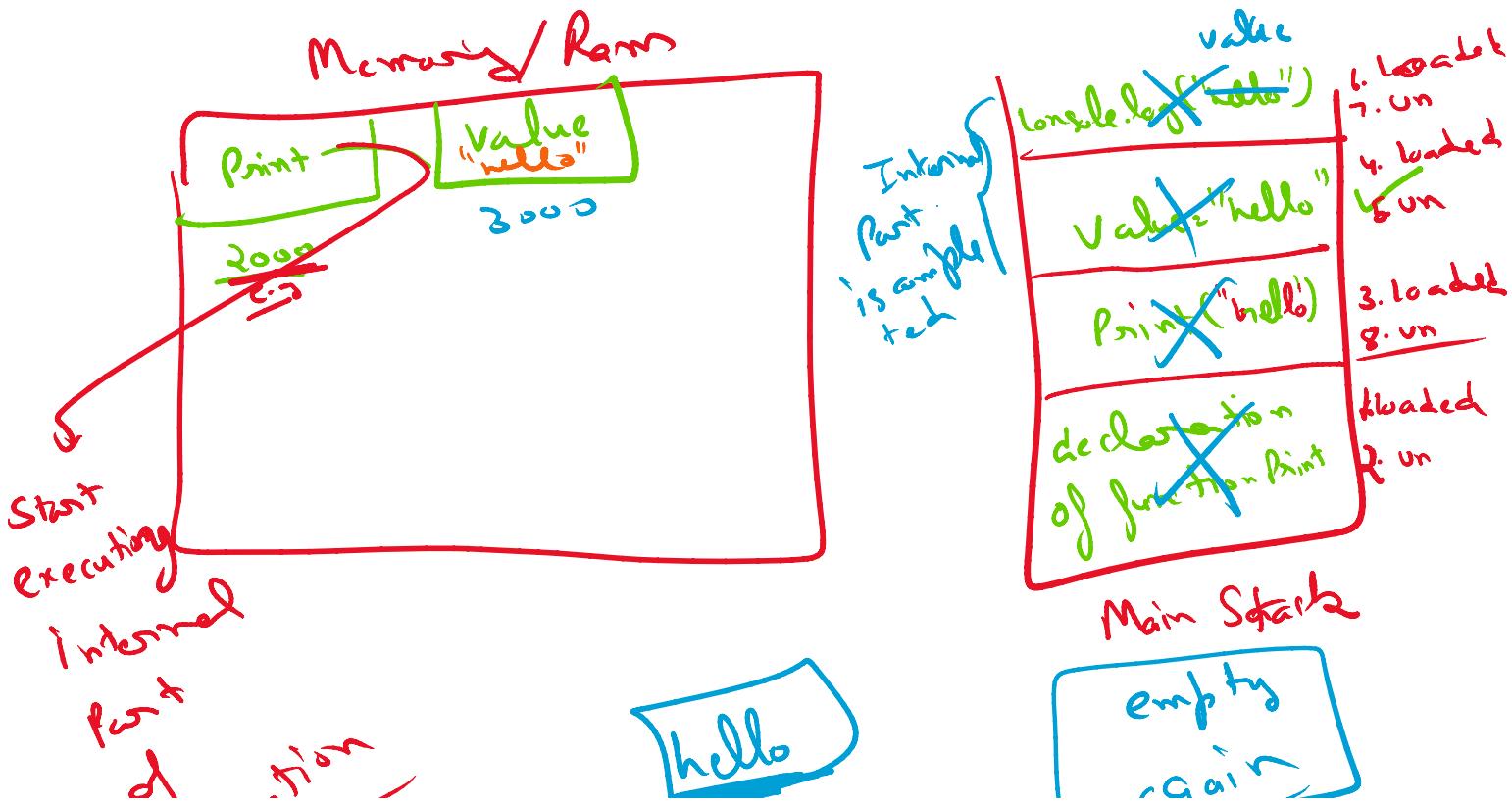
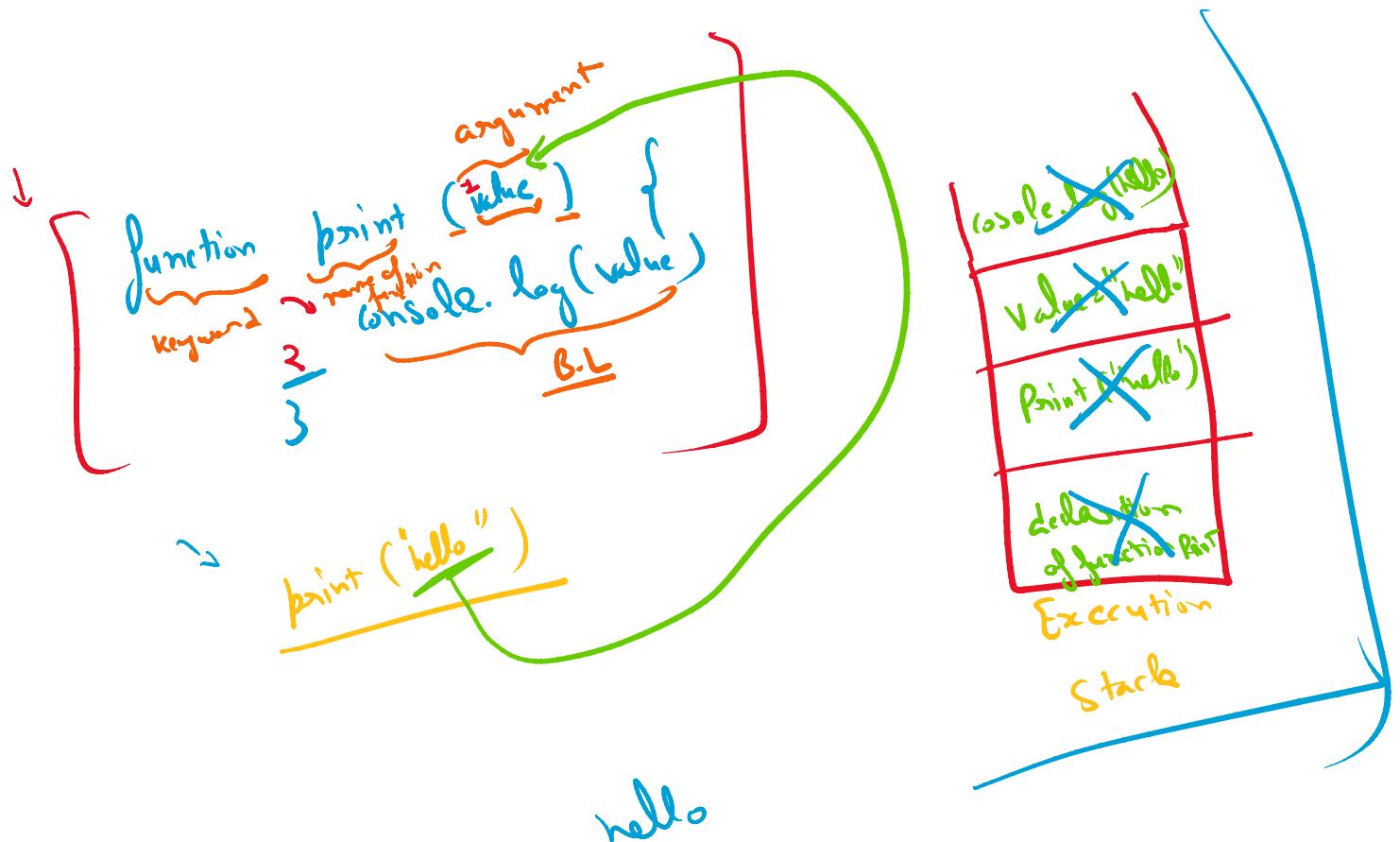
Mathematical →

Web development

{ Python
Rust
R → SS
C/C++ }

Execution in JavaScript

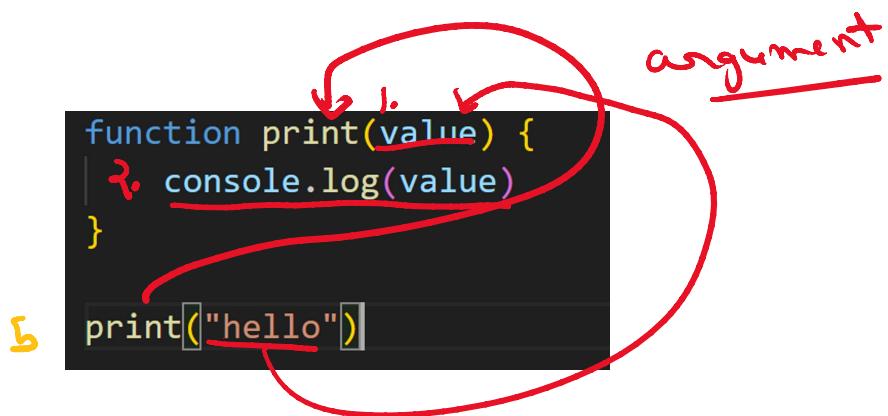
08 August 2023 20:51



for
this function

Hello

exit
again



Debugging tools

17 August 2023 20:51

1. `console.log()` statement

2.

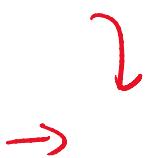
```
functions.js X VM114 VM123 VM155
1 function print(value) { value = "hello"
2 console.log(value)
3
4
5 print("hello")
```

1. Search, you will find all the code

2. Red dots, Breakpoints

Execution will pause
at every breakpoint

Recursion



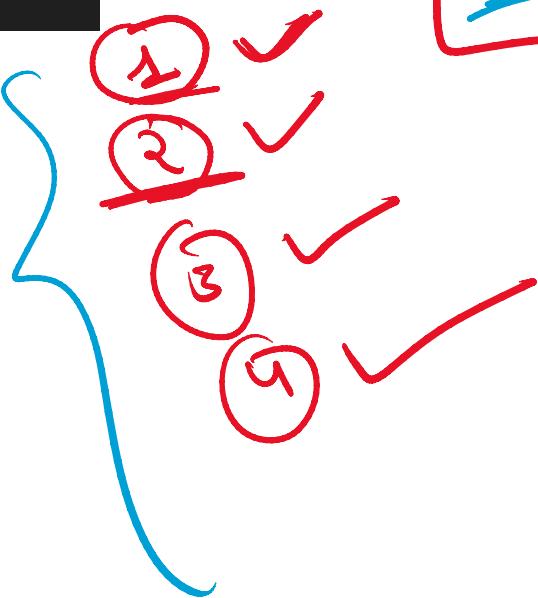
val = 1 & 2

~~function recurse(val) {
 if(val === 5) {
 return
 }~~

val = 1 & 2 & 3 & 4 & 5

|
| recurse(5)|

```
if(val === 5) {  
    return  
}  
✓ console.log(val)  
✓ val = val + 1  
recurse(val),  
}  
  
recurse(1)
```



empty
stack

Explicit type conversion

17 August 2023 20:52

Hoisting

17 August 2023 20:58

Arrow functions

17 August 2023 20:59

Operators

17 August 2023 20:59

Loops and control flow

17 August 2023 20:59