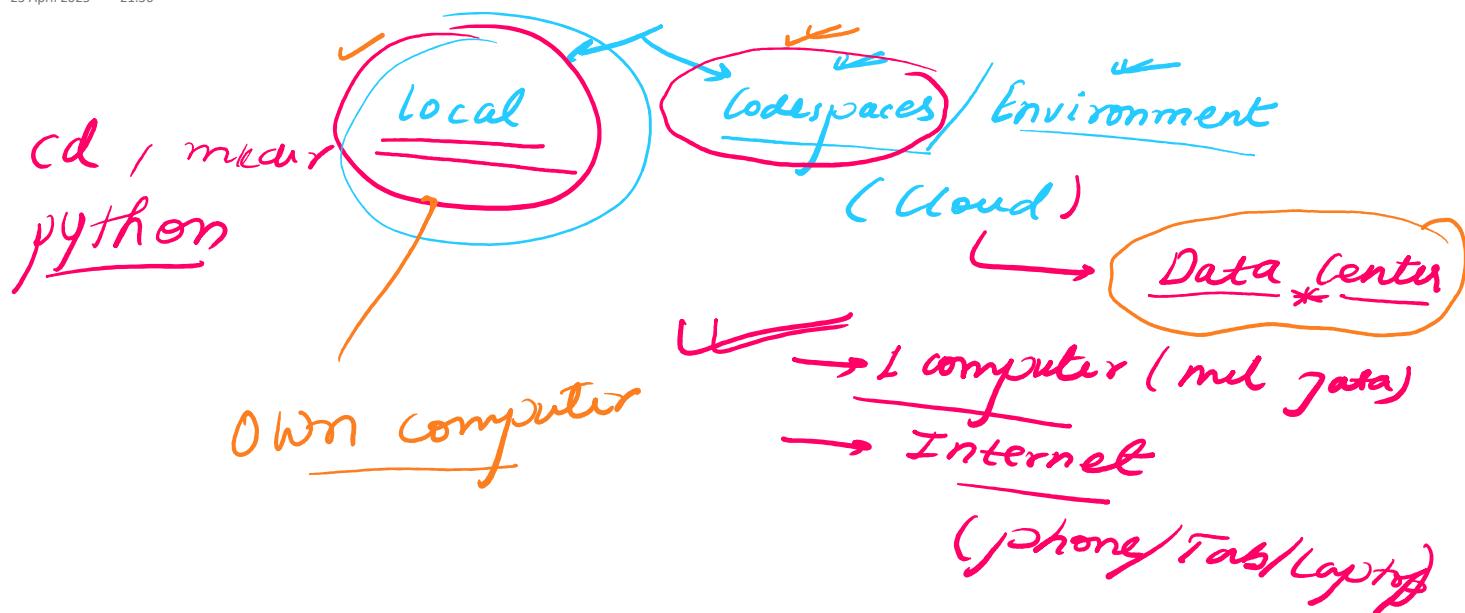
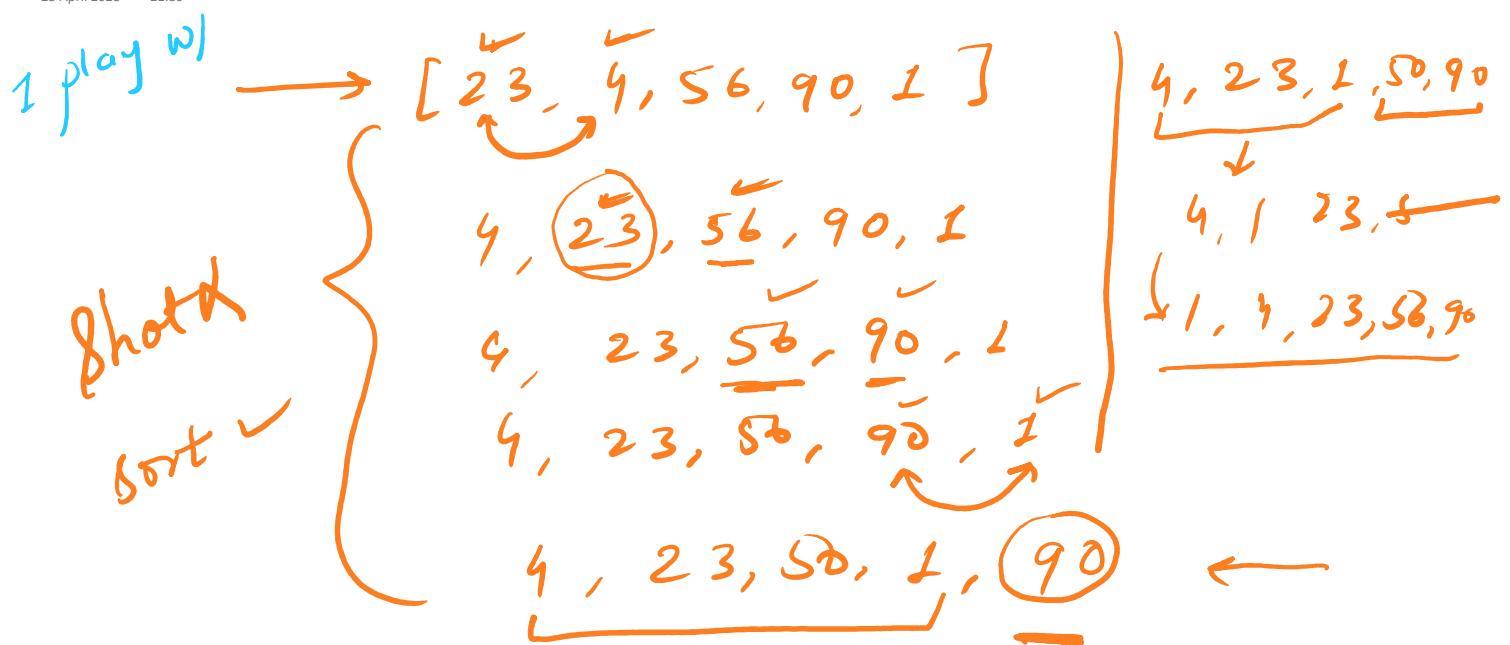
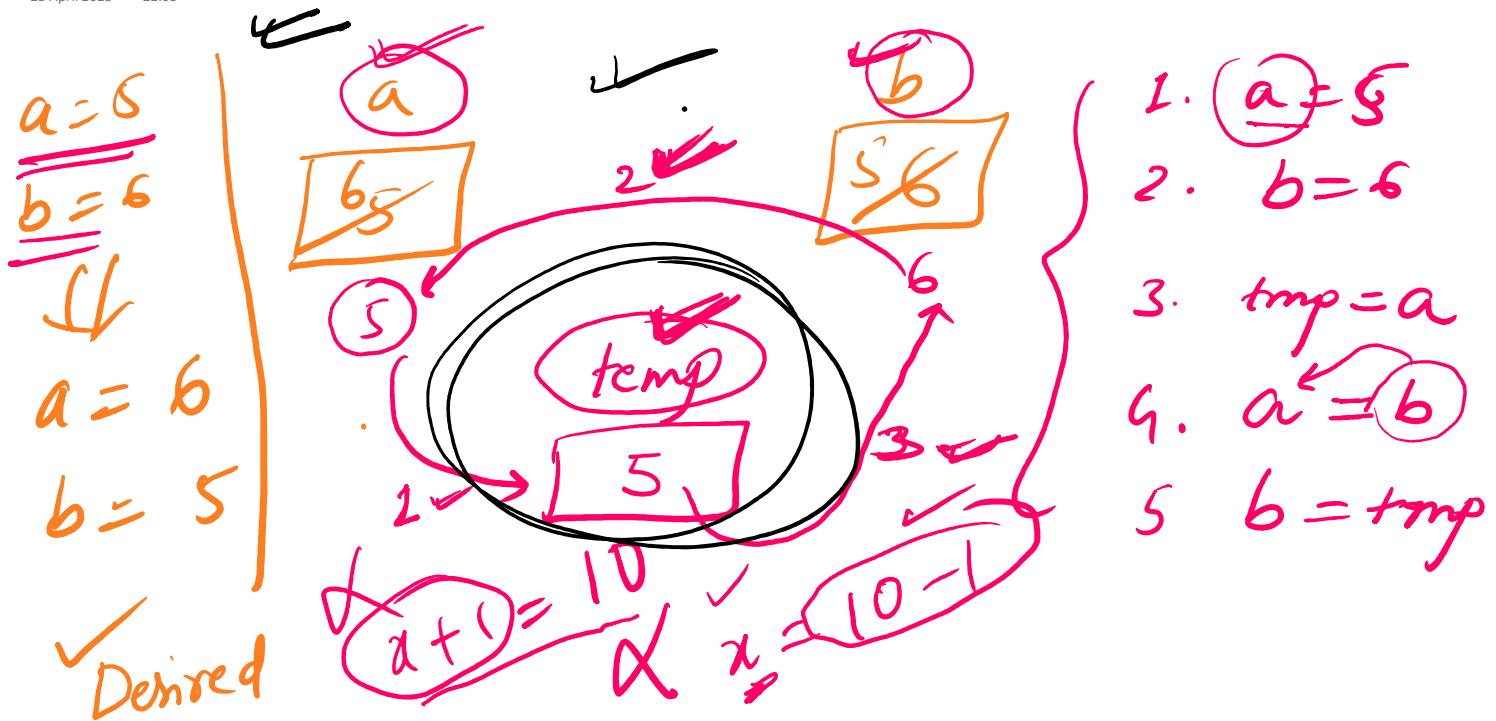


Practice Session

23 April 2025 21:36







Java, C++,
JavaScript,
.NET, TypeScript
 $j = \text{len}(lis)$

$J+L$

[0, 1, 2]
[10, 20, 30]

list Index out of
bound

11

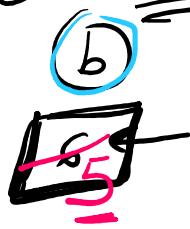
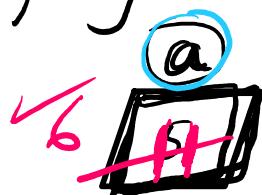
Pythonic

Swapping Deep Dive named mem. location

$a = a + b$
 $b = a - b$
 $a = a - b$

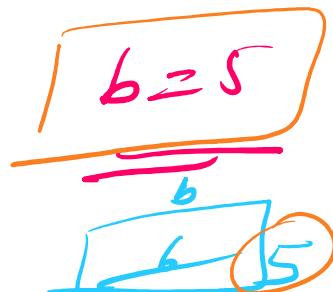
may produce problem

$$11 - 5 \Rightarrow 6$$



$a = a * b$
 $b = a // b$
 $a = a // b$

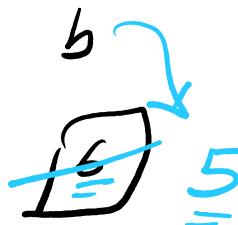
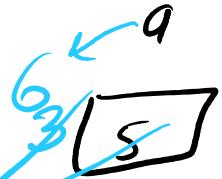
$$30 * 5 \Rightarrow 0 \quad b = 5$$



temp a

$a = a \wedge b$
 $b = a \wedge b$
 $a = a \wedge b$

101



$$\begin{array}{r} 011 \\ + 101 \\ \hline 110 \end{array}$$

$$\begin{array}{r} 5 \\ \Rightarrow 101 \\ 6 \\ \Rightarrow 110 \\ \hline 110 \end{array}$$

$$011 \Rightarrow 3$$

110

$$\begin{array}{r} 011 \\ + 110 \\ \hline 101 \end{array}$$

XOR \oplus

x_1	x_2
0	1
1	0

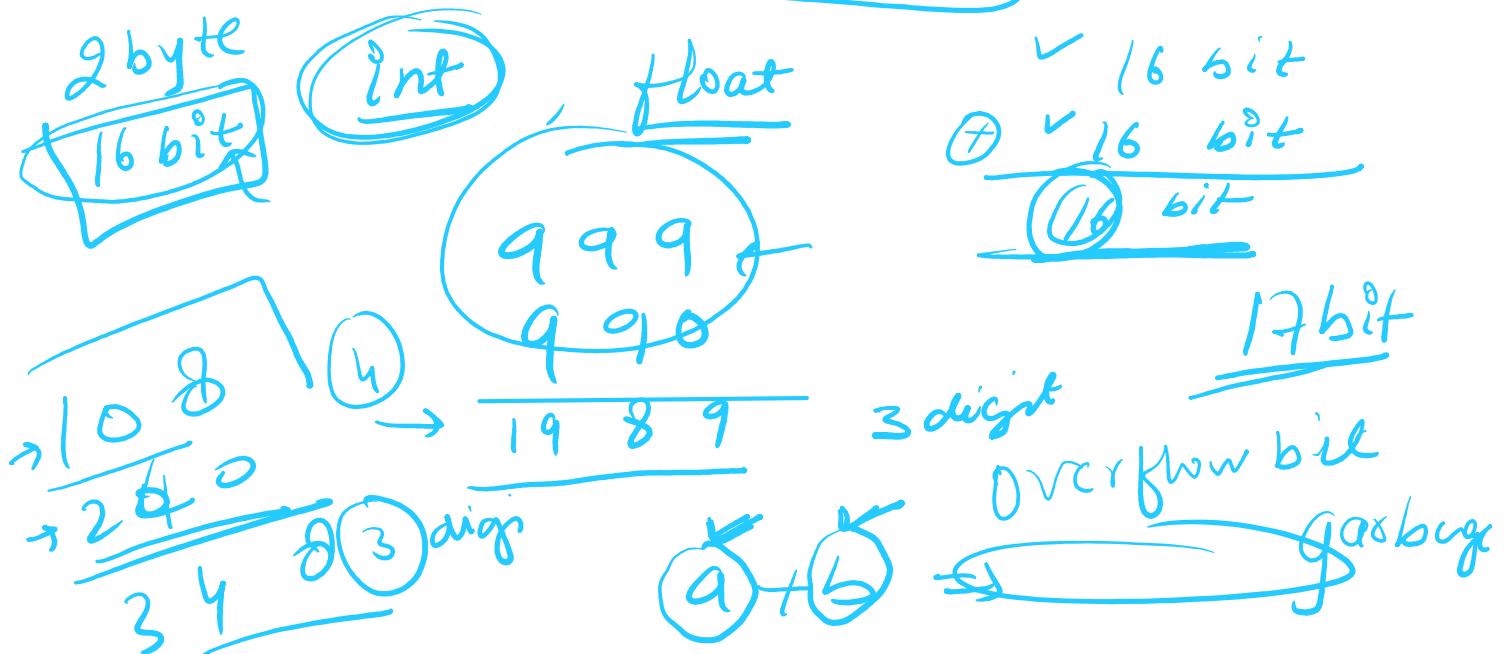
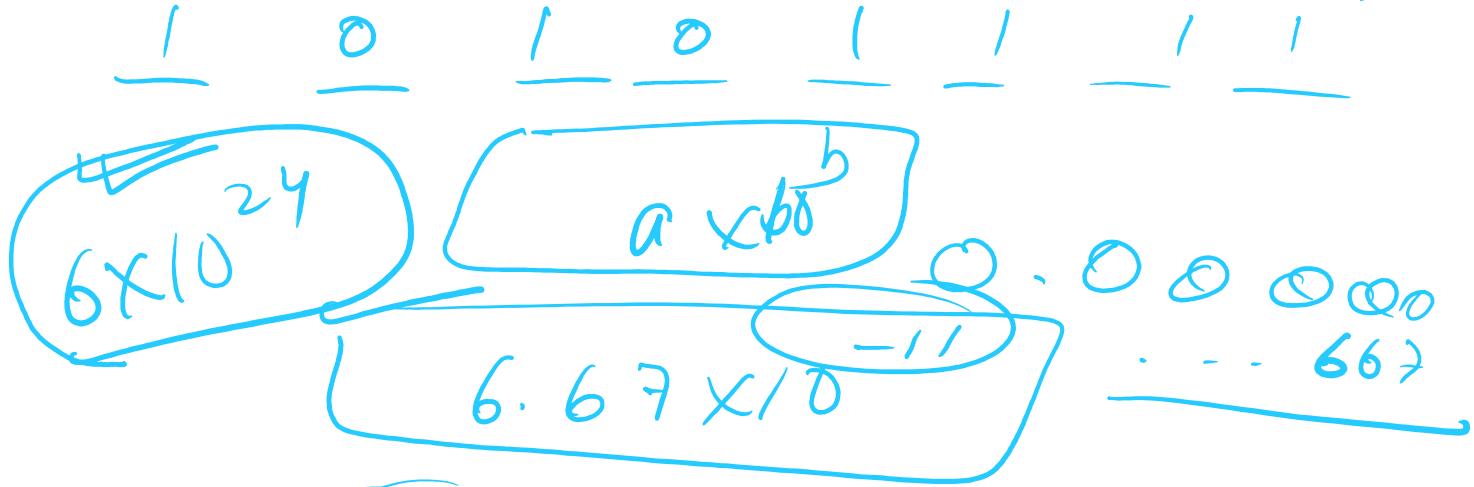
$$\frac{y}{\begin{array}{r} 1 \\ 1 \end{array}} \Rightarrow a = 6, b = 5$$

Different

✓ 1 0
 → 0 0
 → 1 1

1 JY. N

⑤] same

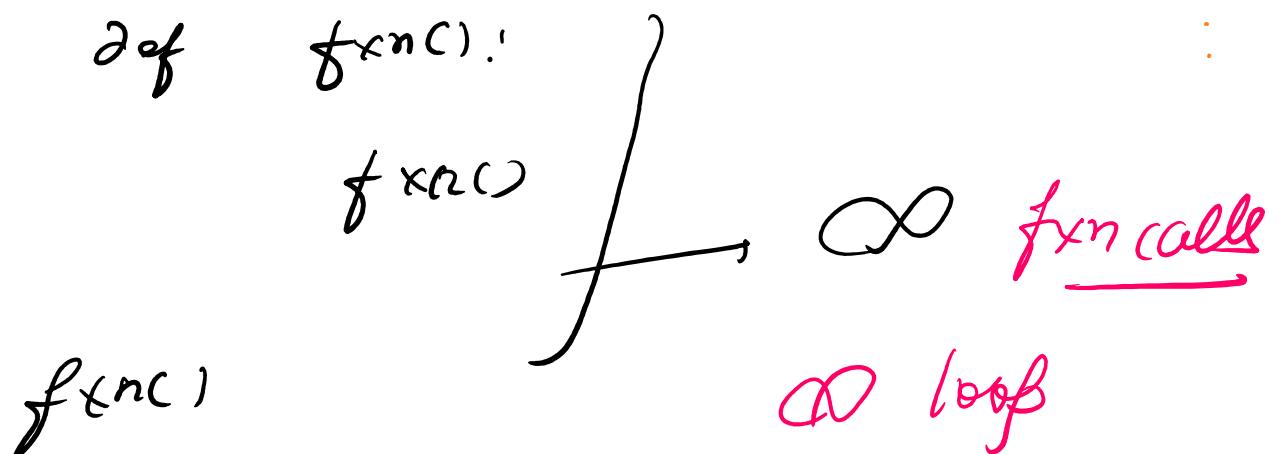
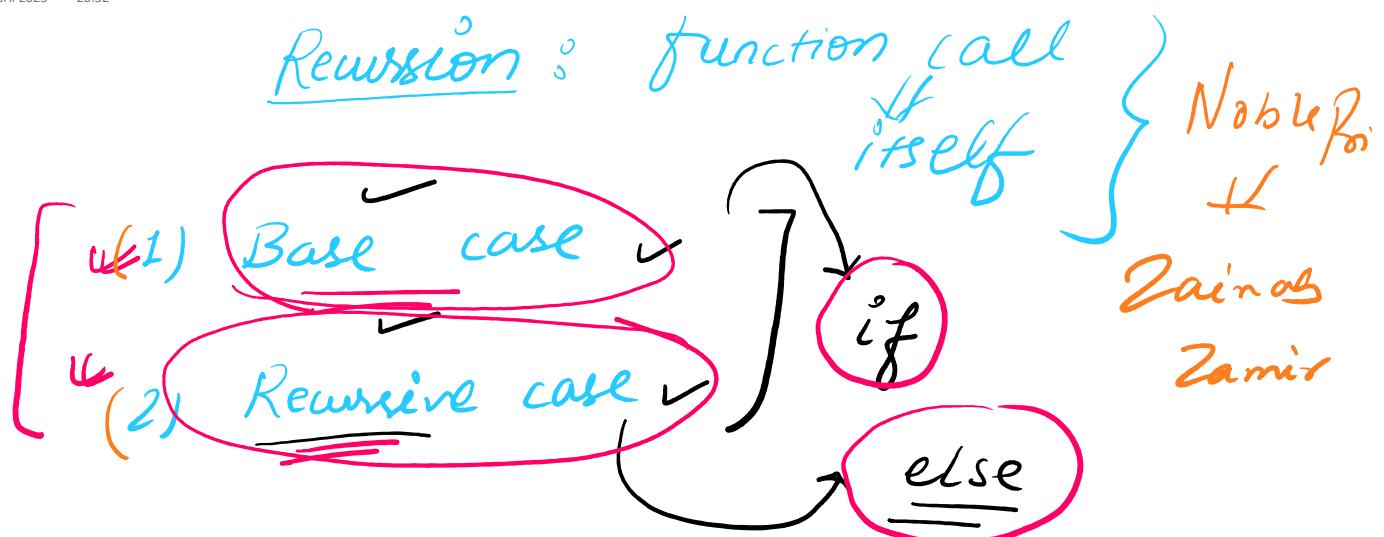


$\begin{array}{l} \text{a: int} \\ \rightarrow b: \text{int} \\ \text{temp: int} \end{array}$ ✓ $\begin{array}{l} \text{temp} = a \\ a = b \\ b = \text{temp} \end{array}$
 Safe { }
 Unsound { }

"In Physics, we apply
 but in CS, "

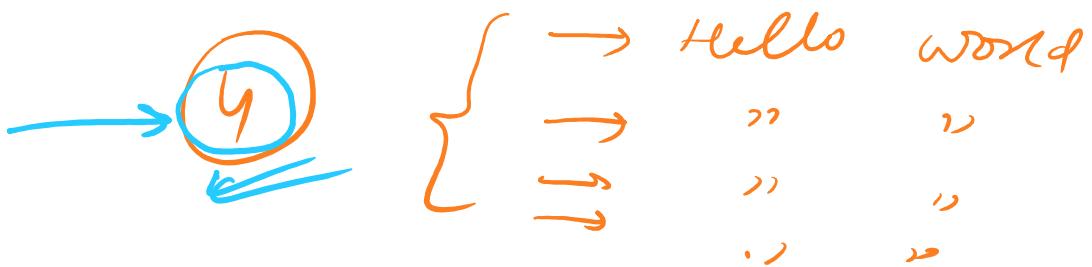
```

    graph LR
      P((P)) --> M((M))
      M --> CS((CS))
      CS --> SKA([SKA])
  
```



"Hello World"

```
for i in range(0,5):
    print("Hello World")
```



def print_hello(n):

~~base case if (n == 4):~~ 1

2

Base case if ($n == 4$):
 return
 Recursive case
 else:
 print("Hello World")
 print-hello($n+1$) *

Home Work:

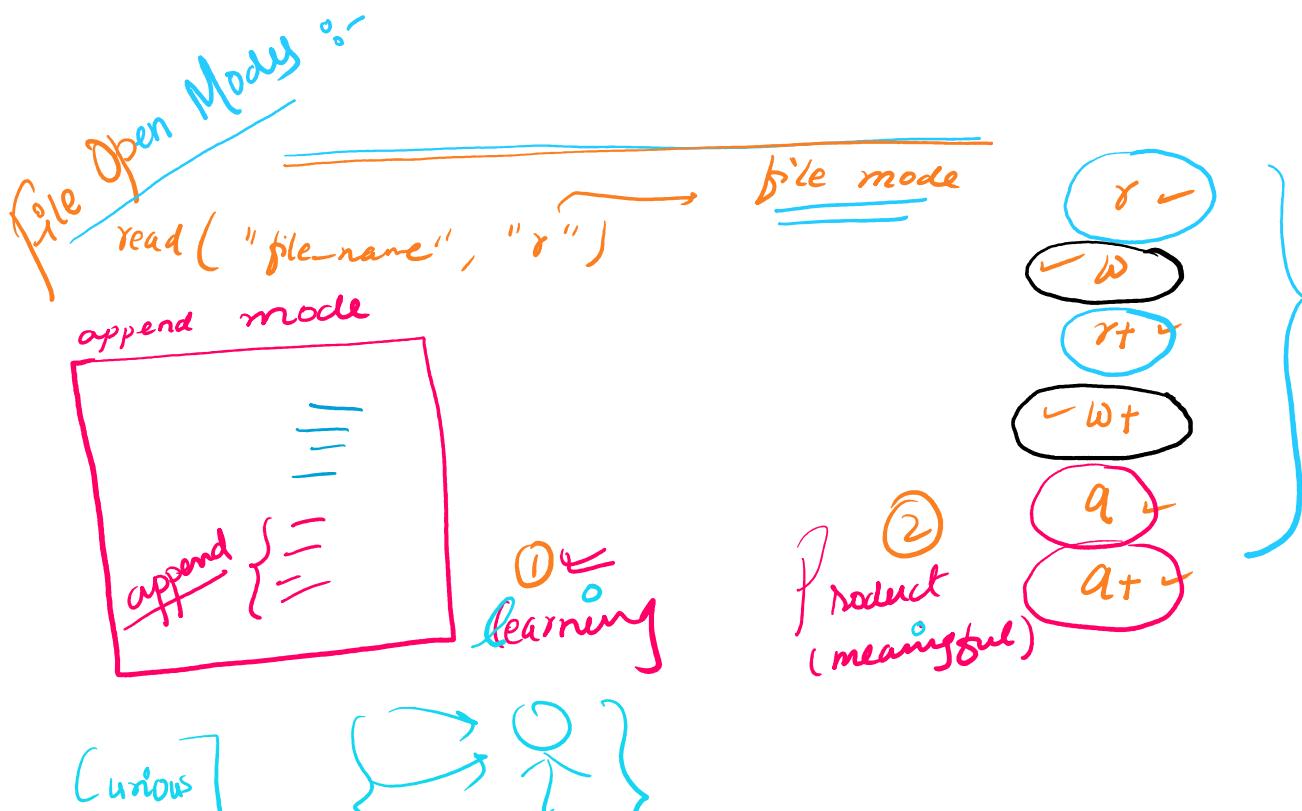
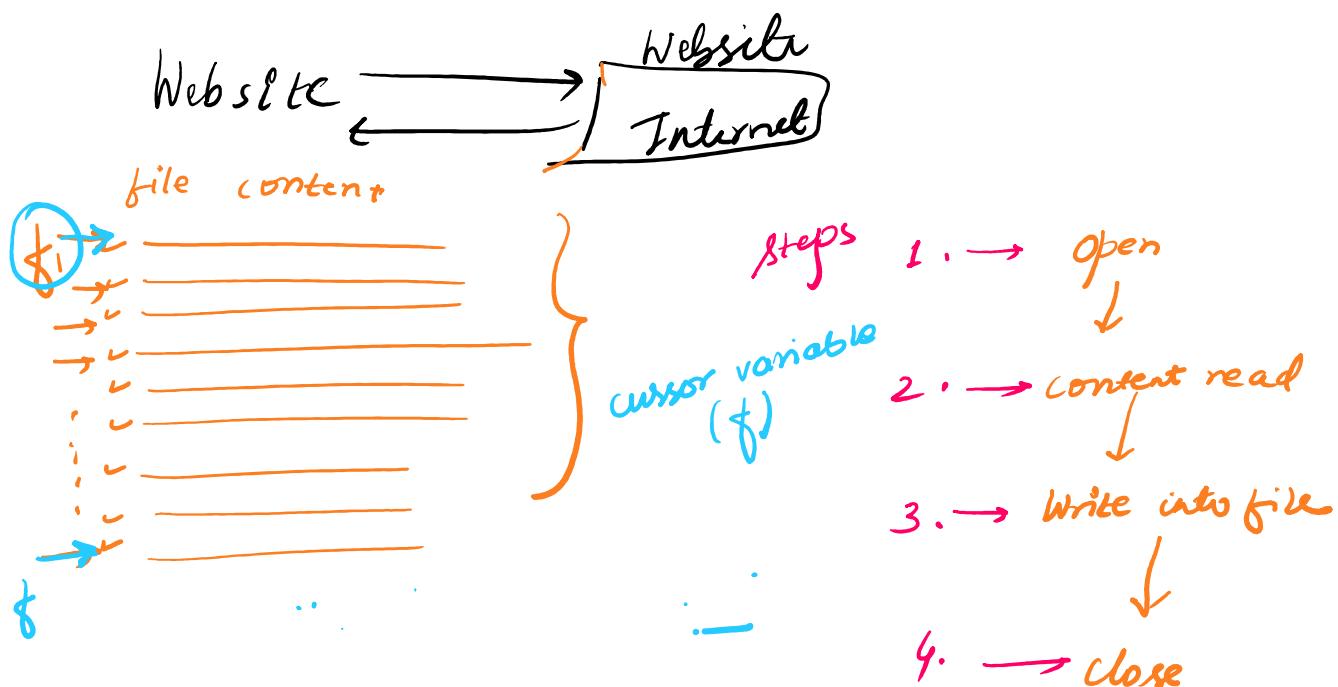
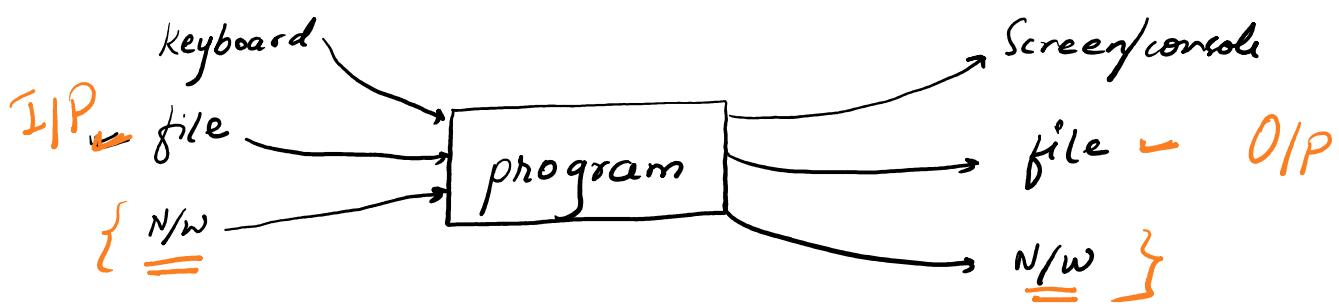
$\boxed{1 \ 2 \ 3 \ 4 \ 5}$

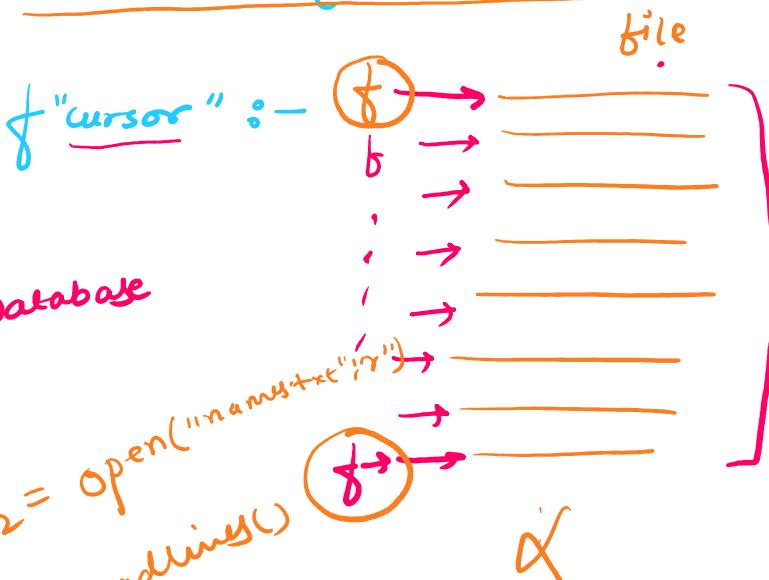
1) print 1 to 5 using loop & recursion both

2) print 5 to 1 >>> " "

$\boxed{5 \ 4 \ 3 \ 2 \ 1}$

File Handling (Why?)

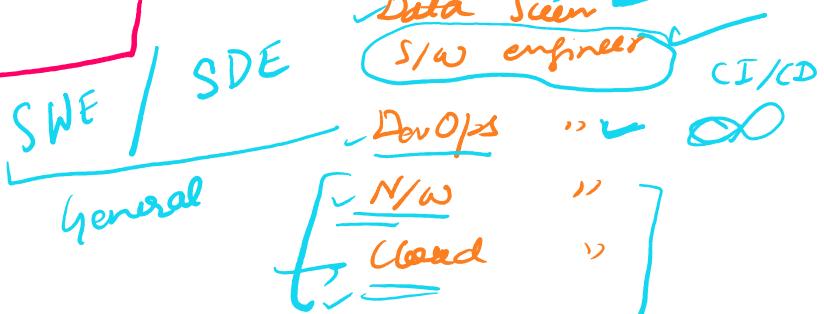




① f = open("name.txt", "r")

names = f.read()

names2 = f.readlines()
print(names2) == []



→ DSA
"Data Structures Algorithms"

Standard * Baat
DSA

→ Summary
 { → file handling
 → f.read
 → content type

→ ASC, DESC

sorting
 ↘
 ↘

{ 1-2 Madarse
 ↓
 Madarse

=====

Luck * School
 (Director)

→ Purani 1/11/11

Test X

6th

→ Purani
→ "Old"
→ belt

U n

6th

terminal

3rd Rank

Half yearly

1st Rank

Annual

1st Rank]

IAS

→
→

School Topper

Dreams

7 8 9

10th

UP Board

District

→ 2 marks

→ 90.17%

→ 9th rank
→ 10th "

↓ 543

Sub-Division

541

25th

11th

IAS

Dishati IAS

Vikas Dityakirti
(All videos)

BCA * 3rd

[BCA
2nd year]

* Personal (Read / Teacher)
Own

Dreams

IAS

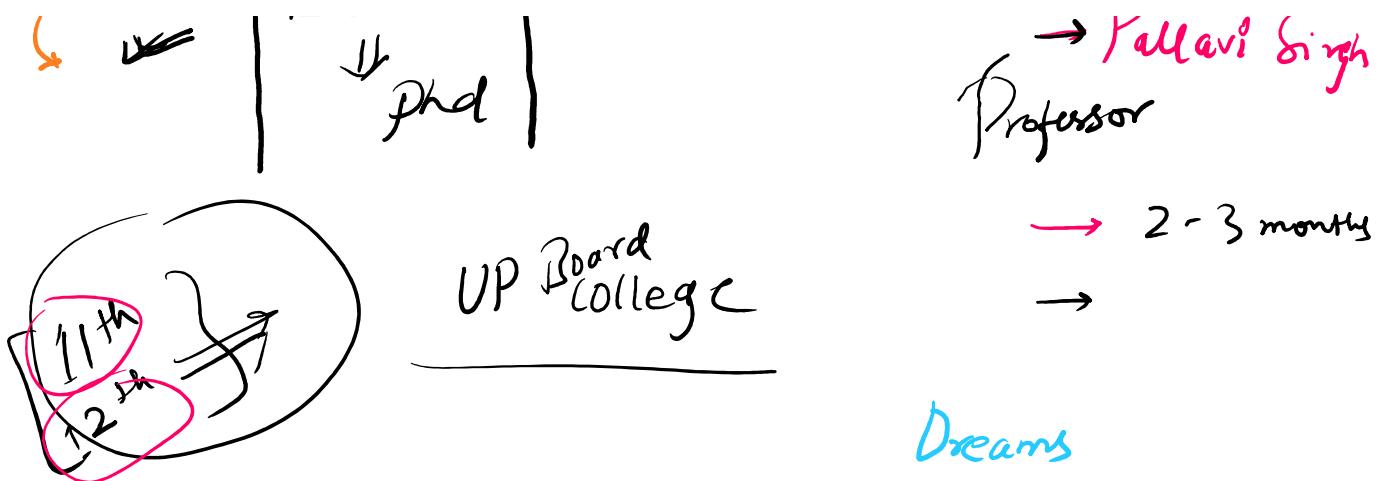
Physics

Dhd

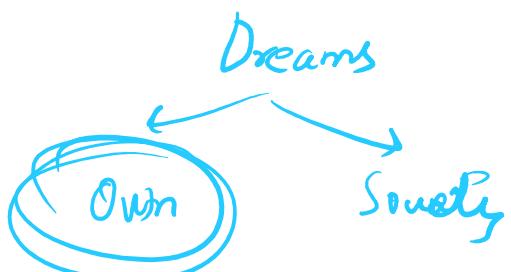
12th

Society (99%)

Pallavi Singh
F... ...



Self Study
 ↗
 BSc



Computer Science
 ↗
 1st 2nd

BSc 98

Maths
 ↗
 3rd : Physics

Physics C.S.

$$\begin{array}{r}
 \text{3rd} \\
 \rightarrow 8.8 \\
 \rightarrow 8.9
 \end{array}$$

1, 2

Matter → Tabassum

Anas

9.4

AMU

$$\begin{array}{r}
 \text{1st} \\
 \text{2nd} \\
 \text{3rd} \\
 \hline
 \text{8.5}
 \end{array}$$

mid term

9.4 3rd Rank

1st, 2nd
 2nd Keshvi

2nd Keshvi

{ 8.5 }

← 9.0
↓ 8.97

[LEARNING]

Pilot

60
Frustrated +

Professor

PGT

Reading / Teacher

Memory Concepts

Type

→ Stack memory

→ Heap memory

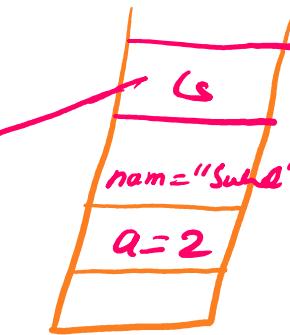
Non-primitive / User defined

→ object

→ DS i.e.

[1, 2, 3, 4]

DS



a = 2

↓ ↓ ↓

ls = [1, 2, 3, 4]

↑

(stack)

{ primitive
int
char
boolean
double
float
str

(Heap)

a = 5

a = 6

~~6 5~~

ls = [1, 2, 3, 4, 5, 6]

ls = [1, 2, 3, 4, 5, 6]

||

Stack

ls[1] = 11

Heap

$dt1 = \{ 'a': 25,$
 $'b': "Suhel" \}$

$firstname = "Suhel"$
 $lastname = "Khan"$

$dt2 = \{ 10: 20,$
 $120: 100 \}$

fullname = $firstname + lastname$
 fullname = "Suhel Khan"

$d3 = \{ 'a': 25,$
 $'b': "Suhel",$
 $10: 20,$
 $120: 100 \}$

for $f = \text{key, value in } dt1.items()$
 iterate
 $dt1.items()$

key = a
 value = 25 "Suhel"
 ✓

Searching

29 April 2025 10:51

Question

$\{ \text{tup} = (\underline{10}, \underline{20}, \underline{13}, \underline{99}, \underline{100}, \underline{150}) \Rightarrow 6$

$\text{item} = \underline{99} \Rightarrow 3$

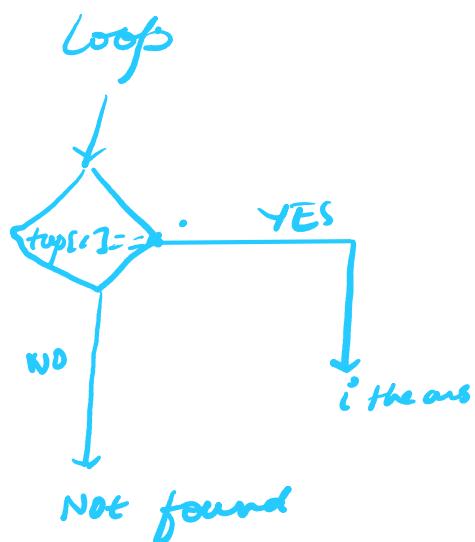
$\text{item} = \underline{150} \Rightarrow 5$

LOGIC

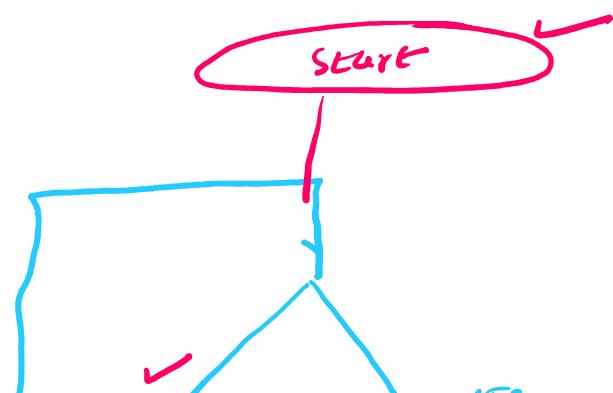
✓ Pseudocode

- Step 1: "for" loop for iteration
- Step 2: check if $\text{tup}[i] == \text{item}$
 i is the position
- Step 3: come out of wif, i

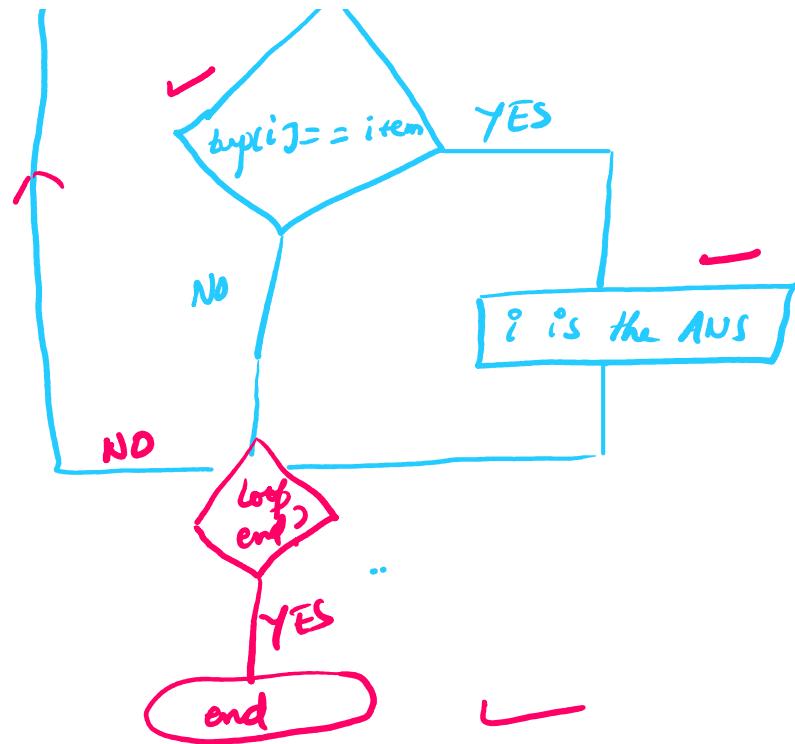
Flowchart



Computer

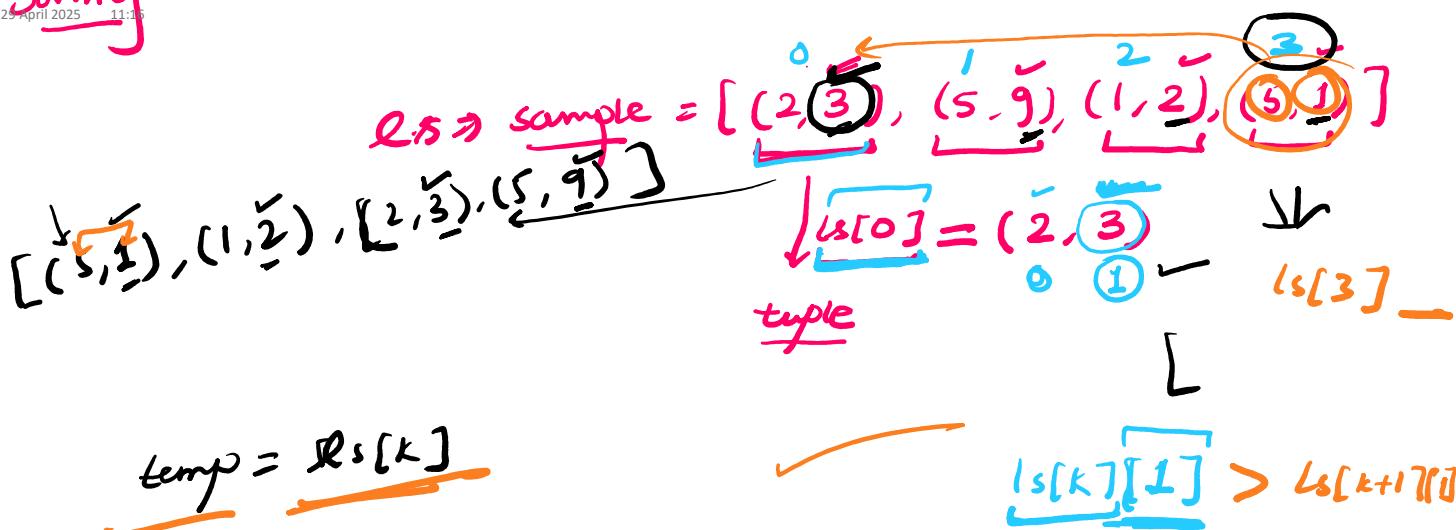


Computer
flowchart



Sorting

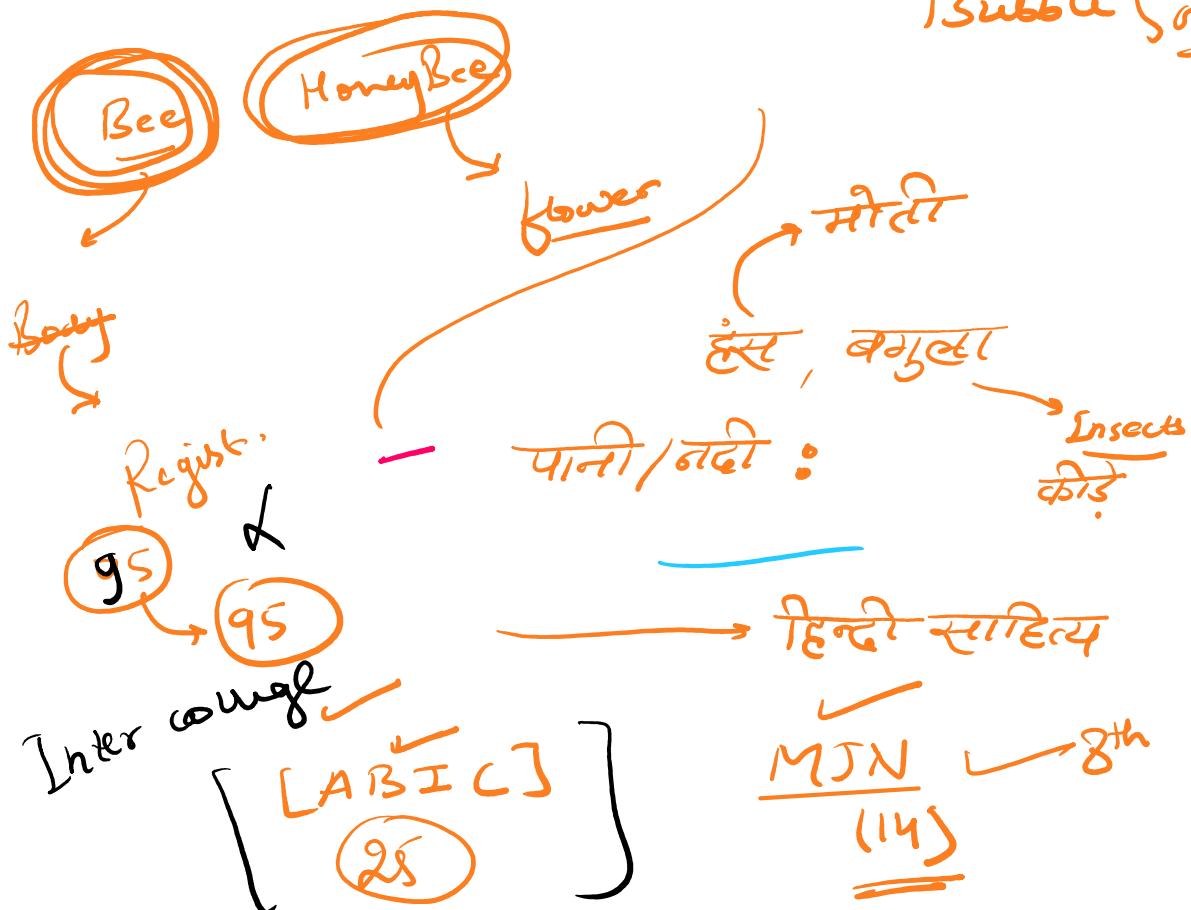
29 April 2025 11:10



Understanding

$\rightarrow [2, 3, 1, 9, 4]$

Bubble Sort



file Handling Project

29 April 2025 11:32

Database

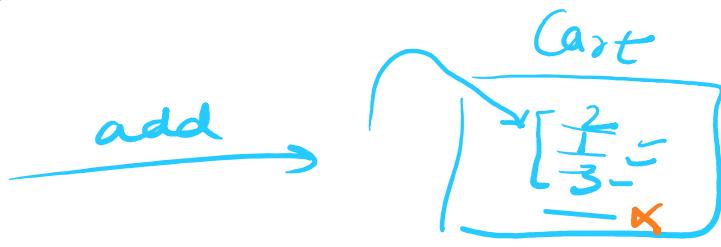
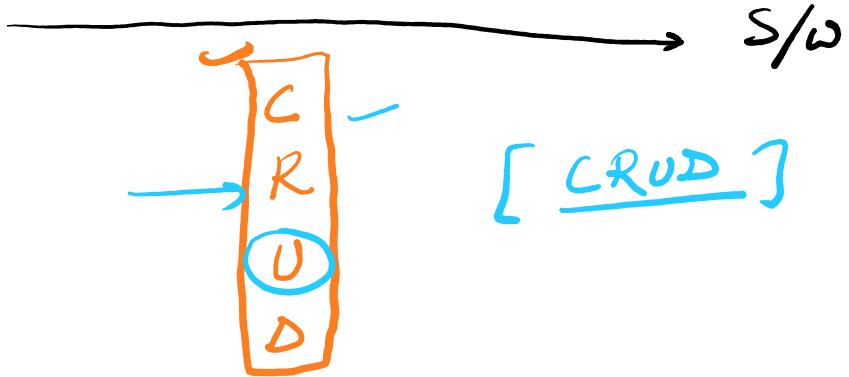
TODO :-

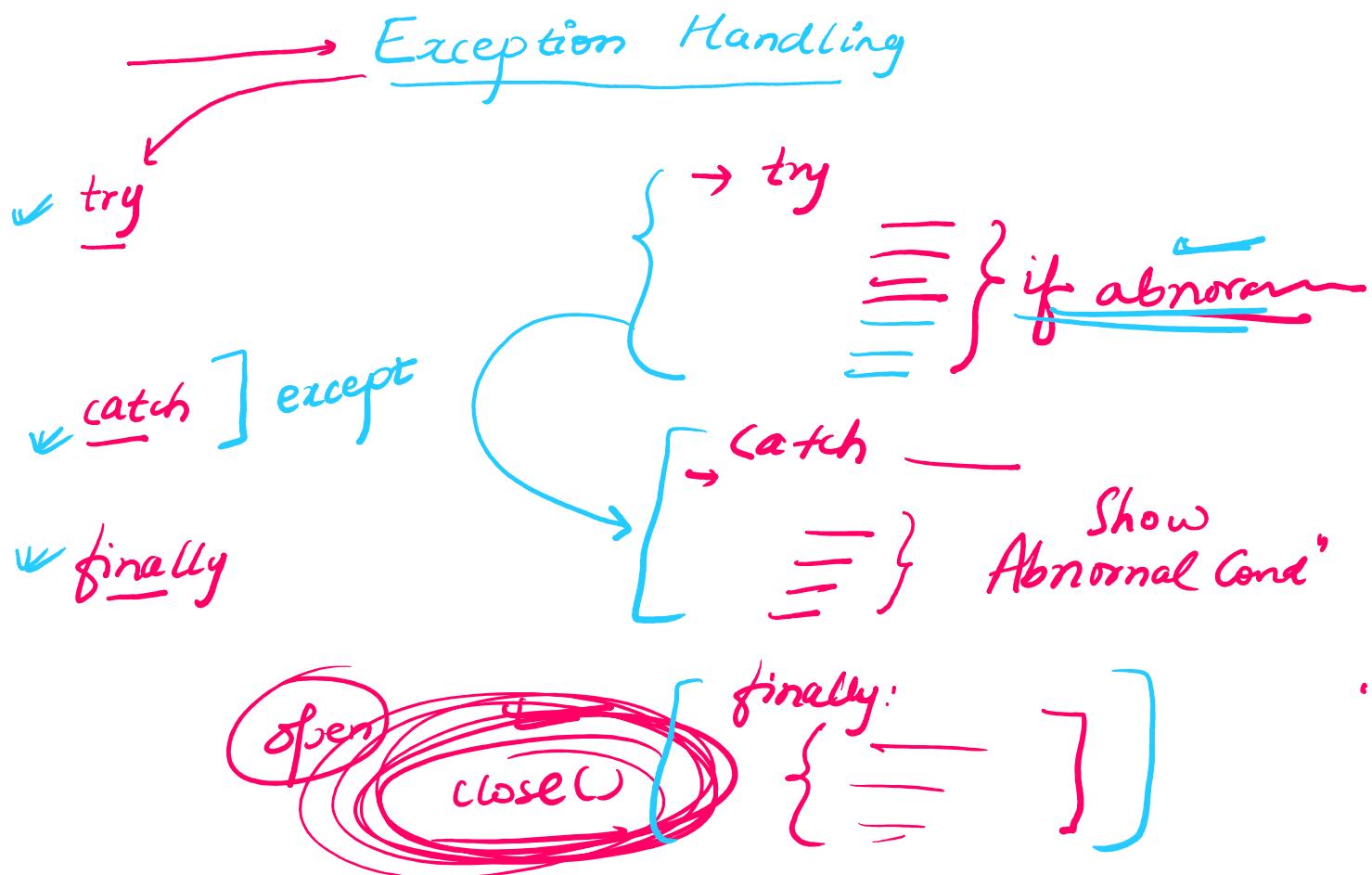
→ file

→ list

→ var / Exception

D.S :-





NOTE:

- 1) try always require catch/except block
- 2) finally block will always execute
 ↓
 Error No Error
- * 3) catch/except will only run when exception is raised/occurred

Robust :- → Conclusion → Robustness

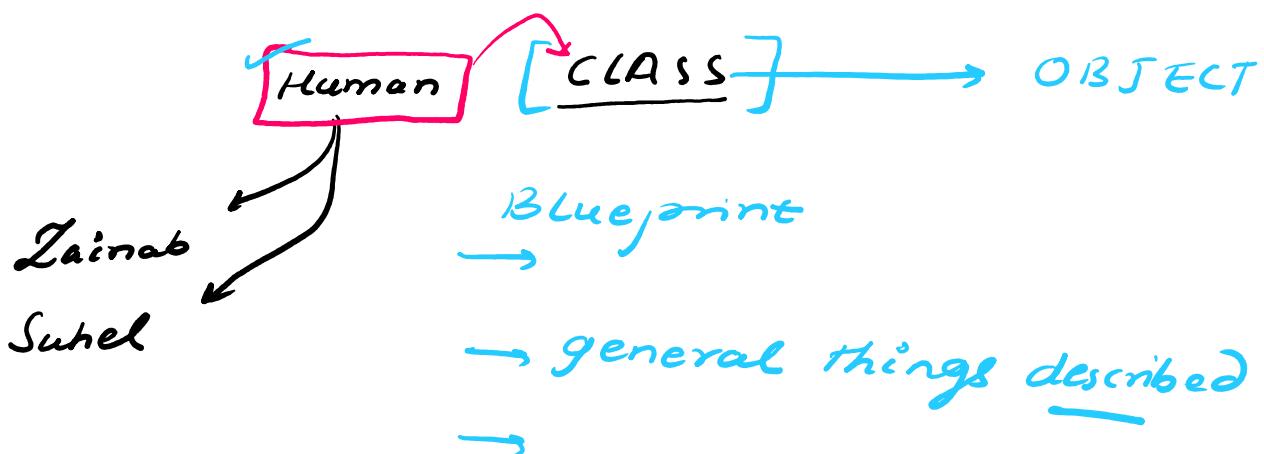
Resilient program
Powerful program

Object Oriented Programming

What is OBJECT?

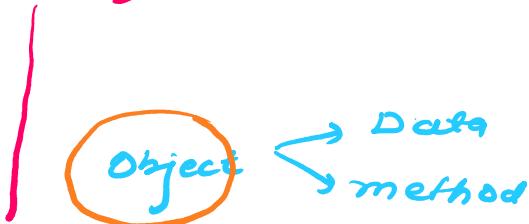
Limitation {
 → variable
 → function

- State (height, eyes, ears...) → variable (Data member)
- Behavior (walk, eat, study...) → function (member method)



Class :- Memory Allocated

↳ Data
 ↳ method



Memory Allocate

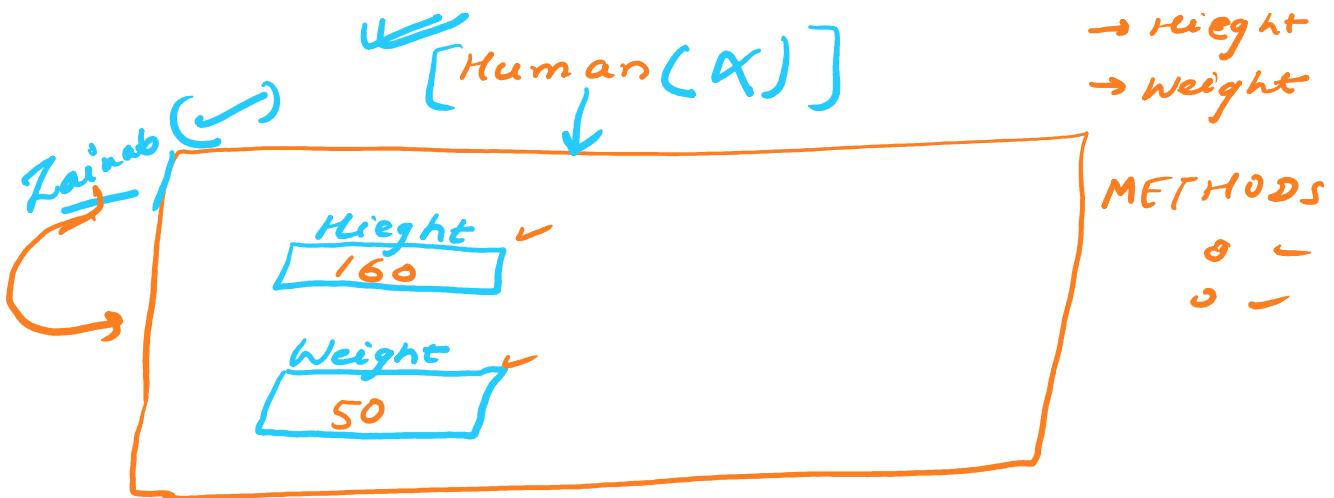
variable
 ↳ Height



container

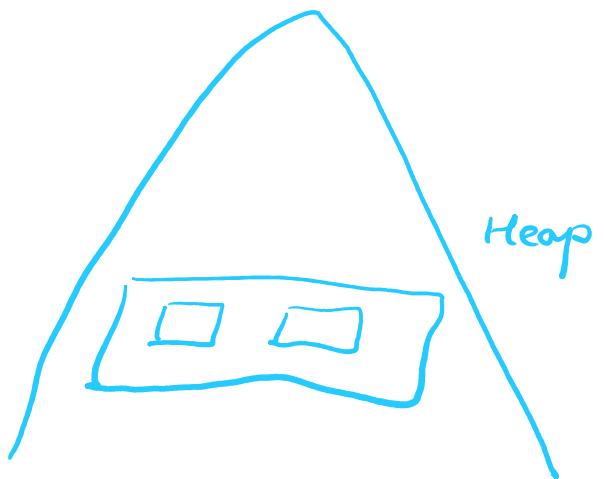
DATA
 → height
 → weight

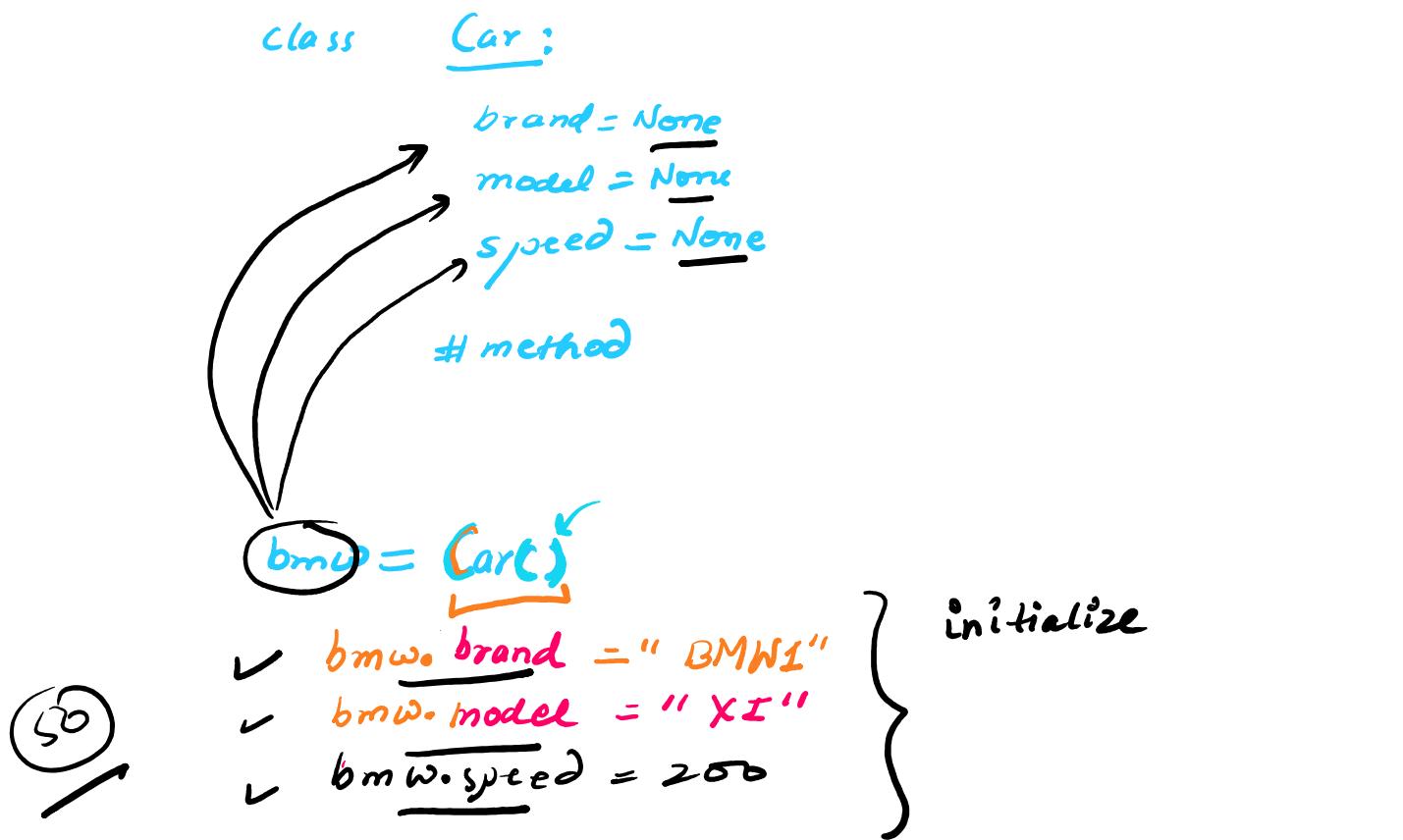
→ [Human(X)]



Mentoring :-

{ Course
Paid } } mentoring
 ↓
 1/2





Constructor :-

→ method

↳ return & value

↳ Object initializer

→ Car() ✓

→ Default constructor

def __init__(self, brand, model, speed):

self.brand = brand
self.model = model
self.speed = speed

fun ≈ method

Self :-

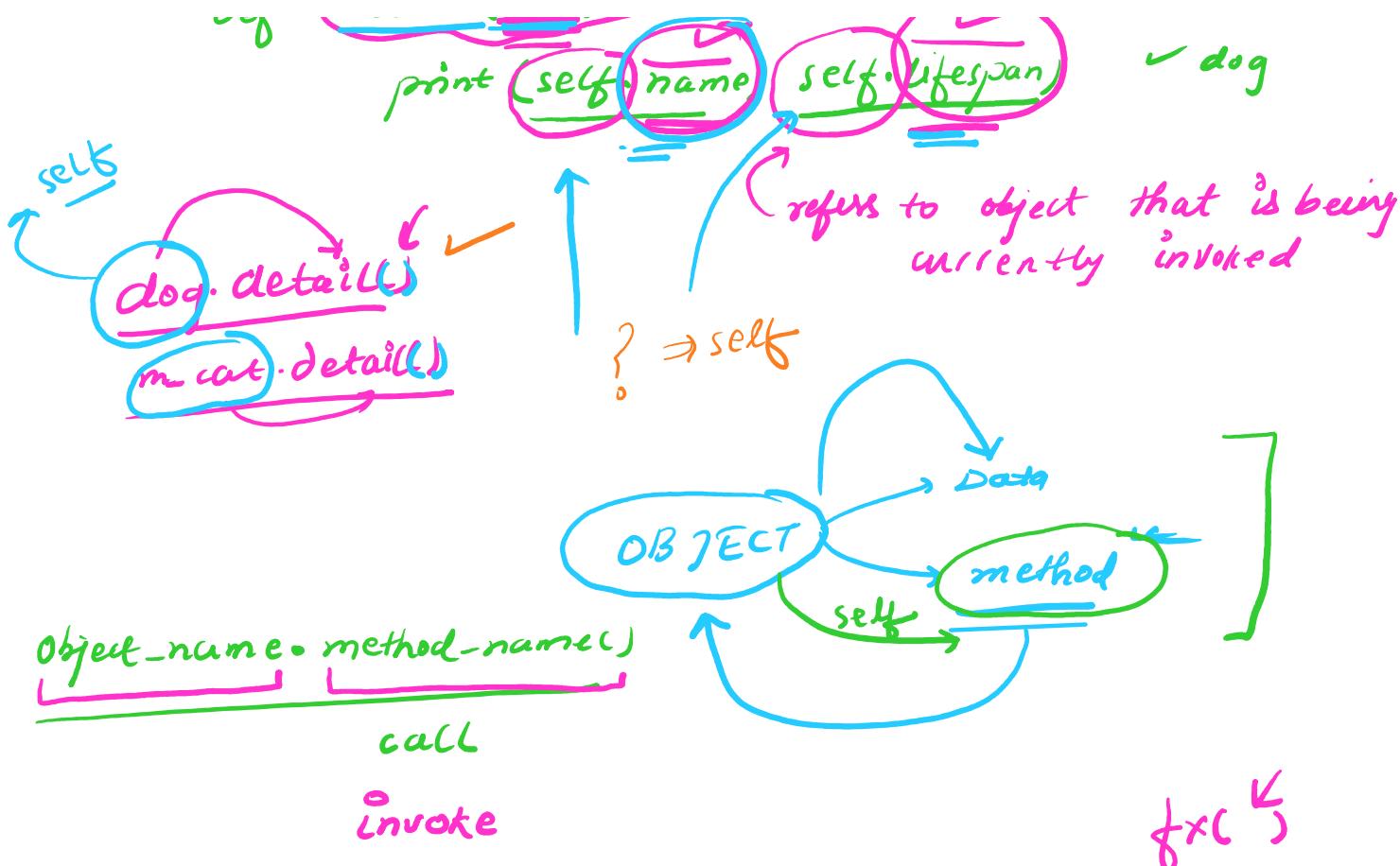
def detail(self):

print(self.name) self.lifespan

→ keyword

→ special

→ munchin
 → catez
 → dog



method
`obj.meth()`

No Difference

Except
in vocation

f(xn)
`b(x_1)`

Pillars of OOPs

- Encapsulation
- Inheritance
- Polymorphism
- Abstraction

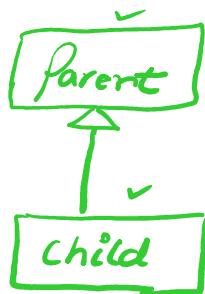
member
`(data + method)`

var attrib
datamember

f(xn)
member method

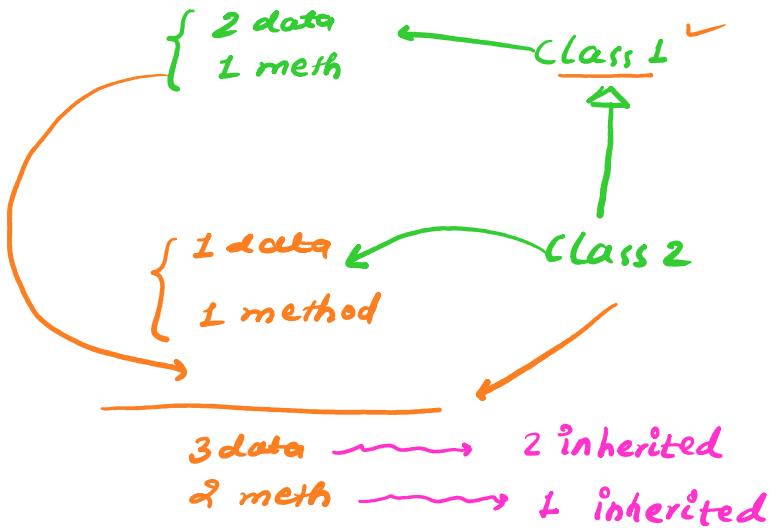
Inheritance

07 May 2025 11:36



Inheritance

OOPs



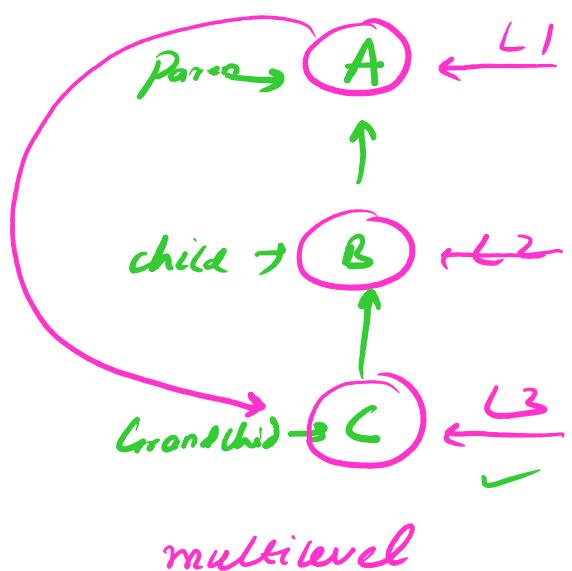
Polymorphism

Having multiple behavioral form

- 1) Runtime time polymorphism → overriding
- 2) Compile time polymorphism → overloading

Inheritance Types:-

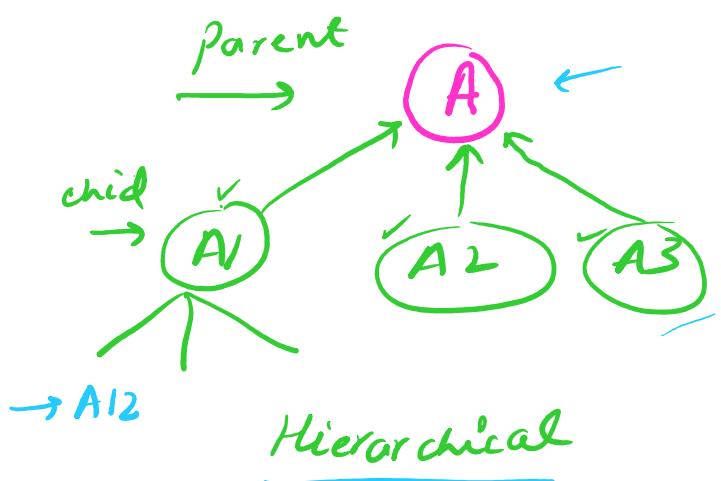
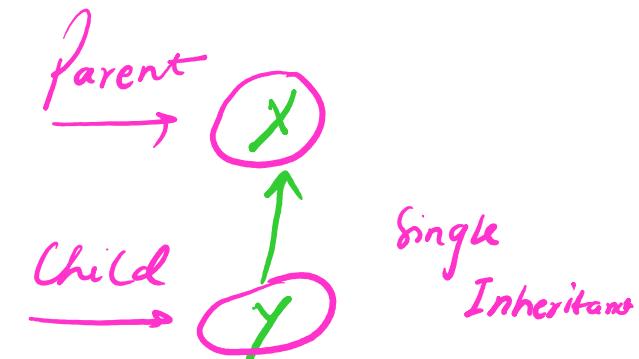
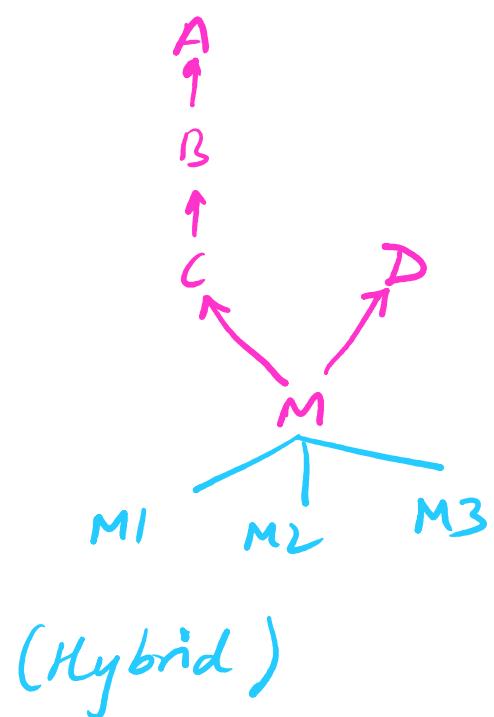
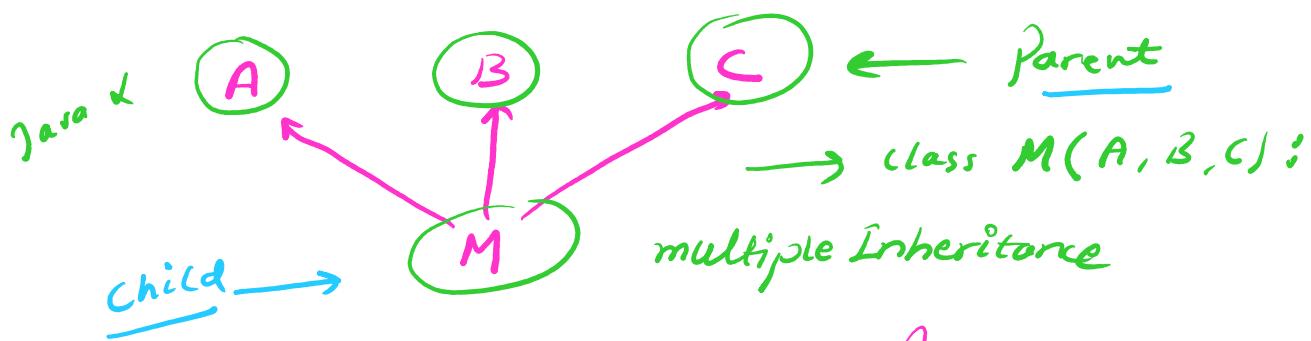
- 1) multilevel Inheritance
- 2) multiple "
- 3) Single "
- 4) Hierarchical "
- 5) Hybrid "

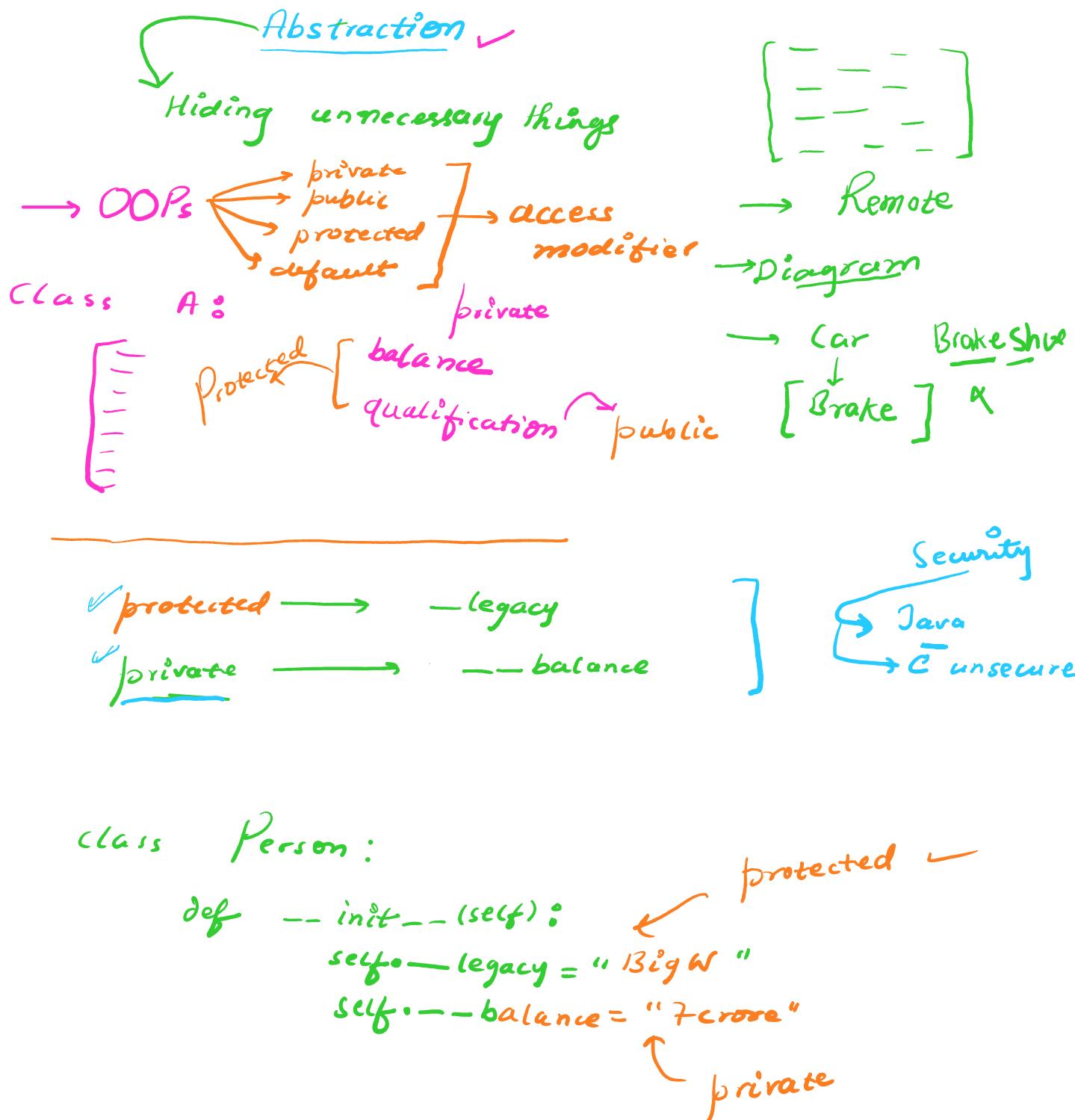


L (A)

(B)

(C) ← Parent





Revision

10 May 2025 11:37

Revision

- Programming / coding *
- Prog. lang. (Python)
- Variables / Data types
- control structures ↗ if else
match
loops
- Data structures (list, dictionaries)
- functions

- File handling
- Exception handling
- Todo project
- OOPS
 - ↳ Encapsulation
 - ↳ Inheritance
 - ↳ Abstraction
 - ↳ Polymorphism

(Beginner → Intermediate)

- multithreading *

HTML, CSS, SASS,
Development

- Networking

Mathematics
(Logics)

→ DSA

[Data Science]
→ AI

- Libraries

Dukaan*
Amazon, google,
Microsoft*, APP
(Product based company)
Algorithm

(Advanced level)



4

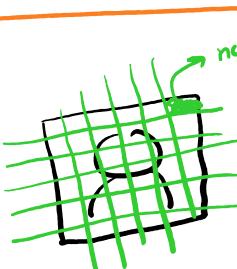
Exception
(Errors)

→ Abnormal
Zero division

→ Maximum depth reached
div → a/b ↗ $b=0$

ALU → Arithmetic logic unit

ASCII



matrix

list []

Cal^n

Revision (Go through)

NY-Yorktown *

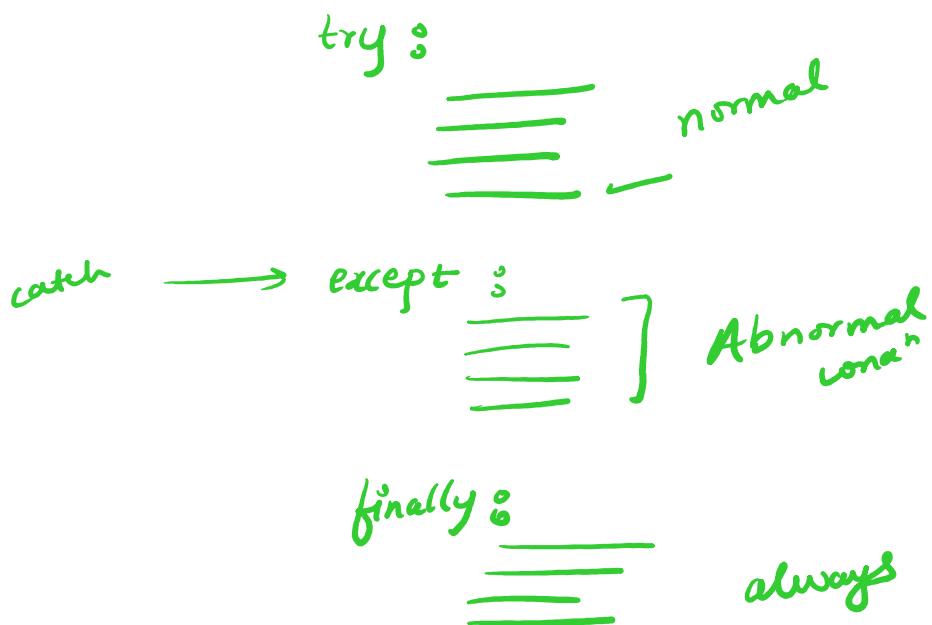
Coding

o Division

→ 0 Zero Division

* Numbers *

If
Sink.



Programming / Coding

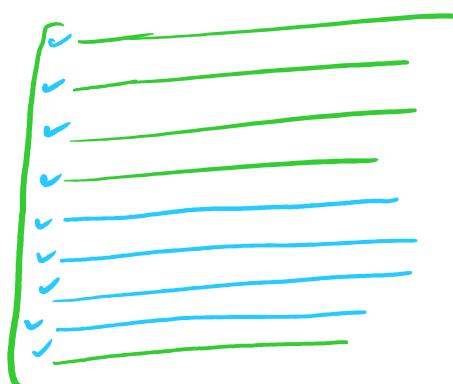
Bigger term

small term

Solving Problem using
prog lang (computer)

Todo =
create if
[]

name = bob



Program (Passive)
Lifeless



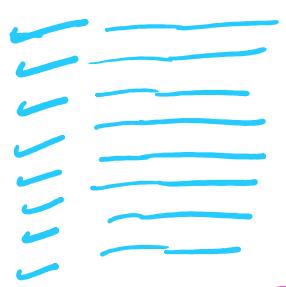
LOC → line of code

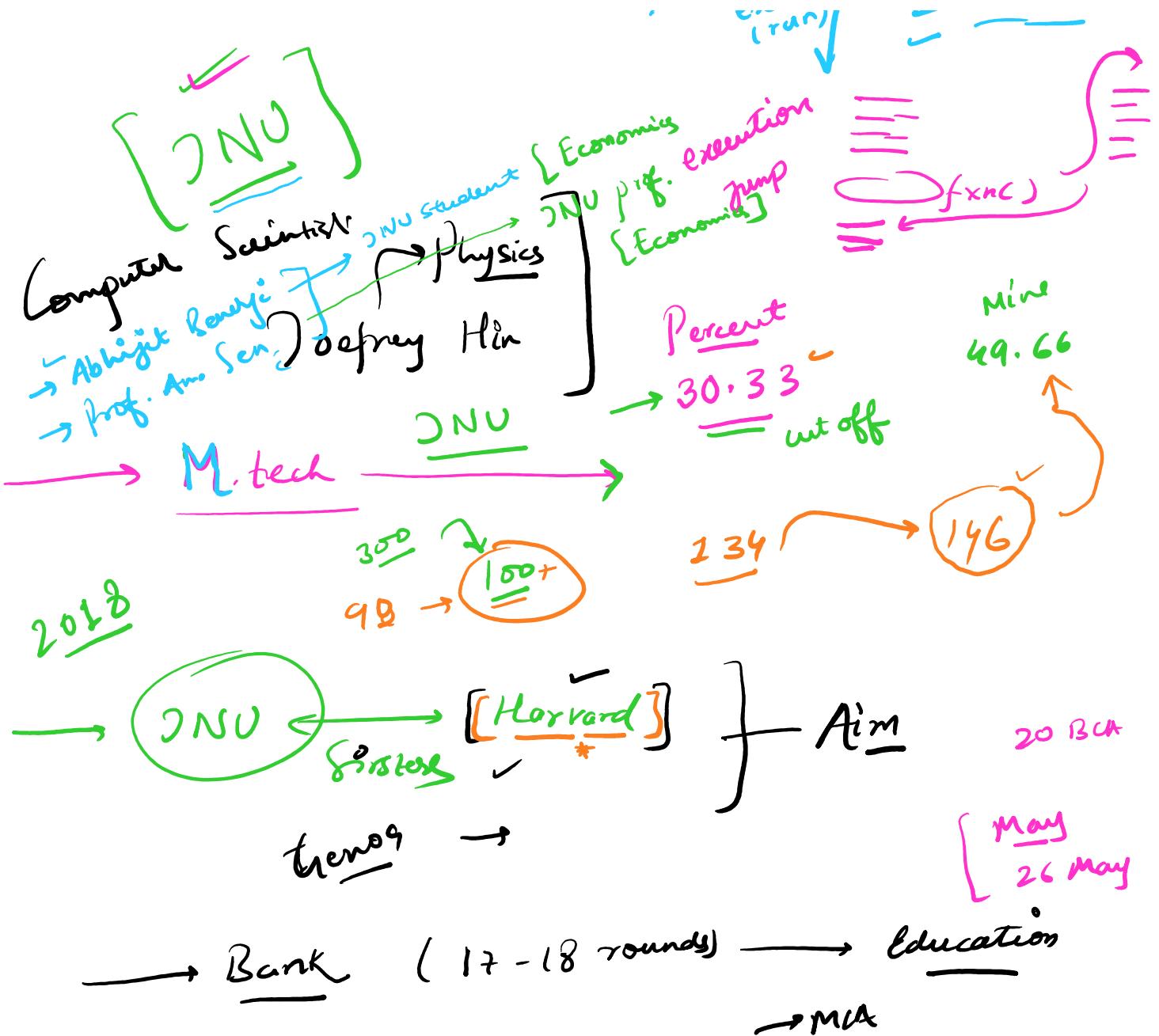
Prog "OR"
machine code

OS

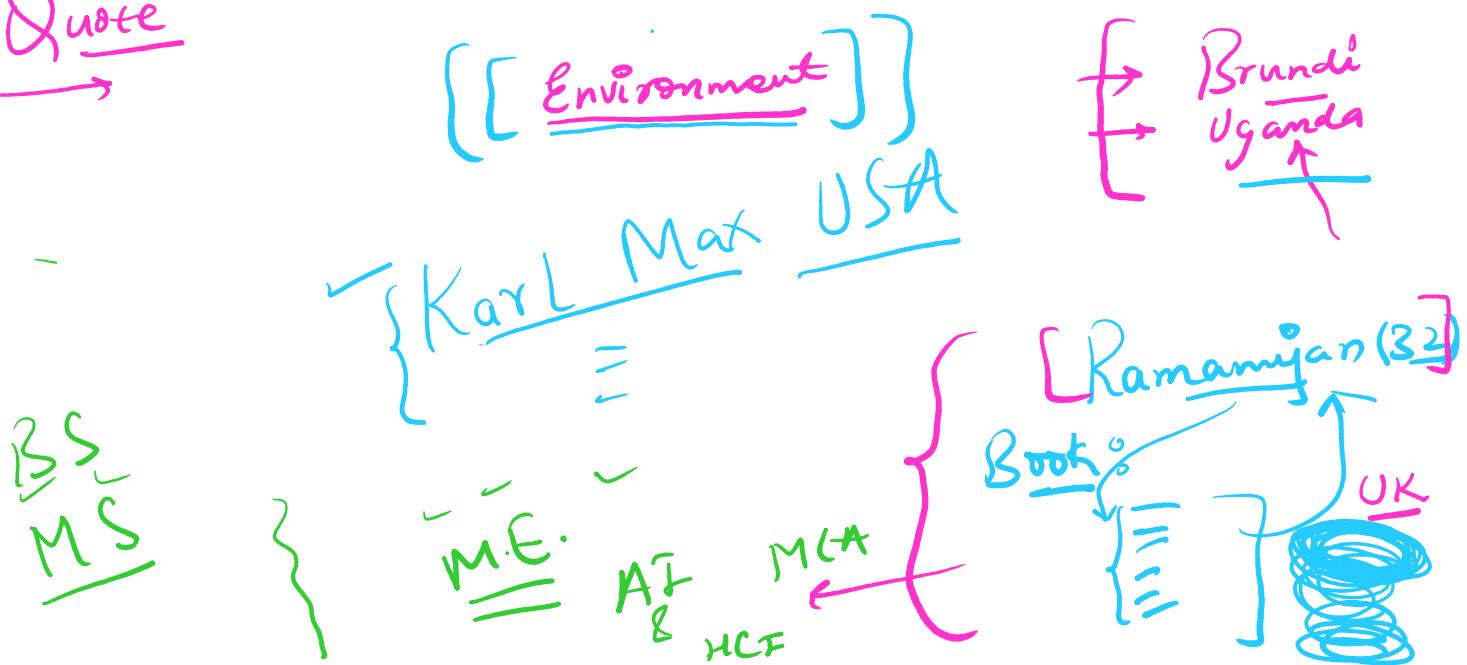
(Active) Process

sequence execution
(run)

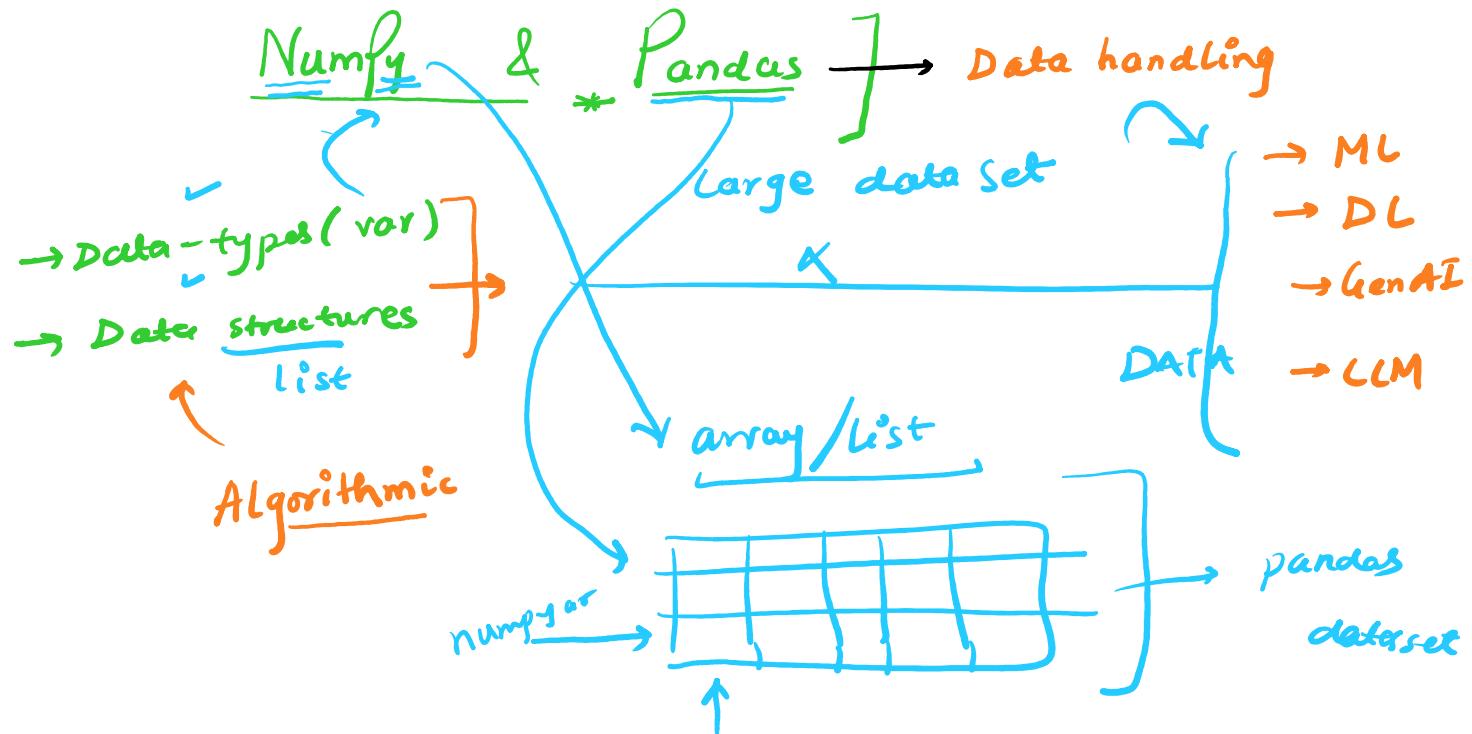


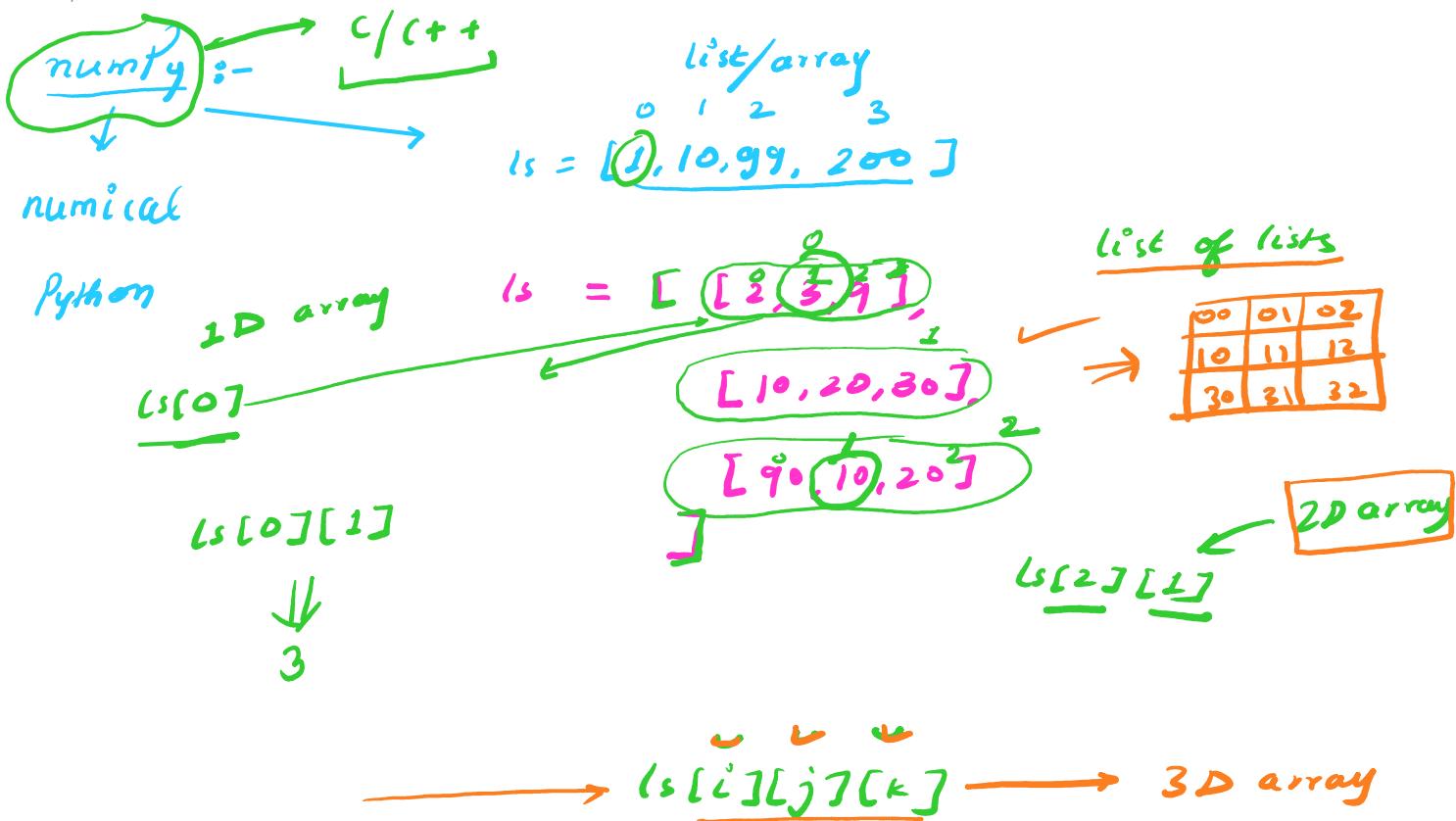


Quote



Independent  choice
circumstance/stance

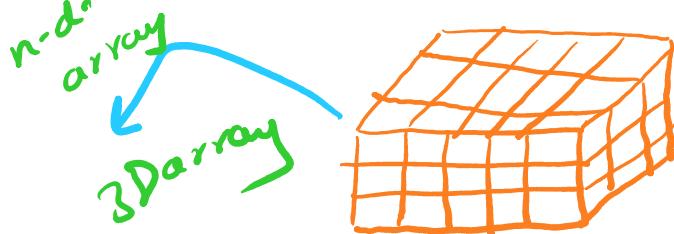




ML/AI

Linear Algebra

↔ n-d space dimensionality n

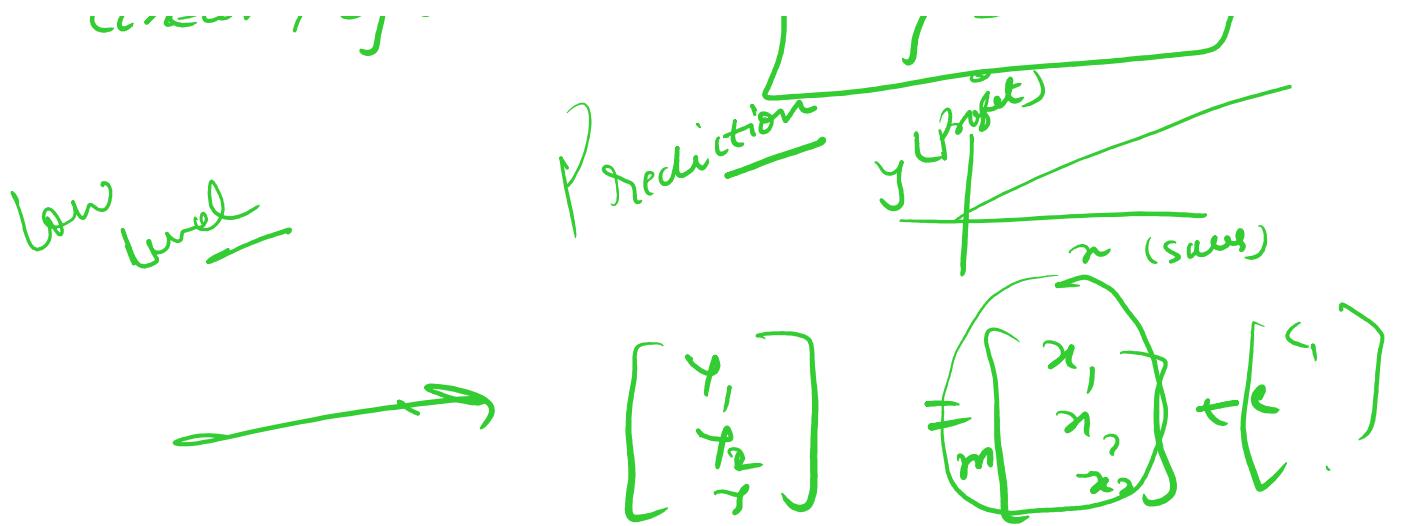


numpy → advance Py library

↳ n-d array

Linear Regression

$$y = mx + c$$



Object X
 C/C++
 Primitive data types

class 'str'
 class 'int'

1 byte = 8 bit
 1 bit \Rightarrow 0/1 ✓

2 byte

4 byte \leftarrow (float) $f = 10.5;$

$a = 5$

int $a = 5;$

1b = 8 bit
 999

✓ 0 0 0 | 0
 ✓ 1 1 1 | 1

0 0 0 | 0
 1 1 1 | 1

$8 \times 8 = 64$

0 to

255

1 1 1 1 1 1 1 1

2^{64}

↑

✓ Statically typed lang.
 C/C++/Java
 int $a = 5$

Dynamically typed lang.

$a = 5$ float
 $a = 5.2$

0 \Rightarrow 0 0 0 0 0 0 0 0

1

$0 \Rightarrow 0000000000$
 $1 \Rightarrow 0000000001$

① bool $\Rightarrow 1\text{ byte}$

② int $\Rightarrow 8\text{ byte}$ ~~4 byte~~

③ intc \Rightarrow

④ intp

⑤ float

⑥ double

⑦ complex

Code

[[Slicing]]

Tomorrow

{ 9:50
11:30 }

Python

IQ
↔

Chatgpt

Project

Chatgpt

Fundamentals

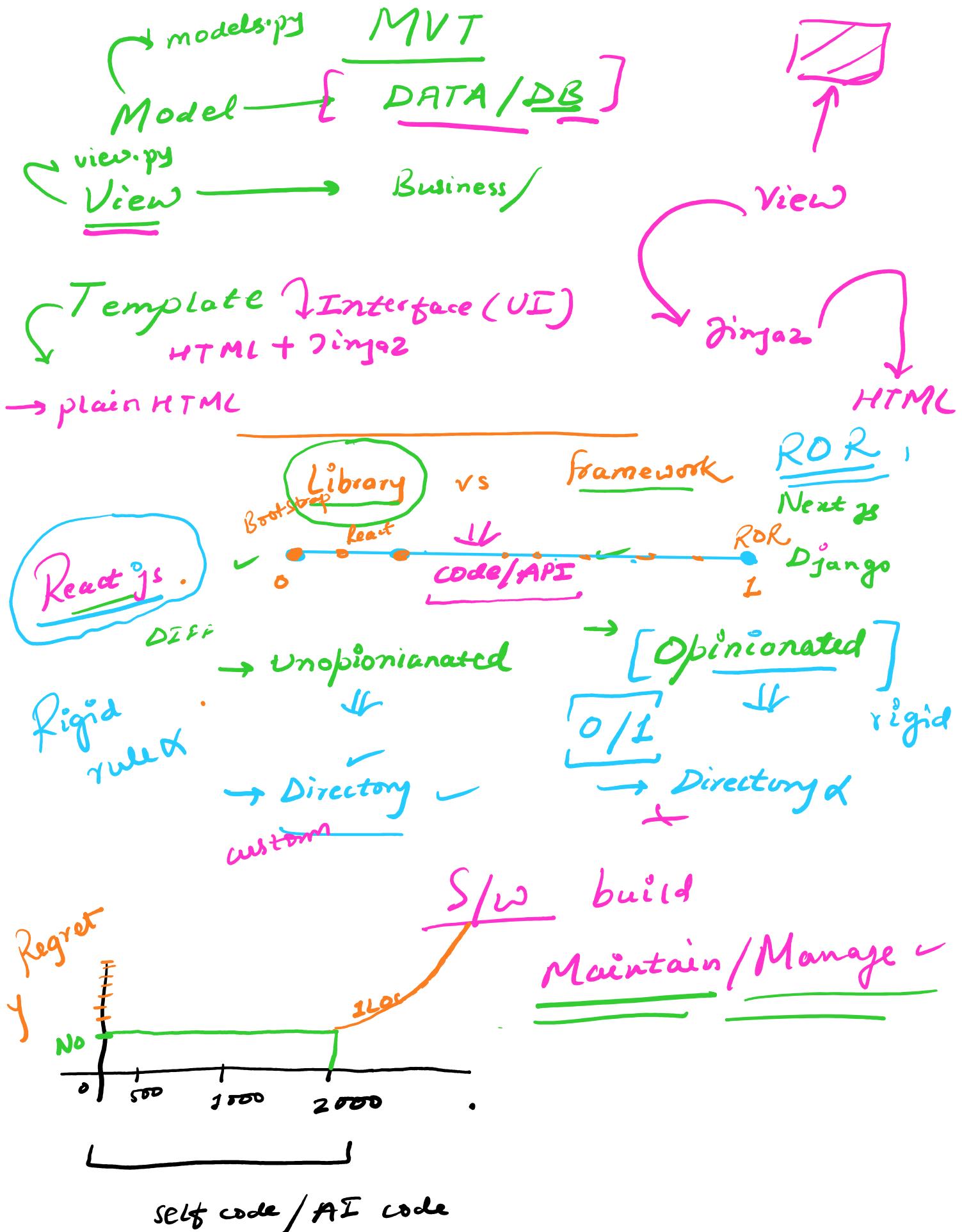
{ 10-20 years }

[GNU]

Where there is a will,
there is a way !

→ 6-7pm

there is a way !



Slicing : np array

18 May 2025 1:36

$$\underline{\text{arr}} = [0, 1, 2, \textcircled{3}, 4]$$

$a_1 = \underline{\text{index}(1 \text{ to } 3)}, \underline{\text{or} \text{Inx}(2, 4)}$



include not included

$\rightarrow a_1 = \text{arr}[1:4]$

$\rightarrow a_2 = \text{arr}[2:5]$

$A[i]$

$A[i:j]$

	0	1	2	3
0				
1				
2				

$\Rightarrow A[\text{row}][\text{col}:]$
↓
 $A[\text{row}, \underline{\text{col}}]$

$a_1 = A[1:3, 1:3]$

	col 1	col 2	col 3	.
row 1				
2				
3				
4				

ML models
↓

[numpy array]

Linear Algebra

	0	1	2	3
row 0	35	10	21	29
1	40	10	22	9
2	55	4	2	6

L.W

$\text{Array}[1:2][1:3]$