LAPORAN PRAKTIKUM

PEMROGRAMAN BERORIENTASI OBJEK LANJUT

2023

Prepared By:

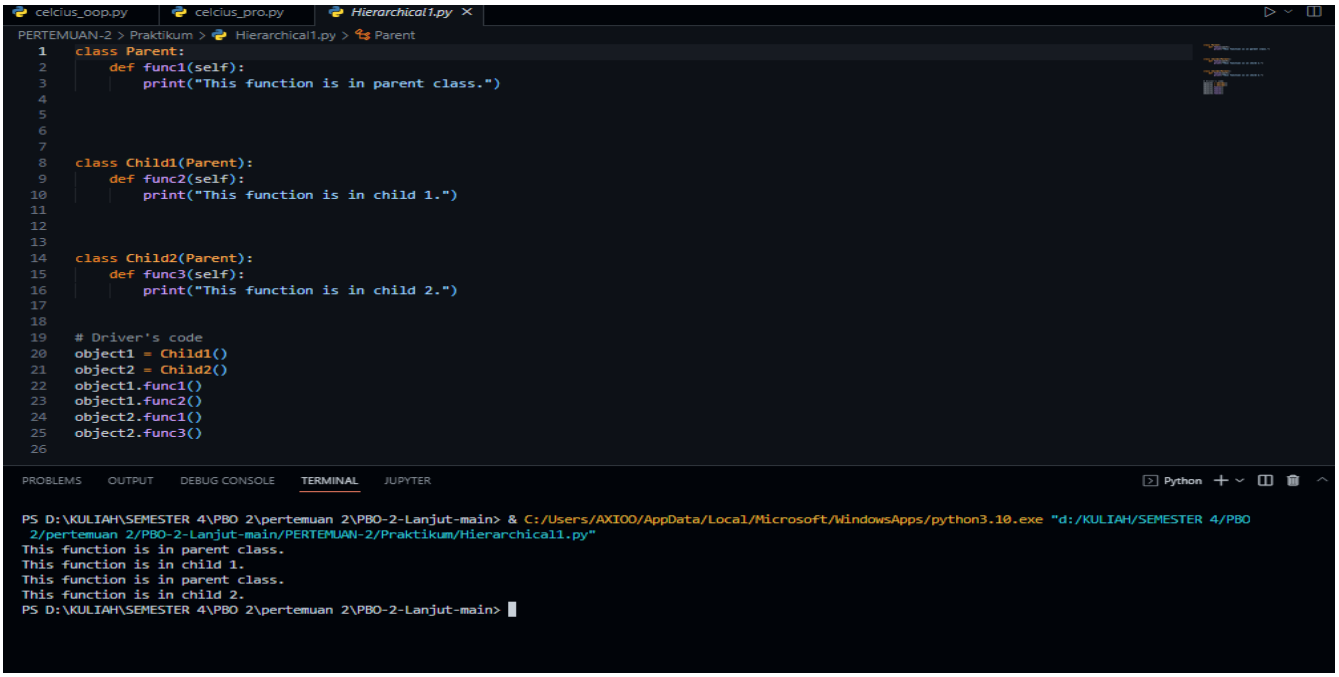
PUTRI

NIM. 210511068

hierarchical 1 :

```
# Tugas Praktikum
# NIM     : 210511068
# Nama    : PUTRI
# Kelas   : TIF21B (R2)

# 1. Buatlah Class yang mengimplementasikan Prosedural, beri nama: celcius_pro.py

class KonversiSuhu:
    @staticmethod
    def celsius1_to_fahrenheit(celsius):
        return (9/5) * celsius1 + 32

    @staticmethod
    def celsius2_to_reamur(celsius):
        return (4/5) * celsius2

    @staticmethod
    def celsius3_to_kelvin(celsius):
        return celsius3 + 273.15


celsius1 = 75
celsius2 = 30
celsius3 = 60

fahrenheit = KonversiSuhu.celsius1_to_fahrenheit(celsius1)
reamur = KonversiSuhu.celsius2_to_reamur(celsius2)
kelvin = KonversiSuhu.celsius3_to_kelvin(celsius3)

print("konversi",celsius1, "derajat celcius adalah ",fahrenheit, "derajat
fahrenheit")
print("konversi",celsius2, "derajat celcius adalah ",reamur, "derajat Reamur")
print("konversi",celsius3, "derajat celcius adalah ",kelvin, "derajat Kelvin")
```

Output :



hierarchical 2 :

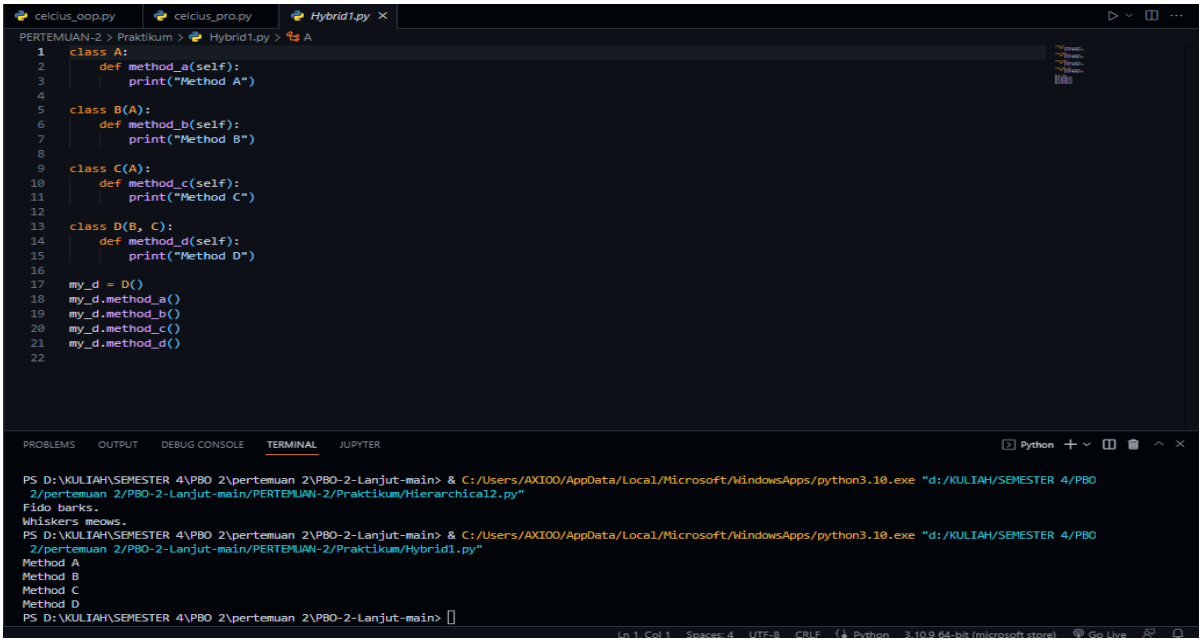
```python
class Animal:
    def __init__(self, name):
        self.name = name

    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

hybrid1:

```
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()
```

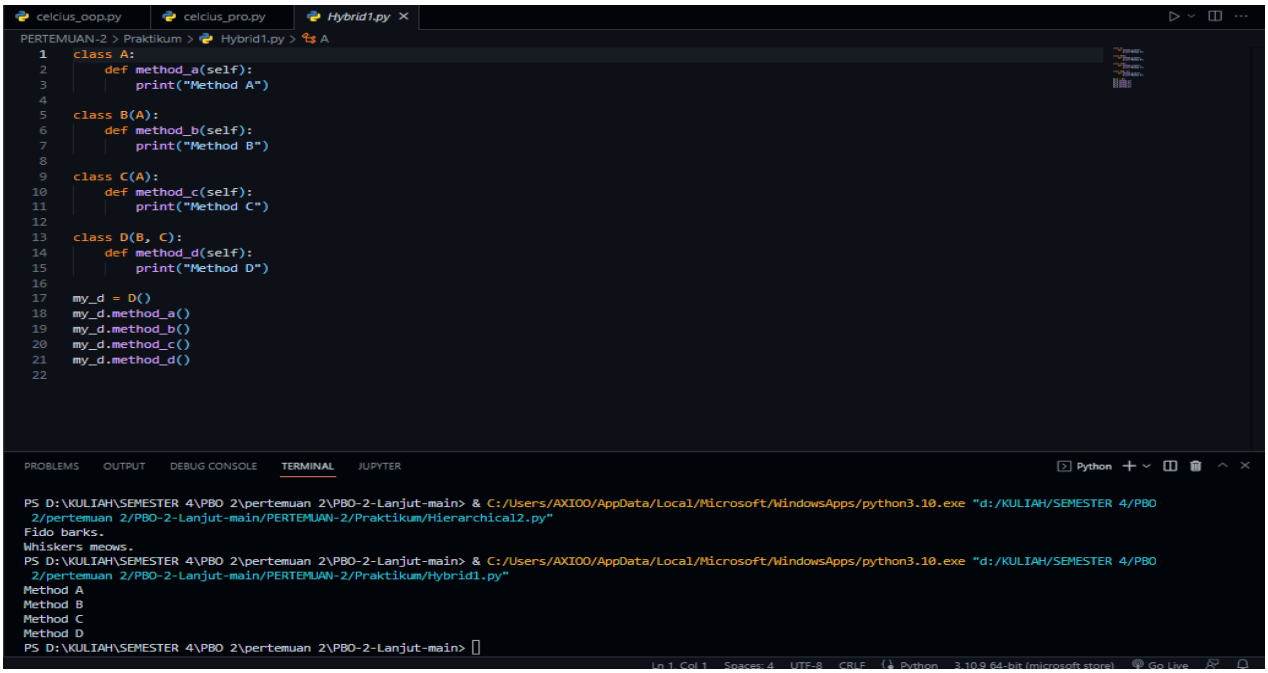
Output


celcius_oop.py
celcius_pro.py
Hybrid1.py
PERTEMUAN-2 > Praktikum > Hybrid1.py > A
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()
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
JUPYTER
Python
PS D:\KULIAH\SEMESTER 4\PBO 2\pertemuan 2\PBO-2-Lanjut-main> & C:/Users/AXIOO/AppData/Local/Microsoft/WindowsApps/python3.10.exe "d:/KULIAH/SEMESTER 4/PBO 2/pertemuan 2/PBO-2-Lanjut-main/PERTEMUAN-2/Praktikum/Hierarchical2.py"
Fido barks.
Whiskers meows.
PS D:\KULIAH\SEMESTER 4\PBO 2\pertemuan 2\PBO-2-Lanjut-main> & C:/Users/AXIOO/AppData/Local/Microsoft/WindowsApps/python3.10.exe "d:/KULIAH/SEMESTER 4/PBO 2/pertemuan 2/PBO-2-Lanjut-main/PERTEMUAN-2/Praktikum/Hybrid1.py"
Method A
Method B
Method C
Method D
PS D:\KULIAH\SEMESTER 4\PBO 2\pertemuan 2\PBO-2-Lanjut-main>

hybrid2:

```
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


celcius_oop.py
celcius_pro.py
Hybrid1.py
PERTEMUAN-2 > Praktikum > Hybrid1.py > A
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()
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
JUPYTER
Python
PS D:\KULIAH\SEMESTER 4\PBO 2\pertemuan 2\PBO-2-Lanjut-main> & C:/Users/AXIOO/AppData/Local/Microsoft/WindowsApps/python3.10.exe "d:/KULIAH/SEMESTER 4/PBO 2/pertemuan 2/PBO-2-Lanjut-main/PERTEMUAN-2/Praktikum/Hierarchical2.py"
Fido barks.
Whiskers meows.
PS D:\KULIAH\SEMESTER 4\PBO 2\pertemuan 2\PBO-2-Lanjut-main> & C:/Users/AXIOO/AppData/Local/Microsoft/WindowsApps/python3.10.exe "d:/KULIAH/SEMESTER 4/PBO 2/pertemuan 2/PBO-2-Lanjut-main/PERTEMUAN-2/Praktikum/Hybrid1.py"
Method A
Method B
Method C
Method D
PS D:\KULIAH\SEMESTER 4\PBO 2\pertemuan 2\PBO-2-Lanjut-main>

Multilevel1 :

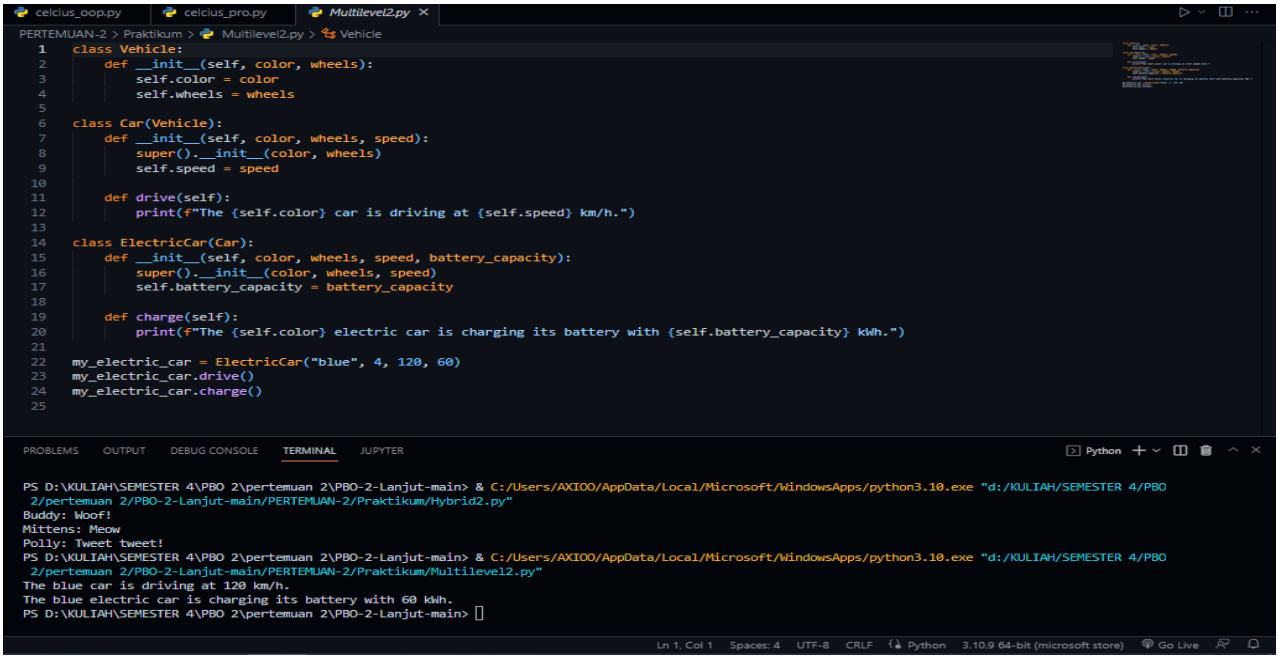
```
class Animal:
    def __init__(self, species):
        self.species = species

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

Output

Multilevel 2:

```
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("blue", 4, 120, 60)
my_electric_car.drive()
my_electric_car.charge()
```

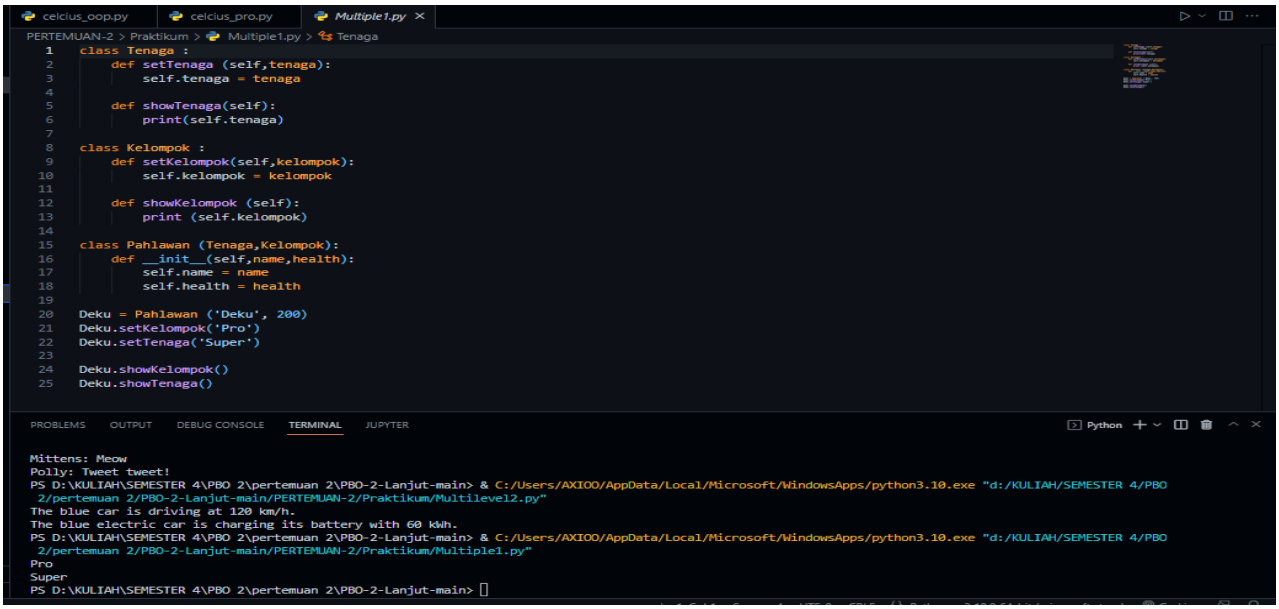
Output



Multiple 1 :

```
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

Deku = Pahlawan ('Deku', 200)
Deku.setKelompok('Pro')
Deku.setTenaga('Super')

Deku.showKelompok()
Deku.showTenaga()
```

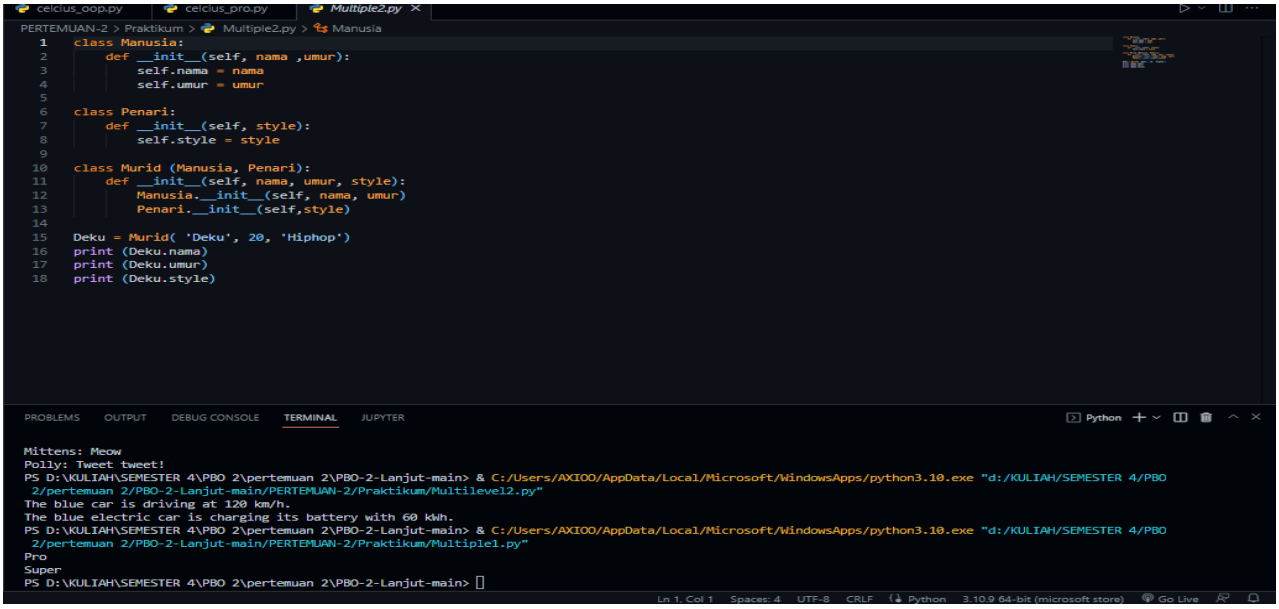
Output



Multiple 2 :

```
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)

Deku = Murid( 'Deku', 20, 'Hiphop')
print (Deku.nama)
print (Deku.umur)
print (Deku.style)
```

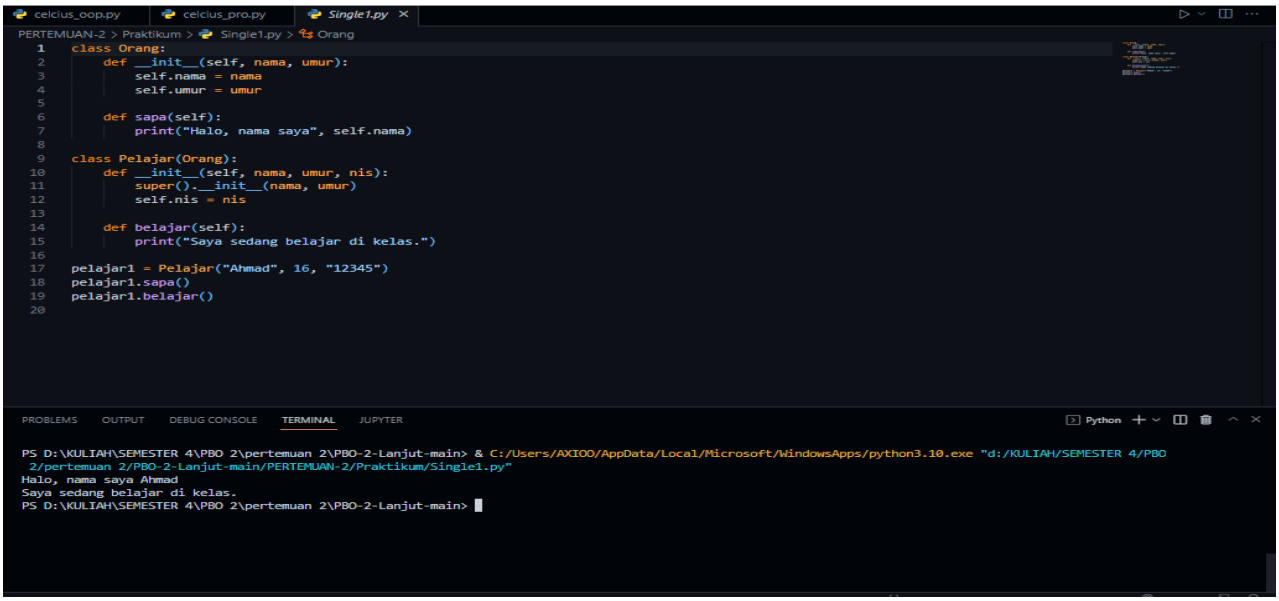
Output



Single 1:

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

    def sapa(self):
        print("Halo, nama saya", self.nama)

class Pelajar(Orang):
    def __init__(self, nama, umur, nis):
        super().__init__(nama, umur)
        self.nis = nis

    def belajar(self):
        print("Saya sedang belajar di kelas.")

pelajar1 = Pelajar("Ahmad", 16, "12345")
pelajar1.sapa()
pelajar1.belajar()
```

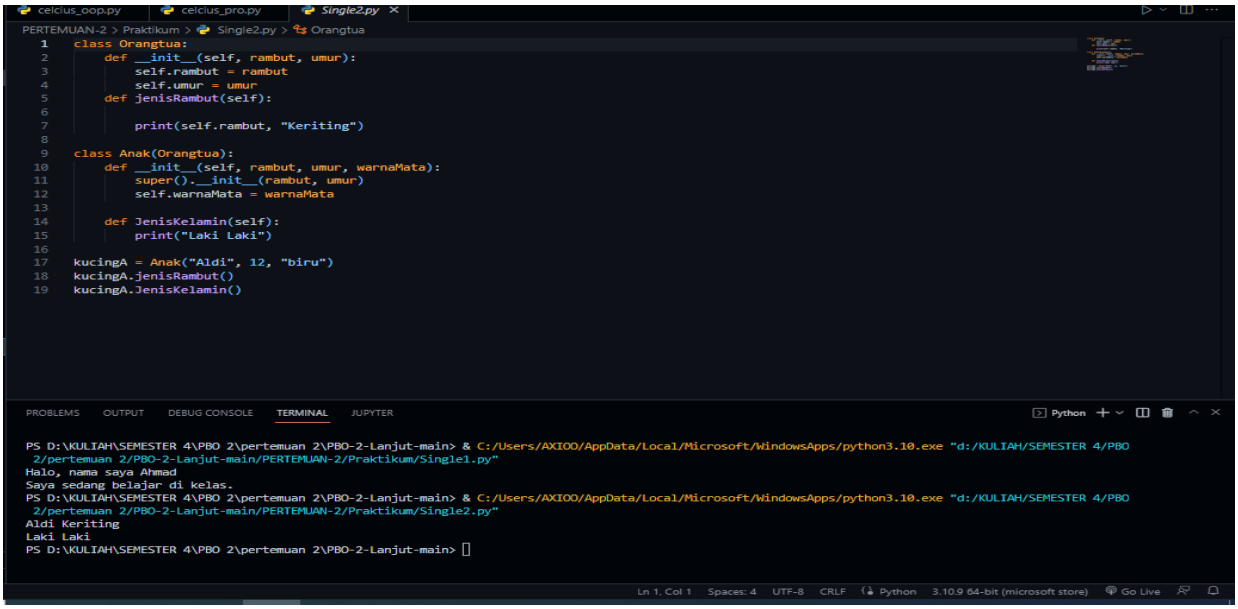
Output



Single 2 :

```
class Orangtua:
    def __init__(self, rambut, umur):
        self.rambut = rambut
        self.umur = umur
    def jenisRambut(self):

        print(self.rambut, "Keriting")

class Anak(Orangtua):
    def __init__(self, rambut, umur, warnaMata):
        super().__init__(rambut, umur)
        self.warnaMata = warnaMata

    def JenisKelamin(self):
        print("Laki Laki")

kucingA = Anak("Aldi", 12, "biru")
kucingA.jenisRambut()
kucingA.JenisKelamin()
```

Output


celcius_oop.py
celcius_pro.py
Single2.py
PERTEMUAN-2 > Praktikum > Single2.py > Orangtua
class Orangtua:
def __init__(self, rambut, umur):
self.rambut = rambut
self.umur = umur
def jenisRambut(self):
print(self.rambut, "Keriting")
class Anak(Orangtua):
def __init__(self, rambut, umur, warnaMata):
super().__init__(rambut, umur)
self.warnaMata = warnaMata
def JenisKelamin(self):
print("Laki Laki")
kucingA = Anak("Aldi", 12, "biru")
kucingA.jenisRambut()
kucingA.JenisKelamin()
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
JUPYTER
Python
PS D:\KULIAH\SEMESTER 4\PBO 2\pertemuan 2\PBO-2-Lanjut-main> & C:/Users/AXIOO/AppData/Local/Microsoft/WindowsApps/python3.10.exe "d:/KULIAH/SEMESTER 4/PBO 2/pertemuan 2/PBO-2-Lanjut-main/PERTEMUAN-2/Praktikum/Single1.py"
Halo, nama saya Ahmad
Saya sedang belajar di kelas.
PS D:\KULIAH\SEMESTER 4\PBO 2\pertemuan 2\PBO-2-Lanjut-main> & C:/Users/AXIOO/AppData/Local/Microsoft/WindowsApps/python3.10.exe "d:/KULIAH/SEMESTER 4/PBO 2/pertemuan 2/PBO-2-Lanjut-main/PERTEMUAN-2/Praktikum/Single2.py"
Aldi Keriting
Laki Laki
PS D:\KULIAH\SEMESTER 4\PBO 2\pertemuan 2\PBO-2-Lanjut-main>
Ln 1, Col 1
Spaces: 4
UTF-8
CRLF
Python
3.10.9 64-bit (microsoft store)
Go Live