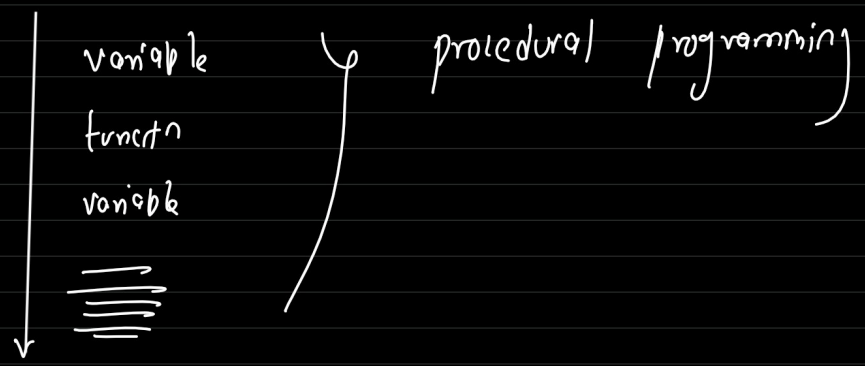
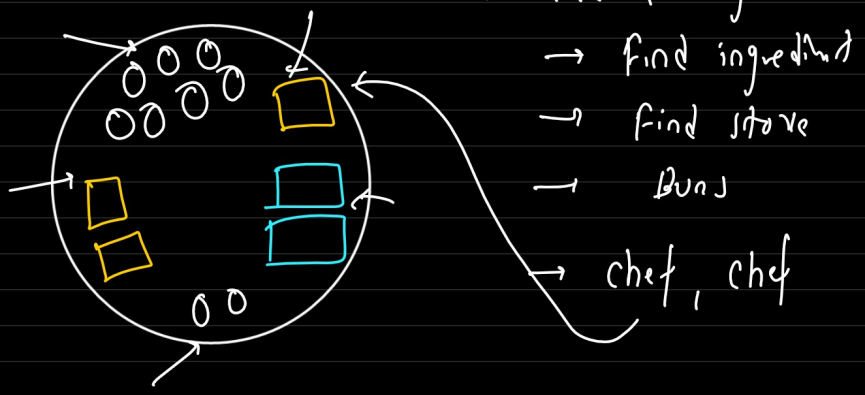
04-02-2026

Agenda:

→ Object oriented programming (OOPs)



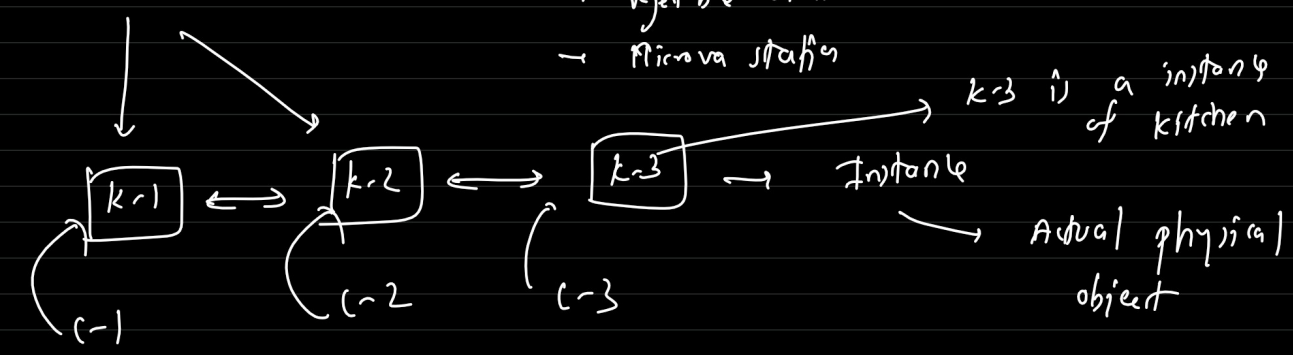
Resturant:



Resturant:

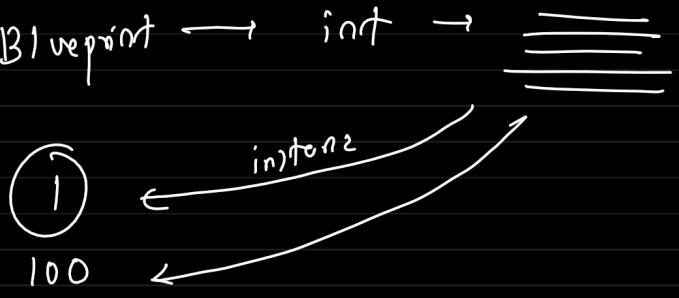
Blueprint: kitchen → on paper.

Main
→ Burger station
→ vegetable station
→ Microwave station



OOP → class → Blueprint
→ object
instance

Blueprint → int →



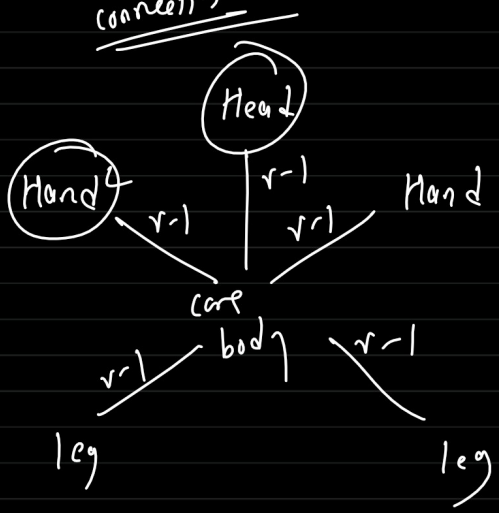
Task: To create a robot.

procedural progr:

→ Head
→ Hands
→ Legs
→ Core body

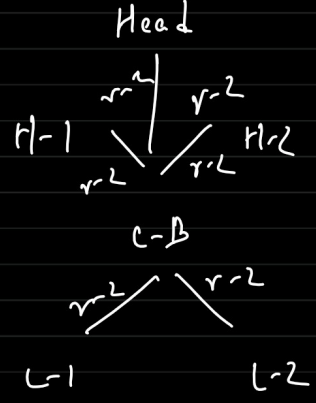
Task: To move 5 m

connection



r-1

→ same connection



class Robot :

def __init__ ( ):

constructor

start → name _
→ hands _
→ legs _

→ auto start

initialize some important variables

def move ( ):

v-1

Robot

def (self):

def (self):

self → tell class about its own body / connection

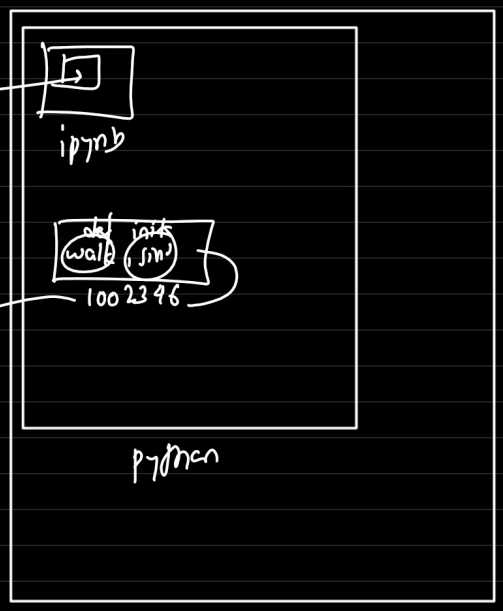
cls

→ name,

```
# blue print
class Robot:
    def __init__(a):
        print("Hello, welcome to robot!.")

    def walk(a):
        print("Walking 10 meters.")

    def sing(a):
        print("Singing a song.")
```



Blueprint

address of 3, stored in ram

16 gb ram

value_1 = 3 → instance of class int

value_1 = Robot() → OS → to allocate memory for the class components

class_name

```
def __init__()
def walk() → a,b,c
def sing()
```

a = [1, 2, 3, 4]

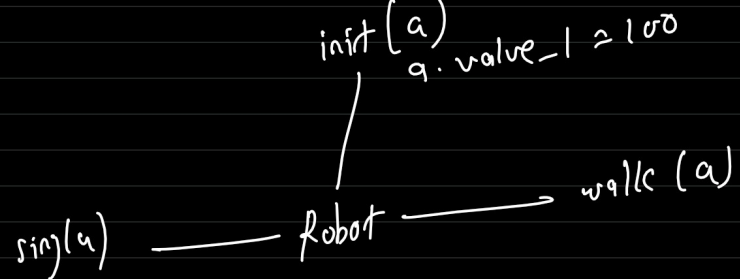
object is an instance of a class

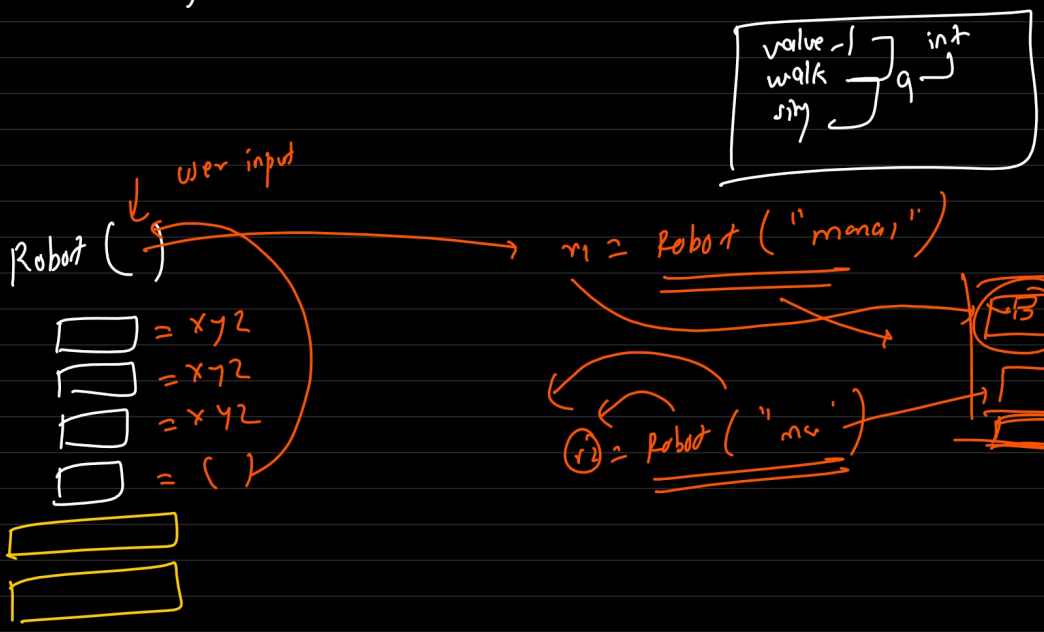
↓

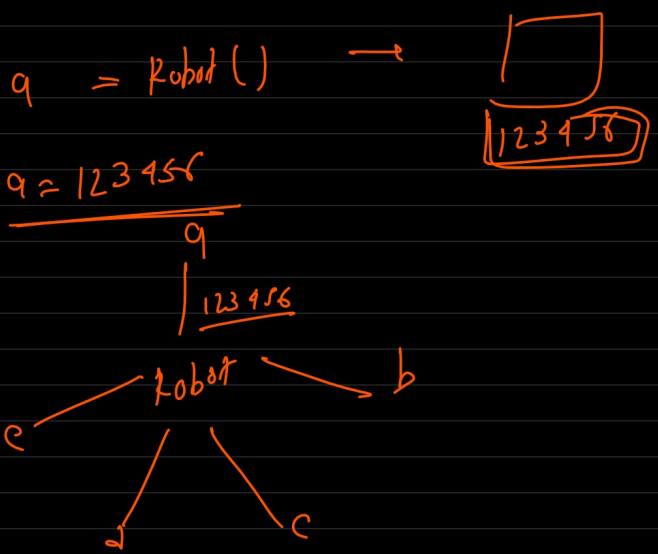
actual data / physical data stored in ram as a package

function inside a class is called as method.

q → attribute → . → value / function







Naming convention :—

class → PascalCase : DataScience, Robot

methods / variable : snake_case : user_account_info, customer_name