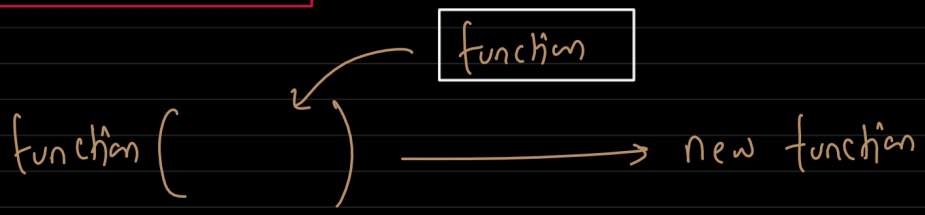
11-02-2026

Agenda :

1. **Decorators**
2. **Static variables and methods (OOP)**

Decorators : It is a function that takes another function as an argument and return a new function. It is used to enhance the functionality of the base function.



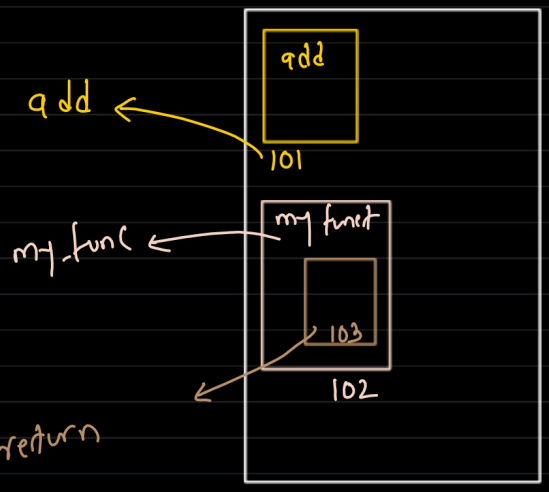

```
def add ( )
    a=1
    ① → return should be present here
    return a
②
```

```
def run ( func ) :
    def wrap ( ) :
        print ( )
        func ( )
        print ( )
    ②   ①
③
```

return wrap → ①
return run → ②

```
def add():
    print("Addition Function")

def my_func(func):

    def wrap():
        print("Start")
        func()
        print("End")
    return wrap

add = my_func(add)

add()
```

→ definition →



add → 101
func → 101

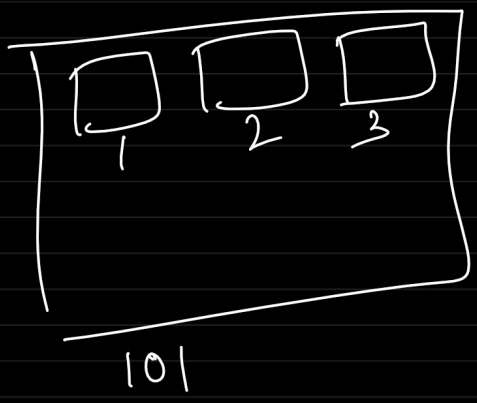
my_funct (func)

def wrap( ) :
print (" ")
func( )
print( )
return wrap → 103 → print func() print

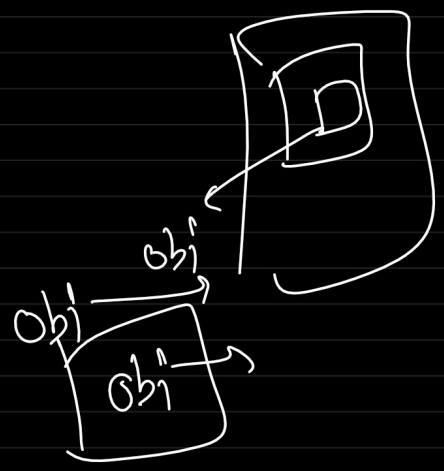
add = my_funct (add) → 101

address of a function called as wrap → 103

add( ) →



101.1
101.2



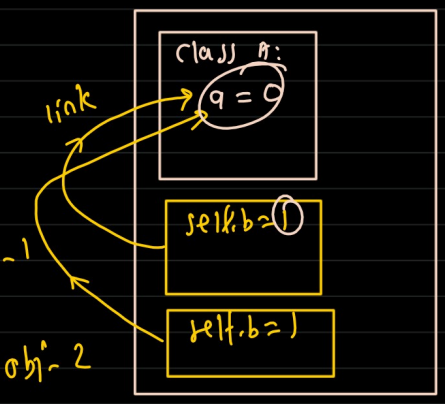
Static

Static = belong to the class, not an object

* shared across all object
* created once
* not tied to any instance

static variable ← Class A:
a = 0
self.b = 1

→ no self.



obj-1 = A()

obj-2 = A()

static variable ← can be accessed by class name directly

← standard method to access static
→ via class name

optional → object