

Nama: Agung Sihab Malawi

Kelas: R1

NIM: 210511047

PBO2 Latihan 4

### Contoh 1

Script:

```
class Pekerja:
    def __init__(self, nama, umur):
        self.nama = nama
        self.umur = umur

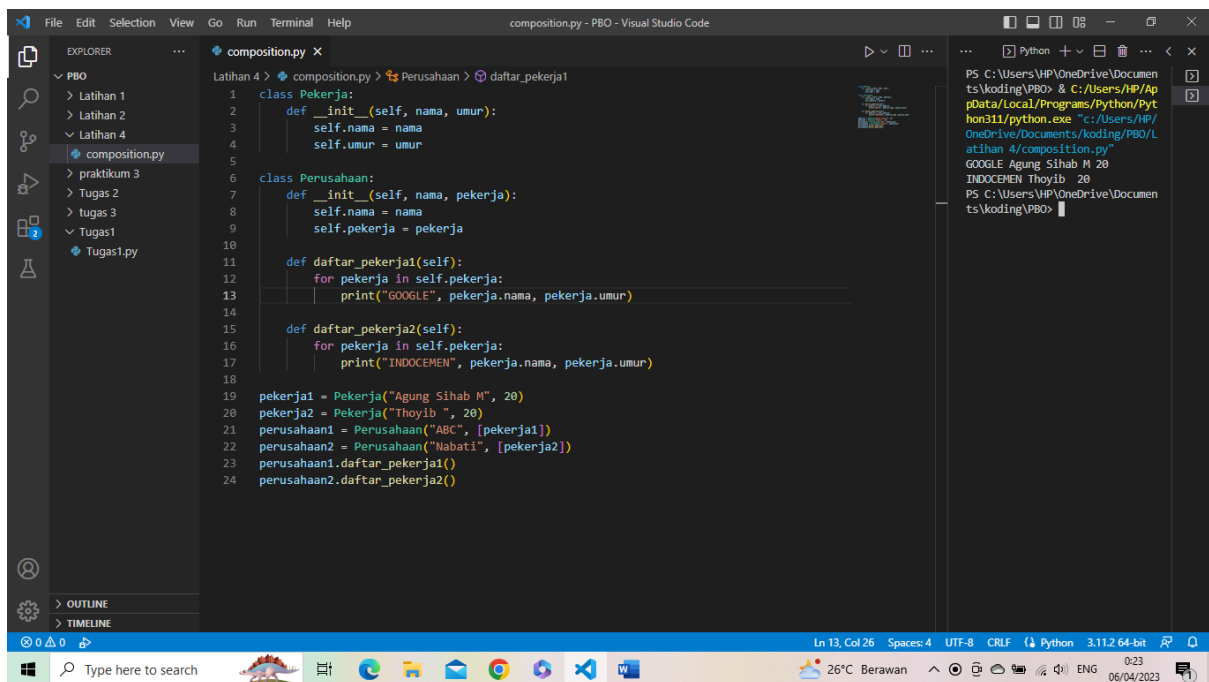
class Perusahaan:
    def __init__(self, nama, pekerja):
        self.nama = nama
        self.pekerja = pekerja

    def daftar_pekerja1(self):
        for pekerja in self.pekerja:
            print("GOOGLE", pekerja.nama, pekerja.umur)

    def daftar_pekerja2(self):
        for pekerja in self.pekerja:
            print("INDOCEMEN", pekerja.nama, pekerja.umur)

pekerja1 = Pekerja("Agung Sihab M", 20)
pekerja2 = Pekerja("Thoyib ", 20)
perusahaan1 = Perusahaan("ABC", [pekerja1])
perusahaan2 = Perusahaan("Nabati", [pekerja2])
perusahaan1.daftar_pekerja1()
perusahaan2.daftar_pekerja2()
```

Hasil Running Program :



The screenshot shows a Visual Studio Code editor window with a file named 'composition.py' open. The code defines two classes: 'Pekerja' and 'Perusahaan'. 'Pekerja' has an '.\_\_init\_\_' method that takes 'nama' and 'umur' as arguments. 'Perusahaan' has an '.\_\_init\_\_' method that takes 'nama' and 'pekerja' as arguments. It also has two methods: 'daftar\_pekerja1' and 'daftar\_pekerja2', both of which iterate over the 'pekerja' list and print the company name, employee name, and age. The script creates two 'Pekerja' objects, 'pekerja1' and 'pekerja2', and two 'Perusahaan' objects, 'perusahaan1' and 'perusahaan2'. It then calls 'daftar\_pekerja1' on 'perusahaan1' and 'daftar\_pekerja2' on 'perusahaan2'.

```
1 class Pekerja:
2     def __init__(self, nama, umur):
3         self.nama = nama
4         self.umur = umur
5
6 class Perusahaan:
7     def __init__(self, nama, pekerja):
8         self.nama = nama
9         self.pekerja = pekerja
10
11     def daftar_pekerja1(self):
12         for pekerja in self.pekerja:
13             print("GOOGLE", pekerja.nama, pekerja.umur)
14
15     def daftar_pekerja2(self):
16         for pekerja in self.pekerja:
17             print("INDOCEN", pekerja.nama, pekerja.umur)
18
19 pekerja1 = Pekerja("Agung Sihab M", 20)
20 pekerja2 = Pekerja("Thoyib ", 20)
21 perusahaan1 = Perusahaan("ABC", [pekerja1])
22 perusahaan2 = Perusahaan("Nabati", [pekerja2])
23 perusahaan1.daftar_pekerja1()
24 perusahaan2.daftar_pekerja2()
```

The terminal output shows the following:

```
PS C:\Users\HP\OneDrive\Documents\koding\PBO> & C:/Users/HP/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/HP/OneDrive/Documents/koding/PBO/Latihan 4/composition.py"
GOOGLE Agung Sihab M 20
INDOCEN Thoyib 20
PS C:\Users\HP\OneDrive\Documents\koding\PBO>
```

Contoh 2

Script:

```
class Player:
    def __init__(self, name):
        self.name = name
        self.inventory = Inventory()
        print("Hero Layla")

class Item:
    def __init__(self, name):
        self.name = name

class Inventory:
    def __init__(self):
        self.items = []

    def add_item(self, item):
        self.items.append(item)
        print("Item", item.name)

    def remove_item(self, item):
        self.items.remove(item)
```

```

player = Player("Hero Layla")
sword = Item("Blade Of Despair (BoD)")
shield = Item("Radiant Armor")

print("="*40)
player.inventory.add_item(sword)
player.inventory.add_item(shield)
player.inventory.items
print(" ")

```

Hasil Running Program:

Contoh 3

Script:

```

class Menu:
    def __init__(self, dishes=None):
        if dishes is None:
            self.dishes = []
        else:
            self.dishes = dishes
    def add_dish(self, dish):
        self.dishes.append(dish)
        print("Menu", dish.name, dish.price)

class Dish:
    def __init__(self, name, price):

```

```

        self.name = name
        self.price = price
        self.menu = Menu()

class Restaurant:
    def __init__(self, name, menu):
        self.name = name
        self.menu = menu
        print("Warung Padang")

dish1 = Dish("Nasi Komplit", 15000)
dish2 = Dish("Mie Goreng", 12000)
menu = Menu([dish1, dish2])
restaurant = Restaurant("Warung Padang", menu)
print("="*40)
restaurant.menu.add_dish(dish1)
restaurant.menu.add_dish(dish2)
restaurant.menu.dishes

```

Hasil Running Program:

```

File Edit Selection View Go Run Terminal Help
composition3.py - PBO - Visual Studio Code

EXPLORER
PBO
  > Latihan 1
  > Latihan 2
  > Latihan 4
  composition.py
  composition2.py
  composition3.py
  praktikum 3
  > Tugas 2
  > tugas 3
  > Tugas1
  Tugas1.py

composition3.py
1 class Menu:
2     def __init__(self, dishes=None):
3         if dishes is None:
4             self.dishes = []
5         else:
6             self.dishes = dishes
7     def add_dish(self, dish):
8         self.dishes.append(dish)
9         print("Menu", dish.name, dish.price)
10
11 class Dish:
12     def __init__(self, name, price):
13         self.name = name
14         self.price = price
15         self.menu = Menu()
16
17 class Restaurant:
18     def __init__(self, name, menu):
19         self.name = name
20         self.menu = menu
21         print("Warung Padang")
22
23 dish1 = Dish("Nasi Komplit", 15000)
24 dish2 = Dish("Mie Goreng", 12000)
25 menu = Menu([dish1, dish2])
26 restaurant = Restaurant("Warung Padang", menu)
27 print("="*40)
28 restaurant.menu.add_dish(dish1)
29 restaurant.menu.add_dish(dish2)
30 restaurant.menu.dishes

Output
PS C:\Users\VHP\OneDrive\Documents\koding\VPBO> & C:/Users/HP/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/VHP/OneDrive/Documents/koding/VPBO/Latihan 4/composition3.py"
Warung Padang
=====
Menu Nasi Komplit 15000
Menu Mie Goreng 12000
PS C:\Users\VHP\OneDrive\Documents\koding\VPBO>

```

## Contoh 4

```

class Song:
    def __init__(self, title, artist):
        self.title = title
        self.artist = artist

```

```

class Playlist:
    def __init__(self):
        self.songs = []

    def add_song(self, song):
        self.songs.append(song)
        print("Title", song.title)

class MediaPlayer:
    def __init__(self, playlist):
        self.playlist = playlist
        print("Play Music")

song1 = Song("Terima Kasih", "HAL")
song2 = Song("Aku Dan Ceritaku", "Icong")
playlist = Playlist()
media_player = MediaPlayer(playlist)
print("="*40)
playlist.add_song(song1)
playlist.add_song(song2)
media_player.playlist.songs

```

Hasil Running Program:

## Contoh 5

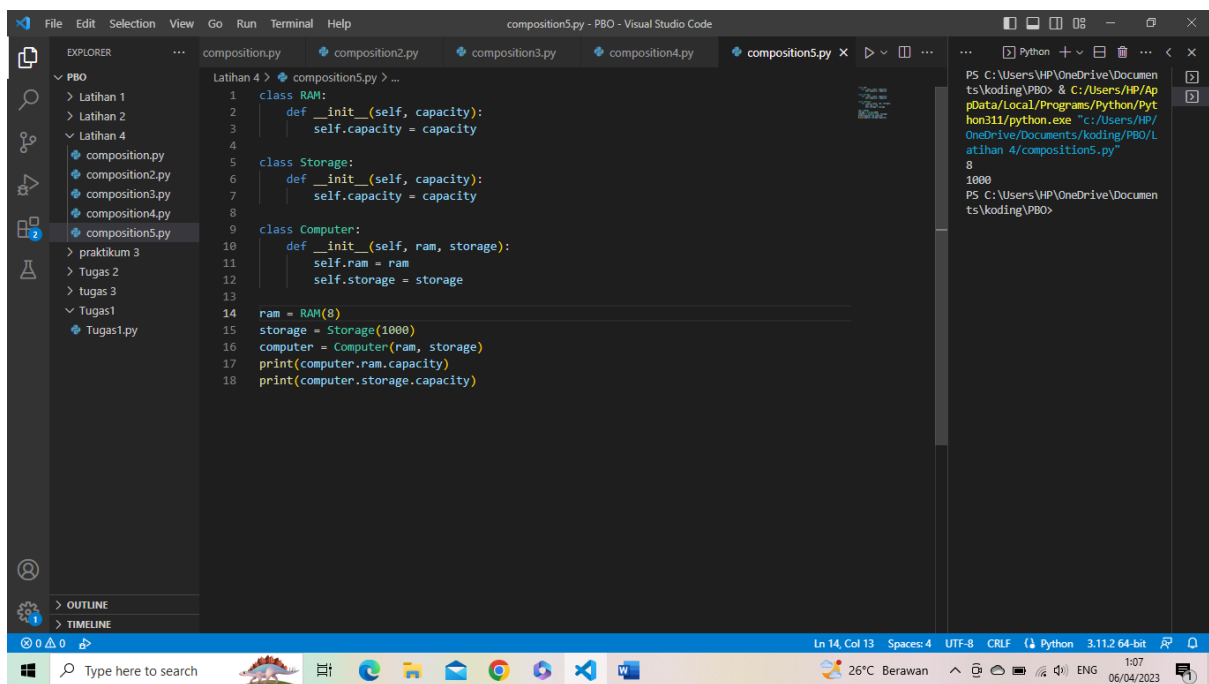
```
class RAM:
    def __init__(self, capacity):
        self.capacity = capacity

class Storage:
    def __init__(self, capacity):
        self.capacity = capacity

class Computer:
    def __init__(self, ram, storage):
        self.ram = ram
        self.storage = storage

ram = RAM(8)
storage = Storage(1000)
computer = Computer(ram, storage)
print(computer.ram.capacity)
print(computer.storage.capacity)
```

Hasil Running Program:



## Contoh 6

```
class Wheel:
    def __init__(self, size):
        self.size = size

class Engine:
    def __init__(self, power):
        self.power = power

class Car:
    def __init__(self, wheels, engine):
        self.wheels = wheels
        self.engine = engine

wheel1 = Wheel(17)
wheel2 = Wheel(17)
wheel3 = Wheel(17)
wheel4 = Wheel(17)
engine = Engine(150)
car = Car([wheel1, wheel2, wheel3, wheel4], engine)
print(car.wheels[0].size)
```

### Hasil Running Program:

