

LAPORAN PRAKTIKUM

PEMROGRAMAN BERORIENTASI OBJEK LANJUT

2023



Prepared By:

Nama : IMANDA

NIM : 210511089

KELAS : TIF21B (R2)

Buatlah masing-masing 2 contoh jenis pewarisan di luar dari contoh yang telah diberikan, beri nama:

1. `single1.py`

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

    def berbunyi(self):
        print(self.nama, "berbunyi")

class Burung(Hewan):
    def __init__(self, nama, umur, jenis_bulu):
        super().__init__(nama, umur)
        self.jenis_bulu = jenis_bulu

    def bersuara(self):
        print("Merdu!")

kucingA = Burung("Chelsi", 2, "Merdu")
kucingA.berbunyi()
kucingA.bersuara()
```

OUTPUT SS

2. single2.py

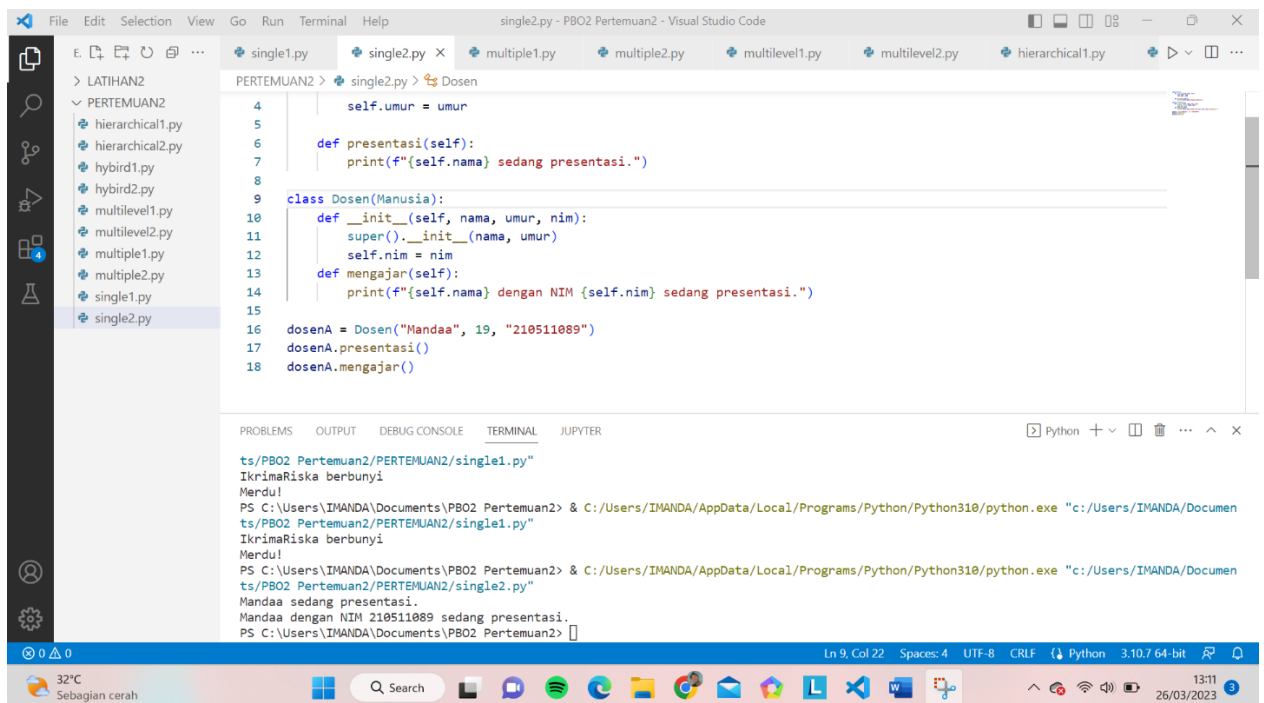
```
class Manusia:
    def __init__(self, nama, umur):
        self.nama = nama
        self.umur = umur

    def presentasi(self):
        print(f"{self.nama} sedang presentasi.")

class Dosen(Manusia):
    def __init__(self, nama, umur, nim):
        super().__init__(nama, umur)
        self.nim = nim
    def mengajar(self):
        print(f"{self.nama} dengan NIM {self.nim} sedang presentasi.")

dosenA = Dosen("Mandaa", 19, "210511089")
dosenA.presentasi()
dosenA.mengajar()
```

OUTPUT SS



```
ts/PBO2 Pertemuan2/PERTEMUAN2/single1.py"
IkrimaRiska berbunyi
Merdu!
PS C:\Users\IMANDA\Documents\PBO2 Pertemuan2> & C:/Users/IMANDA/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/IMANDA/Documen
ts/PBO2 Pertemuan2/PERTEMUAN2/single2.py"
IkrimaRiska berbunyi
Merdu!
PS C:\Users\IMANDA\Documents\PBO2 Pertemuan2> & C:/Users/IMANDA/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/IMANDA/Documen
ts/PBO2 Pertemuan2/PERTEMUAN2/single2.py"
Mandaa sedang presentasi.
Mandaa dengan NIM 210511089 sedang presentasi.
PS C:\Users\IMANDA\Documents\PBO2 Pertemuan2>
```

3. multiple1.py

```

class Tenaga :
    def setTenaga (self,tenaga):
        self.tenaga = tenaga

    def showTenaga(self):
        print(self.tenaga)

class Kelompok :
    def setKelompok(self,kelompok):
        self.kelompok = kelompok

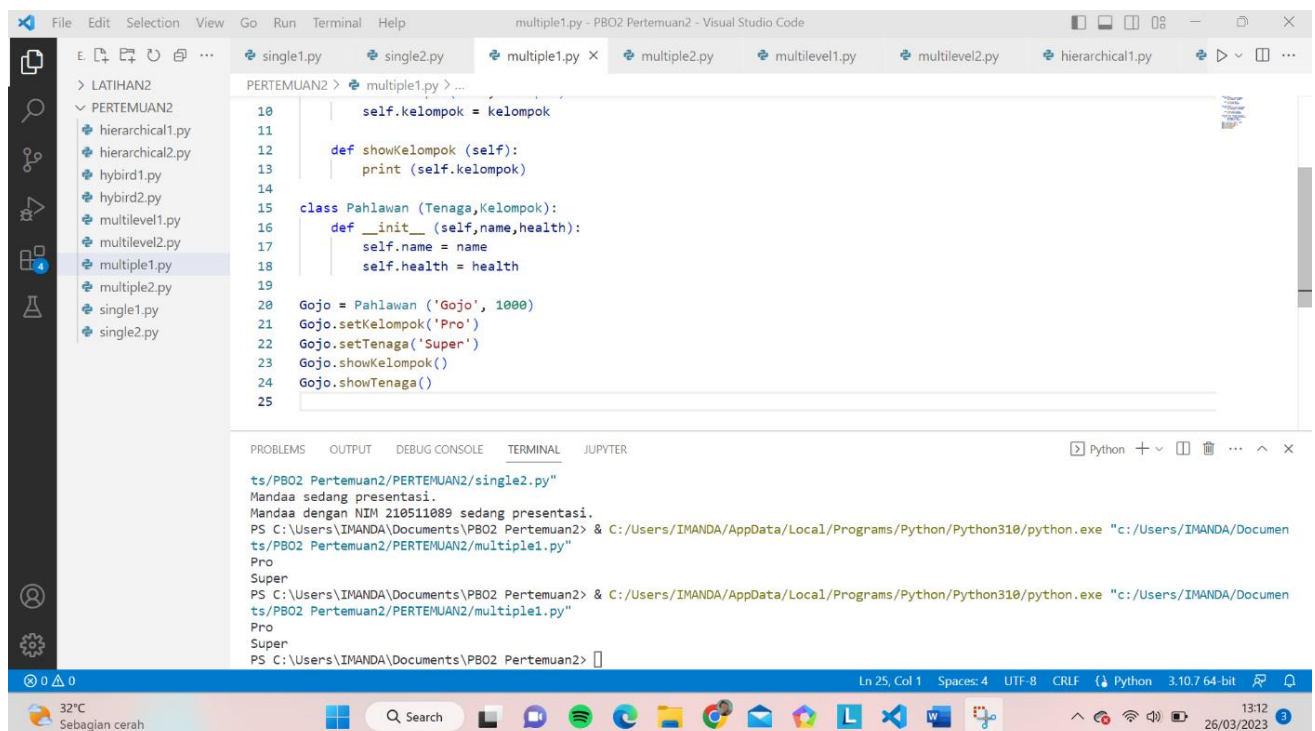
    def showKelompok (self):
        print (self.kelompok)

class Pahlawan (Tenaga,Kelompok):
    def __init__ (self,name,health):
        self.name = name
        self.health = health

Gojo = Pahlawan ('Gojo', 1000)
Gojo.setKelompok('Pro')
Gojo.setTenaga('Super')
Gojo.showKelompok()
Gojo.showTenaga()

```

OUTPUT SS



The screenshot shows a Visual Studio Code window with a file explorer on the left, a code editor in the center, and a terminal at the bottom. The file explorer shows a project named 'PERTEMUAN2' with several Python files. The code editor displays the same Python code as shown in the first block. The terminal shows the output of running the script, which includes the text 'Mandaa sedang presentasi.' and 'Mandaa dengan NIM 210511089 sedang presentasi.' followed by the output of the Gojo object's attributes.

```

ts/PERTEMUAN2/PERTEMUAN2/single2.py"
Mandaa sedang presentasi.
Mandaa dengan NIM 210511089 sedang presentasi.
PS C:\Users\IMANDA\Documents\PERTEMUAN2> & C:/Users/IMANDA/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/IMANDA/Documen
ts/PERTEMUAN2/PERTEMUAN2/multiple1.py"
Pro
Super
PS C:\Users\IMANDA\Documents\PERTEMUAN2> & C:/Users/IMANDA/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/IMANDA/Documen
ts/PERTEMUAN2/PERTEMUAN2/multiple1.py"
Pro
Super
PS C:\Users\IMANDA\Documents\PERTEMUAN2>

```

4. multiple2.py

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

class Penari:
    def __init__(self, style):
        self.style = style

class Murid (Manusia, Penari):
    def __init__(self, nama, umur, style):
        Manusia.__init__(self, nama, umur)
        Penari.__init__(self,style)

Gojo = Murid( 'Gojo', 19, 'Cool')
print (Gojo.nama)
print (Gojo.umur)
print (Gojo.style)
```

OUTPUT SS

5. hierarchical1.Py

```

class Parent:
    def func1(self):
        print("This function is in parent class.")

class Child1(Parent):
    def func2(self):
        print("This function is in child 1.")

class Child2(Parent):
    def func3(self):
        print("This function is in child 2.")

# Driver's code
object1 = Child1()
object2 = Child2()
object1.func1()
object1.func2()
object2.func1()
object2.func3()

```

OUTPUT SS

The screenshot shows the Visual Studio Code interface with the following components:

- File Explorer:** Shows a project structure with files like `hierarchichal1.py`, `hierarchichal2.py`, `hybird1.py`, `hybird2.py`, `multilevel1.py`, `multilevel2.py`, `multiple1.py`, `multiple2.py`, `single1.py`, and `single2.py`.
- Editor:** Displays the code from the previous block, with line numbers 1 through 17.
- Terminal:** Shows the command prompt output:


```

PS C:\Users\IMANDA\Documents\PB02 Pertemuan2> & C:/Users/IMANDA/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/IMANDA/Document
ts/PB02 Pertemuan2/PERTEMUAN2/hybird2.py"
Buddy: Woof!
Mittens: Meow
Polly: Tweet tweet!
PS C:\Users\IMANDA\Documents\PB02 Pertemuan2> & C:/Users/IMANDA/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/IMANDA/Document
ts/PB02 Pertemuan2/PERTEMUAN2/hierarchichal1.py"
This function is in parent class.
This function is in child 1.
This function is in parent class.
This function is in child 2.
PS C:\Users\IMANDA\Documents\PB02 Pertemuan2>

```
- Taskbar:** Shows the system clock at 13:14 on 26/03/2023, along with various application icons.

6. hierarchical2.py

```
class Animal:
    def __init__(self, name):
        self.name = name

    def speak(self):
        print(f"{self.name} speaks.")

class Dog(Animal):
    def __init__(self, name):
        super().__init__(name)

    def speak(self):
        print(f"{self.name} barks.")

class Cat(Animal):
    def __init__(self, name):
        super().__init__(name)

    def speak(self):
        print(f"{self.name} meows.")

my_dog = Dog("Fido")
my_cat = Cat("Whiskers")
my_dog.speak()
my_cat.speak()
```

OUTPUT SS

Visual Studio Code interface showing a Python project named "PERTEMUAN2" with a file explorer on the left. The file explorer lists files: LATIHAN2, PERTEMUAN2, hierarchical1.py, hierarchical2.py, hybrid1.py, hybrid2.py, multilevel1.py, multilevel2.py, multiple1.py, multiple2.py, single1.py, and single2.py. The main editor displays the code for hierarchical2.py, which defines a Cat class and instantiates objects my_dog and my_cat.

```
11
12     def speak(self):
13         print(f"{self.name} barks.")
14
15 class Cat(Animal):
16     def __init__(self, name):
17         super().__init__(name)
18
19     def speak(self):
20         print(f"{self.name} meows.")
21
22 my_dog = Dog("Fido")
23 my_cat = Cat("Whiskers")
24 my_dog.speak()
25 my_cat.speak()
26
```

The terminal output shows the execution of the code, displaying the output of the speak() method for both objects.

```
Polly: Tweet tweet!
PS C:\Users\IMANDA\Documents\PBO2 Pertemuan2> & C:/Users/IMANDA/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/IMANDA/Documen
ts/PBO2 Pertemuan2/PERTEMUAN2/hierarchical1.py"
This function is in parent class.
This function is in child 1.
This function is in parent class.
This function is in child 2.
PS C:\Users\IMANDA\Documents\PBO2 Pertemuan2> & C:/Users/IMANDA/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/IMANDA/Documen
ts/PBO2 Pertemuan2/PERTEMUAN2/hierarchical2.py"
Fido barks.
Whiskers meows.
PS C:\Users\IMANDA\Documents\PBO2 Pertemuan2>
```

The status bar at the bottom indicates the current file is hierarchical2.py, line 15, column 19, with 4 spaces, UTF-8 encoding, and CRLF line endings. The system tray shows the temperature is 32°C and the time is 13:15 on 26/03/2023.

7. multilevel1.py

```
class Animal:
    def __init__(self, species):
        self.species = species

    def eat(self):
        print("The animal is eating.")

class Pet(Animal):
    def __init__(self, name, species):
        super().__init__(species)
        self.name = name

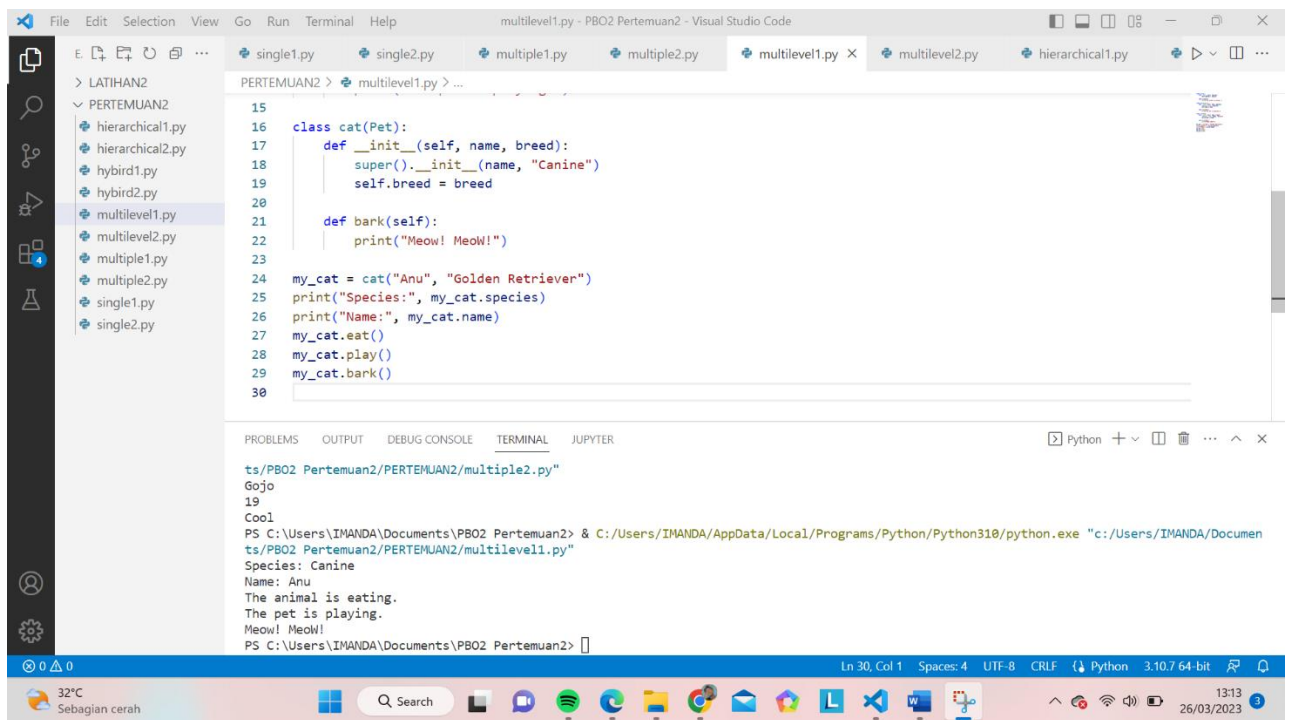
    def play(self):
        print("The pet is playing.")

class cat(Pet):
    def __init__(self, name, breed):
        super().__init__(name, "Canine")
        self.breed = breed

    def bark(self):
        print("Meow! Meow!")

my_cat = cat("Anu", "Golden Retriever")
print("Species:", my_cat.species)
print("Name:", my_cat.name)
my_cat.eat()
my_cat.play()
my_cat.bark()
```

OUTPUT SS



8. multilevel2.py

```

class Vehicle:
    def __init__(self, color, wheels):
        self.color = color
        self.wheels = wheels

class Car(Vehicle):
    def __init__(self, color, wheels, speed):
        super().__init__(color, wheels)
        self.speed = speed

    def drive(self):
        print(f"The {self.color} car is driving at {self.speed}km/h.")

class ElectricCar(Car):
    def __init__(self, color, wheels, speed, battery_capacity):
        super().__init__(color, wheels, speed)
        self.battery_capacity = battery_capacity

    def charge(self):
        print(f"The {self.color} electric car is charging its battery with {self.battery_capacity} kWh.")

my_electric_car = ElectricCar("red", 4, 120, 60)
my_electric_car.drive()
my_electric_car.charge()

```

OUTPUT SS

The screenshot shows a Visual Studio Code editor with a file explorer on the left displaying a project named 'LATIHAN2'. The file 'multilevel2.py' is selected. The editor window shows the following Python code:

```

12 print(f"The {self.color} car is driving at {self.speed}km/h.")
13
14 class ElectricCar(Car):
15     def __init__(self, color, wheels, speed, battery_capacity):
16         super().__init__(color, wheels, speed)
17         self.battery_capacity = battery_capacity
18
19     def charge(self):
20         print(f"The {self.color} electric car is charging its battery with {self.battery_capacity} kWh.")
21
22 my_electric_car = ElectricCar("red", 4, 120, 60)
23 my_electric_car.drive()
24 my_electric_car.charge()
25

```

The terminal window at the bottom shows the output of the program:

```

PS C:\Users\IMANDA\Documents\PB02 Pertemuan2> & C:/Users/IMANDA/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/IMANDA/Documents/PB02 Pertemuan2/PERTEMUAN2/multilevel1.py"
Species: Canine
Name: Anu
The animal is eating.
The pet is playing.
Meow! Meow!
PS C:\Users\IMANDA\Documents\PB02 Pertemuan2> & C:/Users/IMANDA/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/IMANDA/Documents/PB02 Pertemuan2/PERTEMUAN2/multilevel2.py"
The red car is driving at 120km/h.
The red electric car is charging its battery with 60 kWh.
PS C:\Users\IMANDA\Documents\PB02 Pertemuan2>

```

9. hybrid1.py

```
class A:
    def method_a(self):
        print("Method A")

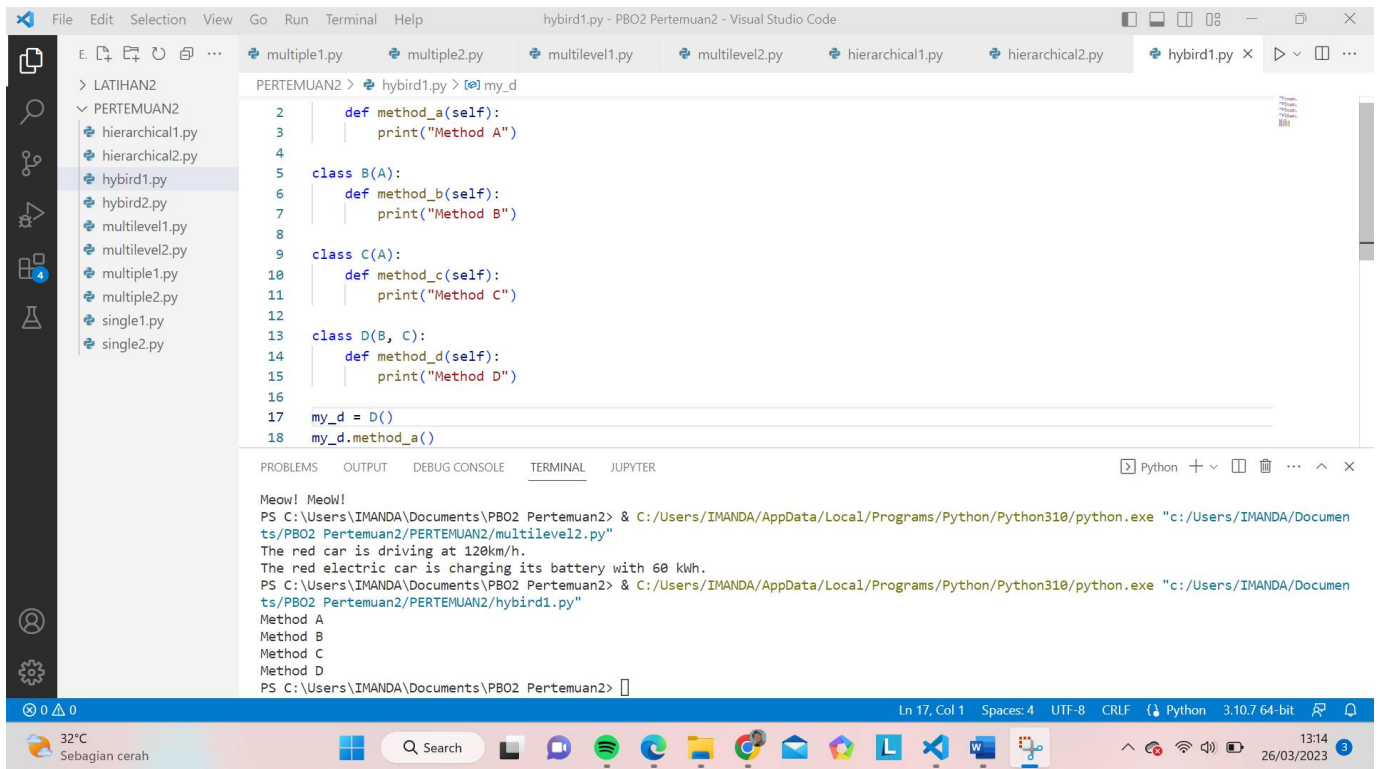
class B(A):
    def method_b(self):
        print("Method B")

class C(A):
    def method_c(self):
        print("Method C")

class D(B, C):
    def method_d(self):
        print("Method D")

my_d = D()
my_d.method_a()
my_d.method_b()
my_d.method_c()
my_d.method_d()
```

OUTPUT SS



The screenshot shows the Visual Studio Code interface with a file named 'hybird1.py' open. The file contains the following Python code:

```
PERTEMUAN2 > hybird1.py > my_d
2     def method_a(self):
3         print("Method A")
4
5     class B(A):
6         def method_b(self):
7             print("Method B")
8
9     class C(A):
10        def method_c(self):
11            print("Method C")
12
13    class D(B, C):
14        def method_d(self):
15            print("Method D")
16
17    my_d = D()
18    my_d.method_a()
```

The terminal output shows the execution of the code:

```
Meow! Meow!
PS C:\Users\IMANDA\Documents\PBO2 Pertemuan2> & C:/Users/IMANDA/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/IMANDA/Documen
ts/PBO2 Pertemuan2/PERTEMUAN2/multilevel2.py"
The red car is driving at 120km/h.
The red electric car is charging its battery with 60 kWh.
PS C:\Users\IMANDA\Documents\PBO2 Pertemuan2> & C:/Users/IMANDA/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/IMANDA/Documen
ts/PBO2 Pertemuan2/PERTEMUAN2/hybird1.py"
Method A
Method B
Method C
Method D
PS C:\Users\IMANDA\Documents\PBO2 Pertemuan2>
```

10. hybrid2.py

```
`class Animal:
    def __init__(self, name):
        self.name = name

    def speak(self):
        pass

class Dog(Animal):
    def speak(self):
        return "Woof!"

class Cat(Animal):
    def speak(self):
        return "Meow"

class Bird(Animal):
    def speak(self):
        return "Tweet tweet!"

def main():
    dog = Dog("Buddy")
    cat = Cat("Mittens")
    bird = Bird("Polly")
```

```

        print(dog.name + ": " + dog.speak())
        print(cat.name + ": " + cat.speak())
        print(bird.name + ": " + bird.speak())
if __name__ == "__main__":
    main()

```

OUTPUT SS

The screenshot shows the Visual Studio Code interface with a Python file named `hybird2.py` open. The file contains a class hierarchy where `Bird` inherits from `Animal`. The `main` function creates instances of `Dog`, `Cat`, and `Bird`, and prints their names and speech methods. The terminal at the bottom shows the command to run the script and the resulting output.

```

13     def speak(self):
14         return "Meow"
15
16     class Bird(Animal):
17         def speak(self):
18             return "Tweet tweet!"
19
20 def main():
21     dog = Dog("Buddy")
22     cat = Cat("Mittens")
23     bird = Bird("Polly")
24     print(dog.name + ": " + dog.speak())
25     print(cat.name + ": " + cat.speak())
26     print(bird.name + ": " + bird.speak())
27 if __name__ == "__main__":
28     main()
29

```

Terminal Output:

```

PS C:\Users\IMANDA\Documents\PB02 Pertemuan2> & C:/Users/IMANDA/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/IMANDA/Documen
ts/PB02 Pertemuan2/PERTEMUAN2/hybird2.py"
Method A
Method B
Method C
Method D
PS C:\Users\IMANDA\Documents\PB02 Pertemuan2> & C:/Users/IMANDA/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/IMANDA/Documen
ts/PB02 Pertemuan2/PERTEMUAN2/hybird2.py"
Buddy: Woof!
Mittens: Meow
Polly: Tweet tweet!
PS C:\Users\IMANDA\Documents\PB02 Pertemuan2>

```