LAPORAN PRAKTIKUM

PEMROGRAMAN BERORIENTASI OBJEK LANJUT

2023

Prepared By:

Aini Salsabila
210511065 / R2

Soal Praktikum 2

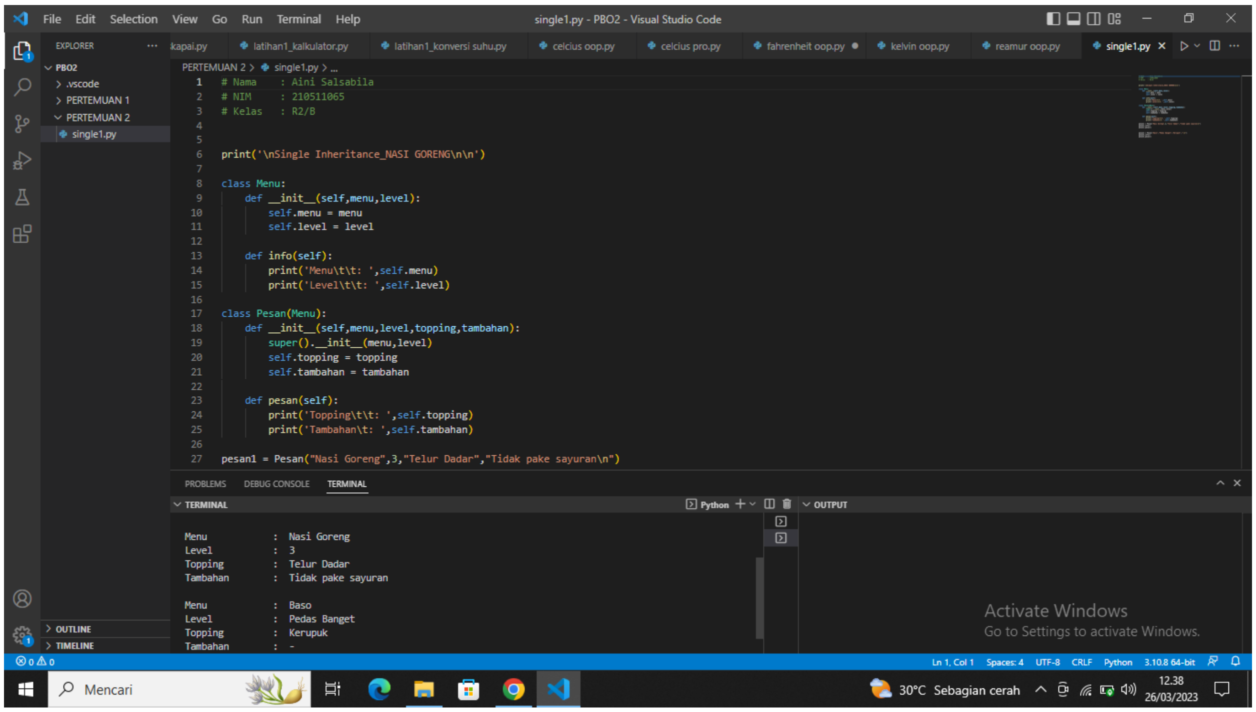
Buatlah masing-masing dua jenis pewarisan di lluar dari contoh yang di berikan:

1. **Single Inheritance:**
   a. contoh 1:

```
# Nama    : Aini Salsabila

# NIM     : 210511065

# Kelas   : R2/B


print('\nSingle Inheritance_NASI GORENG\n\n')
class Menu:

    def __init__(self,menu,level):

        self.menu = menu

        self.level = level

    def info(self):

        print('Menu\t\t: ',self.menu)

        print('Level\t\t: ',self.level)

class Pesan(Menu):

    def __init__(self,menu,level,topping,tambahan):

        super().__init__(menu,level)

        self.topping = topping

        self.tambahan = tambahan

    def pesan(self):

        print('Topping\t\t: ',self.topping)

        print('Tambahan\t: ',self.tambahan)

pesan1 = Pesan("Nasi Goreng",3,"Telur Dadar","Tidak pake sayuran\n")

pesan1.info()

pesan1.pesan()
```

```
pesan2 = Pesan("Baso","Pedas Banget","Kerupuk","-\n")
pesan2.info()

pesan2.pesan()
```

output:



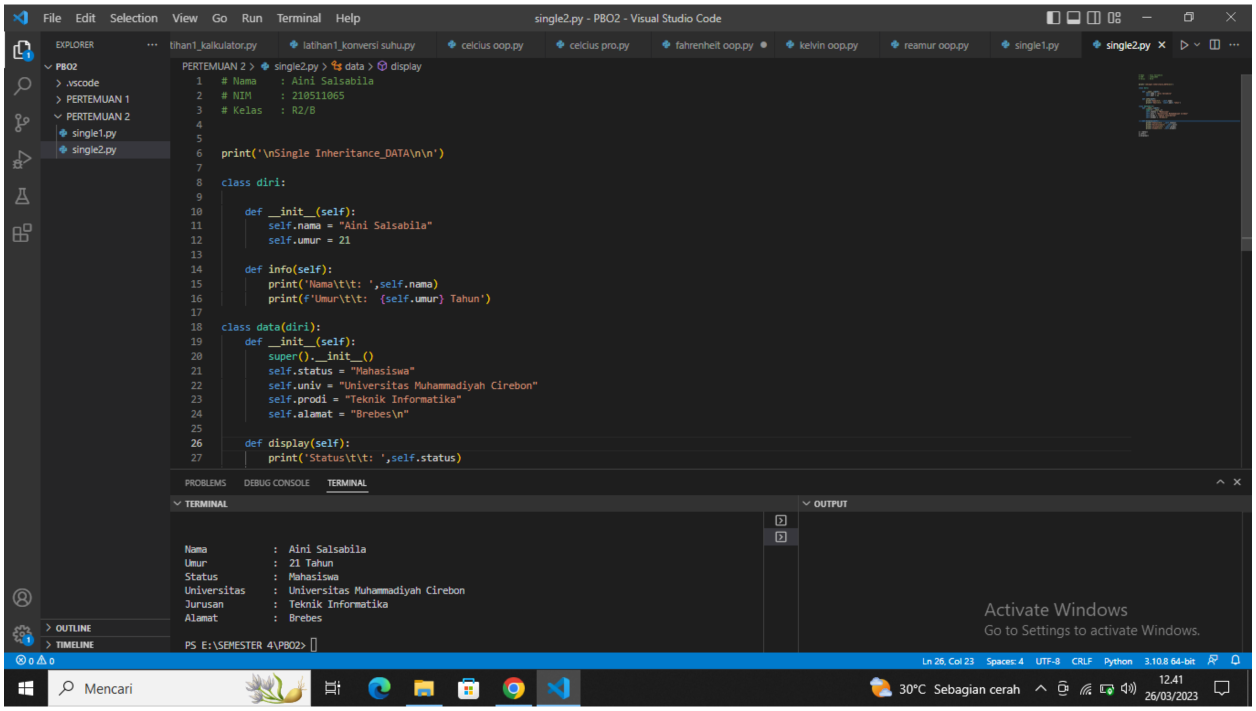
```
# Nama    : Aini Salsabila
# NIM     : 210511065
# Kelas   : R2/B


print('\nSingle Inheritance_NASI GORENG\n\n')

class Menu:
    def __init__(self,menu,level):
        self.menu = menu
        self.level = level

    def info(self):
        print('Menu\t\t: ',self.menu)
        print('Level\t\t: ',self.level)

class Pesan(Menu):
    def __init__(self,menu,level,topping,tambahan):
        super().__init__(menu,level)
        self.topping = topping
        self.tambahan = tambahan

    def pesan(self):
        print('Topping\t\t: ',self.topping)
        print('Tambahan\t: ',self.tambahan)

pesan1 = Pesan("Nasi Goreng",3,"Telur Dadar","Tidak pake sayuran\n")
```

```
Menu            :  Nasi Goreng
Level           :  3
Topping         :  Telur Dadar
Tambahan        :  Tidak pake sayuran

Menu            :  Baso
Level           :  Pedas Banget
Topping         :  Kerupuk
Tambahan        :  -
```

b. contoh 2 :

```
# Nama    : Aini Salsabila
# NIM     : 210511065
# Kelas   : R2/B


print('\nSingle Inheritance_DATA\n\n')

class diri:

    def __init__(self):
        self.nama = "Aini Salsabila"
        self.umur = 21

    def info(self):
        print('Nama\t\t: ',self.nama)
        print(f'Umur\t\t:  {self.umur} Tahun')

class data(diri):
    def __init__(self):
        super().__init__()
        self.status = "Mahasiswa"
        self.univ = "Universitas Muhammadiyah Cirebon"
        self.prodi = "Teknik Informatika"
        self.alamat = "Brebes\n"

    def display(self):
        print('Status\t\t: ',self.status)
        print('Universitas\t: ',self.univ)
        print('Jurusan\t\t: ',self.prodi)
        print('Alamat\t\t: ',self.alamat)

a = data()
a.info()
a.display()
```

output:



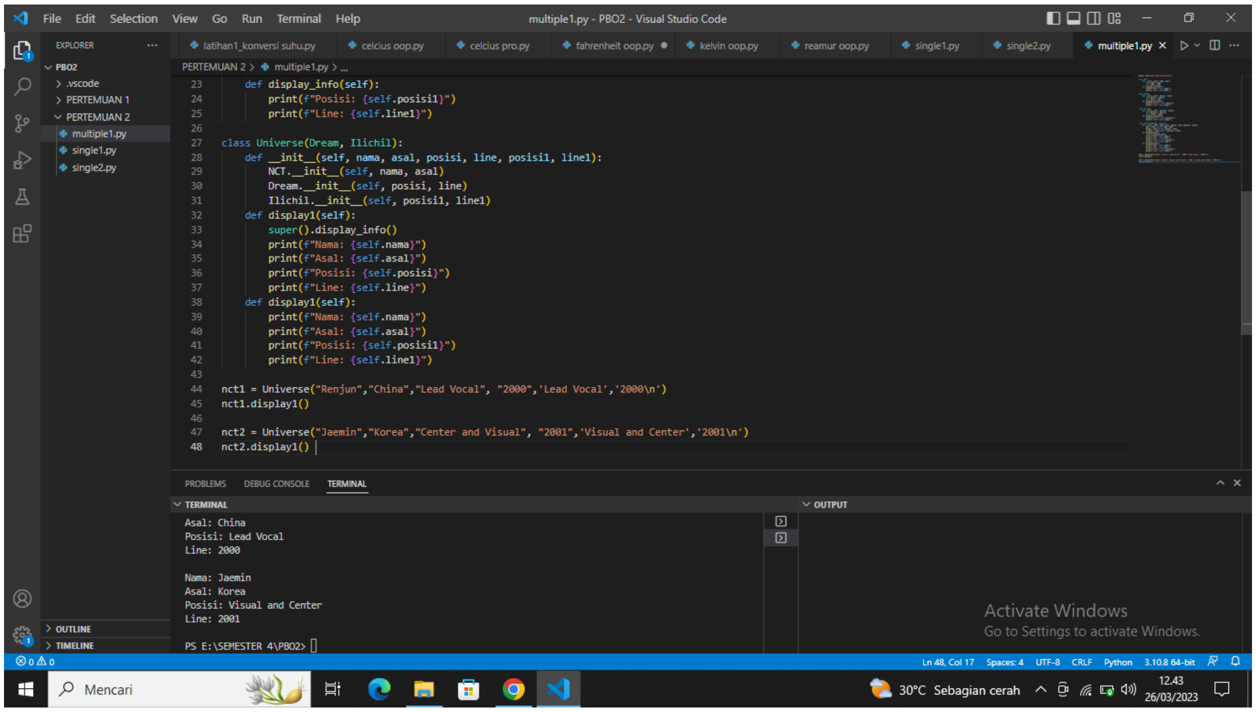
```
# Nama    : Aini Salsabila
# NIM     : 210511065
# Kelas   : R2/B


print('\nSingle Inheritance_DATA\n\n')

class diri:

    def __init__(self):
        self.nama = "Aini Salsabila"
        self.umur = 21

    def info(self):
        print('Nama\t\t: ',self.nama)
        print(f'Umur\t\t:  {self.umur} Tahun')

class data(diri):
    def __init__(self):
        super().__init__()
        self.status = "Mahasiswa"
        self.univ = "Universitas Muhammadiyah Cirebon"
        self.prodi = "Teknik Informatika"
        self.alamat = "Brebes\n"

    def display(self):
        print('Status\t\t: ',self.status)
```

```
Nama            :  Aini Salsabila
Umur            :  21 Tahun
Status          :  Mahasiswa
Universitas     :  Universitas Muhammadiyah Cirebon
Jurusan         :  Teknik Informatika
Alamat          :  Brebes

PS E:\SEMESTER 4\PBO2>
```

2. **Multiple Inheritance :**
   a. contoh 1:

```
print('\nMultiple Inheritance_NCT\n')

class NCT:
    def __init__(self, nama, asal):
        self.nama = nama
        self.asal = asal
    def display_info(self):
        print(f"Nama: {self.nama}")
        print(f"Asal: {self.asal}")

class Dream:
    def __init__(self, posisi, line):
        self.posisi = posisi
        self.line = line
    def display_info(self):
        print(f"Posisi: {self.posisi}")
        print(f"Line: {self.line}")

class Ilichil:
    def __init__(self, posisi1, line1):
        self.posisi1 = posisi1
        self.line1 = line1
    def display_info(self):
        print(f"Posisi: {self.posisi1}")
        print(f"Line: {self.line1}")

class Universe(Dream, Ilichil):
    def __init__(self, nama, asal, posisi, line, posisi1, line1):
        NCT.__init__(self, nama, asal)
        Dream.__init__(self, posisi, line)
        Ilichil.__init__(self, posisi1, line1)
    def display1(self):
        super().display_info()
        print(f"Nama: {self.nama}")
        print(f"Asal: {self.asal}")
        print(f"Posisi: {self.posisi}")
        print(f"Line: {self.line}")
    def display1(self):
        print(f"Nama: {self.nama}")
 print(f"Asal: {self.asal}")
        print(f"Posisi: {self.posisi1}")
        print(f"Line: {self.line1}")

nct1 = Universe("Renjun","China","Lead Vocal", "2000",'Lead Vocal','2000\n')
nct1.display1()

nct2 = Universe("Jaemin","Korea","Center and Visual", "2001",'Visual and Center','2001\n')
nct2.display1()
```

output:



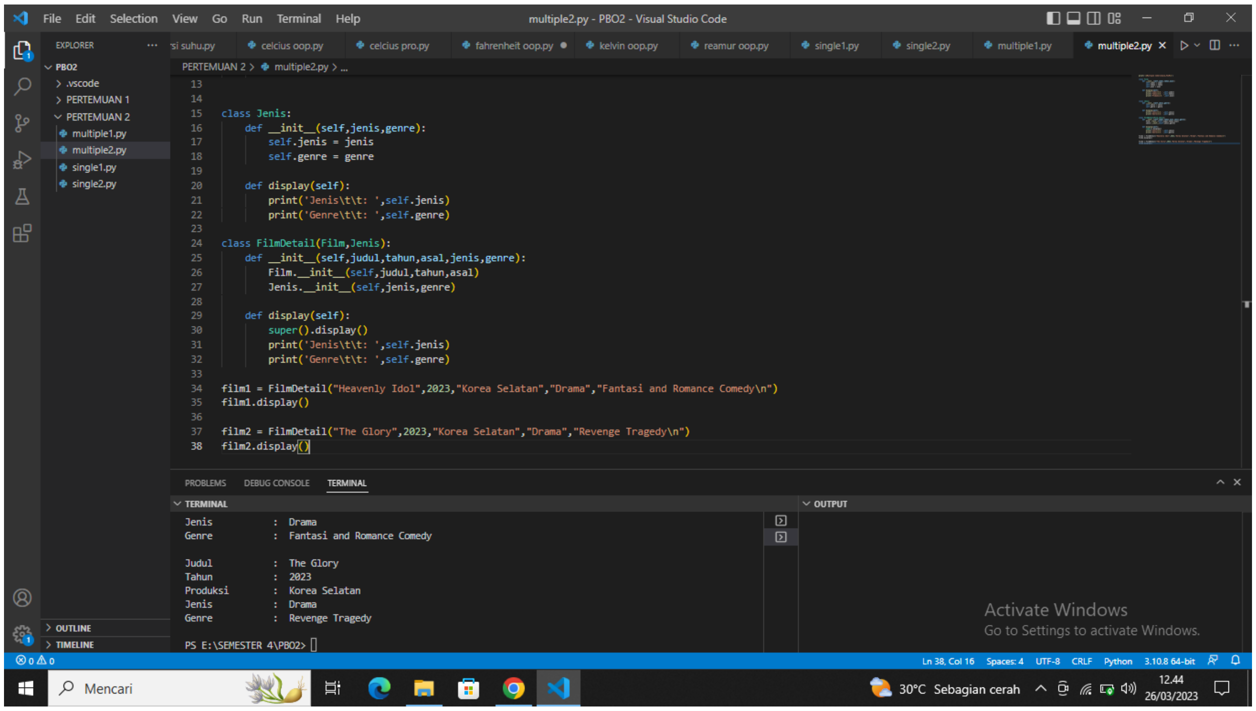
```
    def display_info(self):
        print(f"Posisi: {self.posisi1}")
        print(f"Line: {self.line1}")

class Universe(Dream, Ilichil):
    def __init__(self, nama, asal, posisi, line, posisi1, line1):
        NCT.__init__(self, nama, asal)
        Dream.__init__(self, posisi, line)
        Ilichil.__init__(self, posisi1, line1)
    def display1(self):
        super().display_info()
        print(f"Nama: {self.nama}")
        print(f"Asal: {self.asal}")
        print(f"Posisi: {self.posisi}")
        print(f"Line: {self.line}")
    def display1(self):
        print(f"Nama: {self.nama}")
        print(f"Asal: {self.asal}")
        print(f"Posisi: {self.posisi1}")
        print(f"Line: {self.line1}")

nct1 = Universe("Renjun","China","Lead Vocal", "2000",'Lead Vocal','2000\n')
nct1.display1()

nct2 = Universe("Jaemin","Korea","Center and Visual", "2001",'Visual and Center','2001\n')
nct2.display1()
```

```
Asal: China
Posisi: Lead Vocal
Line: 2000

Nama: Jaemin
Asal: Korea
Posisi: Visual and Center
Line: 2001

PS E:\SEMESTER 4\PBO2>
```

b. contoh 2:

```
print('\nMultiple Inheritance_FILM\n')

class Film:
    def __init__(self,judul,tahun,asal):
        self.judul = judul
        self.tahun = tahun
        self.asal = asal

    def display(self):
        print('Judul\t\t: ',self.judul)
        print('Tahun\t\t: ',self.tahun)
        print('Produksi\t: ',self.asal)

class Jenis:
    def __init__(self,jenis,genre):
        self.jenis = jenis
        self.genre = genre

    def display(self):
        print('Jenis\t\t: ',self.jenis)
        print('Genre\t\t: ',self.genre)

class FilmDetail(Film,Jenis):
    def __init__(self,judul,tahun,asal,jenis,genre):
        Film.__init__(self,judul,tahun,asal)
        Jenis.__init__(self,jenis,genre)

    def display(self):
        super().display()
        print('Jenis\t\t: ',self.jenis)
        print('Genre\t\t: ',self.genre)

film1 = FilmDetail("Heavenly Idol",2023,"Korea Selatan","Drama","Fantasi and Romance
Comedy\n")
film1.display()

film2 = FilmDetail("The Glory",2023,"Korea Selatan","Drama","Revenge Tragedy\n")
film2.display()
```

output:



File Edit Selection View Go Run Terminal Help
multiple2.py - PBO2 - Visual Studio Code
EXPLORER
PBO2
.vscode
PERTEMUAN 1
PERTEMUAN 2
multiple1.py
multiple2.py
single1.py
single2.py
OUTLINE
TIMELINE
PERTEMUAN 2 > multiple2.py > ...
PROBLEMS DEBUG CONSOLE TERMINAL
OUTPUT
Activate Windows
Go to Settings to activate Windows.
Ln 38, Col 16 Spaces: 4 UTF-8 CRLF Python 3.10.8 64-bit
Mencari
30°C Sebagian cerah
12.44
26/03/2023

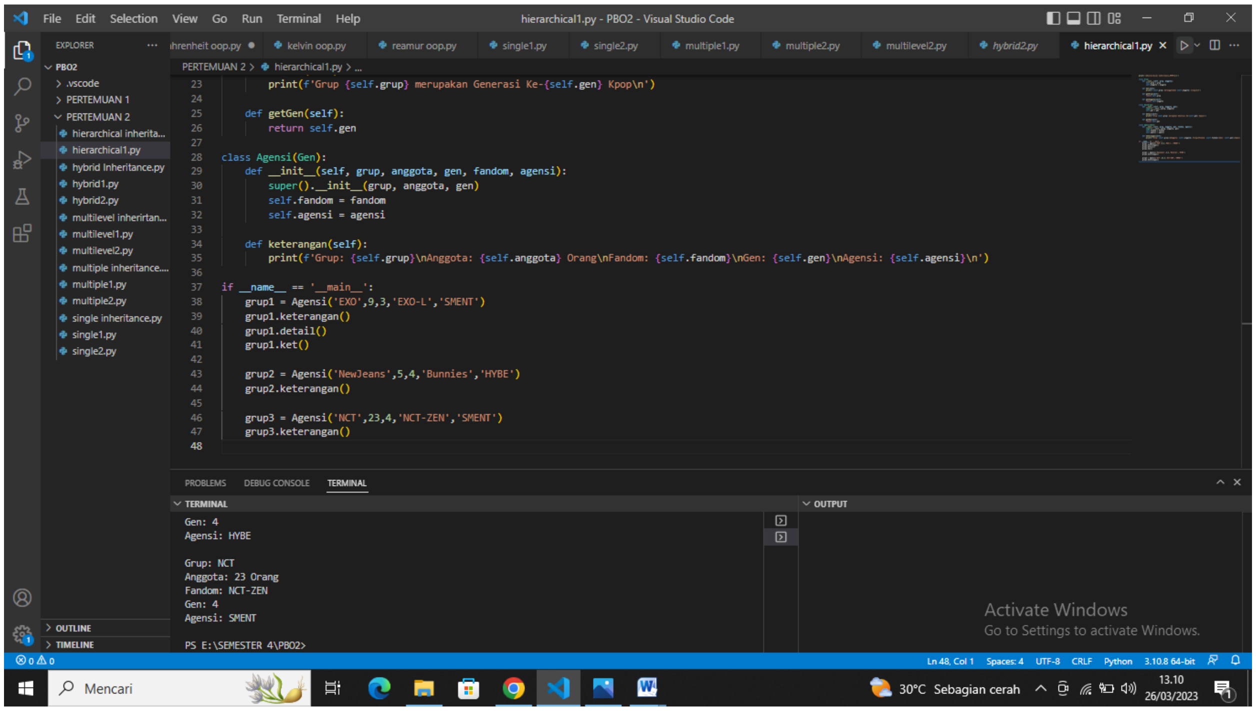
```
class Jenis:
    def __init__(self,jenis,genre):
        self.jenis = jenis
        self.genre = genre

    def display(self):
        print('Jenis\t\t: ',self.jenis)
        print('Genre\t\t: ',self.genre)

class FilmDetail(Film,Jenis):
    def __init__(self,judul,tahun,asal,jenis,genre):
        Film.__init__(self,judul,tahun,asal)
        Jenis.__init__(self,jenis,genre)

    def display(self):
        super().display()
        print('Jenis\t\t: ',self.jenis)
        print('Genre\t\t: ',self.genre)

film1 = FilmDetail("Heavenly Idol",2023,"Korea Selatan","Drama","Fantasi and Romance Comedy\n")
film1.display()

film2 = FilmDetail("The Glory",2023,"Korea Selatan","Drama","Revenge Tragedy\n")
film2.display()
```

```
Jenis           :  Drama
Genre           :  Fantasi and Romance Comedy

Judul           :  The Glory
Tahun           :  2023
Produksi        :  Korea Selatan
Jenis           :  Drama
Genre           :  Revenge Tragedy

PS E:\SEMESTER 4\PBO2>
```

3. **Hierarchial Inheritance :**
   a. contoh 1:

```
print('\nHierarchical Inheritance_KPOP\n\n')

class Grup:
    def __init__(self, grup, anggota):
        self.grup = grup
        self.anggota = anggota

    def ket(self):
        print(f'{self.grup} beranggotakan {self.anggota} orang\n\n')

    def getGrup(self):
        return self.grup

    def getAnggota(self):
        return self.anggota

class Gen(Grup):
    def __init__(self, grup, anggota, gen):
        super().__init__(grup, anggota)
        self.gen = gen

    def detail(self):
        print(f'Grup {self.grup} merupakan Generasi Ke-{self.gen} Kpop\n')

    def getGen(self):
        return self.gen

class Agensi(Gen):
    def __init__(self, grup, anggota, gen, fandom, agensi):
        super().__init__(grup, anggota, gen)
        self.fandom = fandom
        self.agensi = agensi

    def keterangan(self):
        print(f'Grup: {self.grup}\nAnggota: {self.anggota} Orang\nFandom: {self.fandom}\nGen:
{self.gen}\nAgensi: {self.agensi}\n')

if __name__ == '__main__':
    grup1 = Agensi('EXO',9,3,'EXO-L','SMENT')
    grup1.keterangan()
    grup1.detail()
    grup1.ket()

    grup2 = Agensi('NewJeans',5,4,'Bunnies','HYBE')
    grup2.keterangan()

    grup3 = Agensi('NCT',23,4,'NCT-ZEN','SMENT')
    grup3.keterangan()
```

output:



```
        print(f'Grup {self.grup} merupakan Generasi Ke-{self.gen} Kpop\n')

    def getGen(self):
        return self.gen

class Agensi(Gen):
    def __init__(self, grup, anggota, gen, fandom, agensi):
        super().__init__(grup, anggota, gen)
        self.fandom = fandom
        self.agensi = agensi

    def keterangan(self):
        print(f'Grup: {self.grup}\nAnggota: {self.anggota} Orang\nFandom: {self.fandom}\nGen: {self.gen}\nAgensi: {self.agensi}\n')

if __name__ == '__main__':
    grup1 = Agensi('EXO',9,3,'EXO-L','SMENT')
    grup1.keterangan()
    grup1.detail()
    grup1.ket()

    grup2 = Agensi('NewJeans',5,4,'Bunnies','HYBE')
    grup2.keterangan()

    grup3 = Agensi('NCT',23,4,'NCT-ZEN','SMENT')
    grup3.keterangan()
```

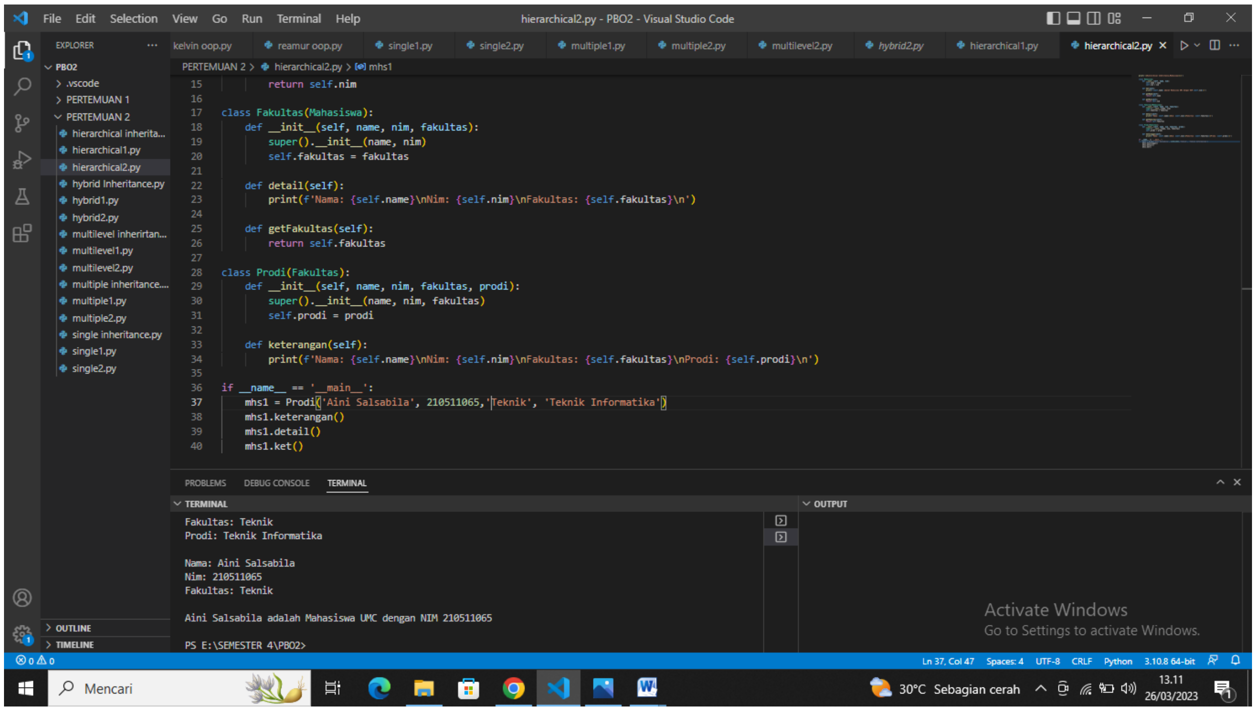
```
Gen: 4
Agensi: HYBE

Grup: NCT
Anggota: 23 Orang
Fandom: NCT-ZEN
Gen: 4
Agensi: SMENT

PS E:\SEMESTER 4\PBO2>
```

b. contoh 2:

```
print('\nHierarchical Inheritance_Mahasiswa\n\n')

class Mahasiswa:
    def __init__(self, name, nim):
        self.name = name
        self.nim = nim

    def ket(self):
        print(f'{self.name} adalah Mahasiswa UMC dengan NIM {self.nim}\n')

    def getName(self):
        return self.name

    def getNim(self):
        return self.nim

class Fakultas(Mahasiswa):
    def __init__(self, name, nim, fakultas):
        super().__init__(name, nim)
        self.fakultas = fakultas

    def detail(self):
```

```python
        print(f'Nama: {self.name}\nNim: {self.nim}\nFakultas: {self.fakultas}\n')

    def getFakultas(self):
        return self.fakultas

class Prodi(Fakultas):
    def __init__(self, name, nim, fakultas, prodi):
        super().__init__(name, nim, fakultas)
        self.prodi = prodi

    def keterangan(self):
        print(f'Nama: {self.name}\nNim: {self.nim}\nFakultas: {self.fakultas}\nProdi: {self.prodi}\n')

if __name__ == '__main__':
    mhs1 = Prodi('Aini Salsabila', 210511065,'Teknik', 'Teknik Informatika')
    mhs1.keterangan()
    mhs1.detail()
    mhs1.ket()
```

output:



```python
            return self.nim

class Fakultas(Mahasiswa):
    def __init__(self, name, nim, fakultas):
        super().__init__(name, nim)
        self.fakultas = fakultas

    def detail(self):
        print(f'Nama: {self.name}\nNim: {self.nim}\nFakultas: {self.fakultas}\n')

    def getFakultas(self):
        return self.fakultas

class Prodi(Fakultas):
    def __init__(self, name, nim, fakultas, prodi):
        super().__init__(name, nim, fakultas)
        self.prodi = prodi

    def keterangan(self):
        print(f'Nama: {self.name}\nNim: {self.nim}\nFakultas: {self.fakultas}\nProdi: {self.prodi}\n')

if __name__ == '__main__':
    mhs1 = Prodi('Aini Salsabila', 210511065,'Teknik', 'Teknik Informatika')
    mhs1.keterangan()
    mhs1.detail()
    mhs1.ket()
```

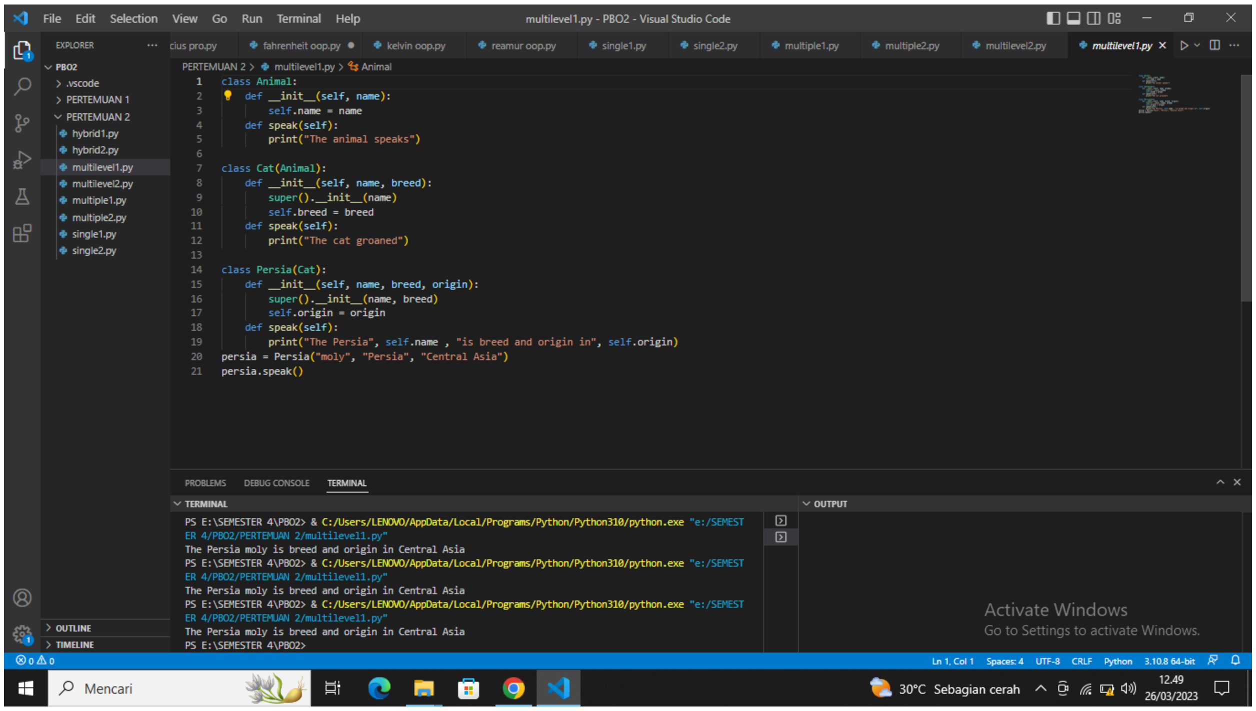
```
Fakultas: Teknik
Prodi: Teknik Informatika

Nama: Aini Salsabila
Nim: 210511065
Fakultas: Teknik

Aini Salsabila adalah Mahasiswa UMC dengan NIM 210511065

PS E:\SEMESTER 4\PBO2>
```

4. Multilevel Inheritance :

a. Contoh 1:

```
class Animal:
    def __init__(self, name):
        self.name = name
    def speak(self):
        print("The animal speaks")

class Cat(Animal):
    def __init__(self, name, breed):
        super().__init__(name)
        self.breed = breed
    def speak(self):
        print("The cat groaned")

class Persia(Cat):
    def __init__(self, name, breed, origin):
        super().__init__(name, breed)
        self.origin = origin
    def speak(self):
        print("The Persia", self.name , "is breed and origin in", self.origin)
persia = Persia("moly", "Persia", "Central Asia")
persia.speak()
```

Output:



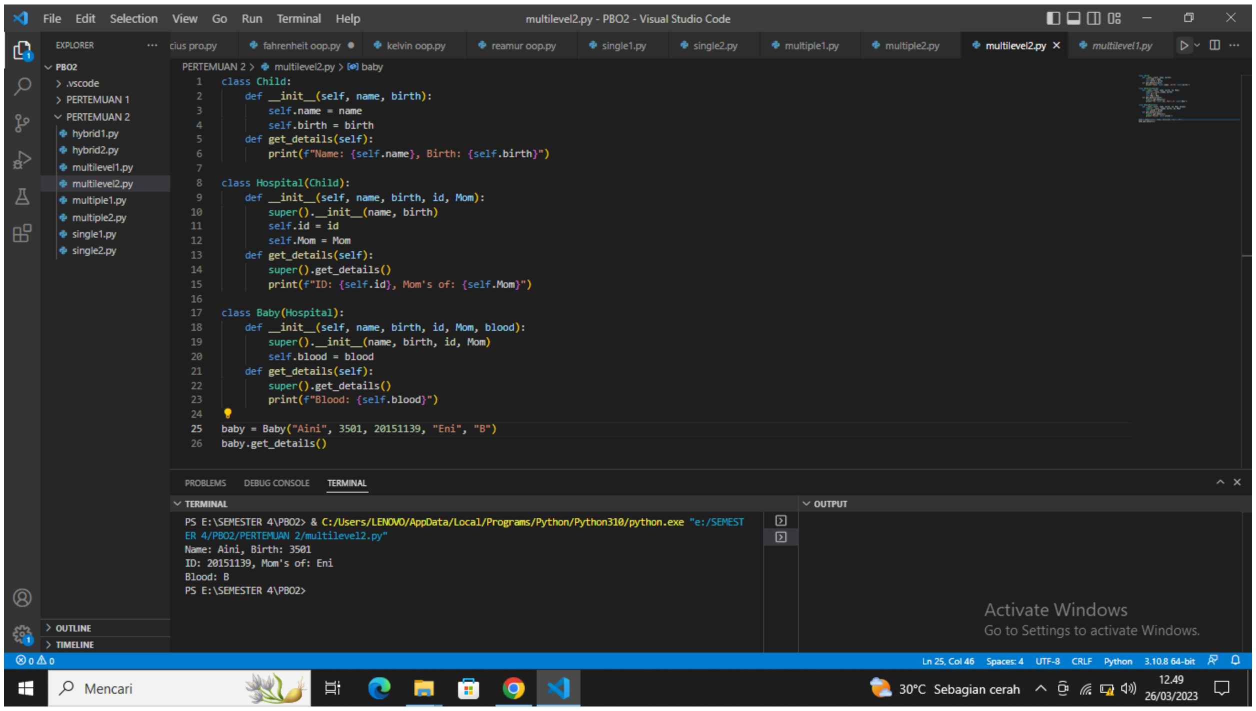
```
class Animal:
    def __init__(self, name):
        self.name = name
    def speak(self):
        print("The animal speaks")

class Cat(Animal):
    def __init__(self, name, breed):
        super().__init__(name)
        self.breed = breed
    def speak(self):
        print("The cat groaned")

class Persia(Cat):
    def __init__(self, name, breed, origin):
        super().__init__(name, breed)
        self.origin = origin
    def speak(self):
        print("The Persia", self.name , "is breed and origin in", self.origin)
persia = Persia("moly", "Persia", "Central Asia")
persia.speak()
```

```
PS E:\SEMESTER 4\PBO2> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python310/python.exe "e:/SEMEST
ER 4/PBO2/PERTEMUAN 2/multilevel1.py"
The Persia moly is breed and origin in Central Asia
PS E:\SEMESTER 4\PBO2> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python310/python.exe "e:/SEMEST
ER 4/PBO2/PERTEMUAN 2/multilevel1.py"
The Persia moly is breed and origin in Central Asia
PS E:\SEMESTER 4\PBO2> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python310/python.exe "e:/SEMEST
ER 4/PBO2/PERTEMUAN 2/multilevel1.py"
The Persia moly is breed and origin in Central Asia
PS E:\SEMESTER 4\PBO2>
```

b. Contoh 2:

```
class Child:
    def __init__(self, name, birth):
        self.name = name
        self.birth = birth
    def get_details(self):
        print(f"Name: {self.name}, Birth: {self.birth}")

class Hospital(Child):
    def __init__(self, name, birth, id, Mom):
        super().__init__(name, birth)
        self.id = id
        self.Mom = Mom
    def get_details(self):
        super().get_details()
        print(f"ID: {self.id}, Mom's of: {self.Mom}")

class Baby(Hospital):
    def __init__(self, name, birth, id, Mom, blood):
        super().__init__(name, birth, id, Mom)
        self.blood = blood
    def get_details(self):
        super().get_details()
        print(f"Blood: {self.blood}")

baby = Baby("Aini", 3501, 20151139, "Eni", "B")
baby.get_details()
```

Output:



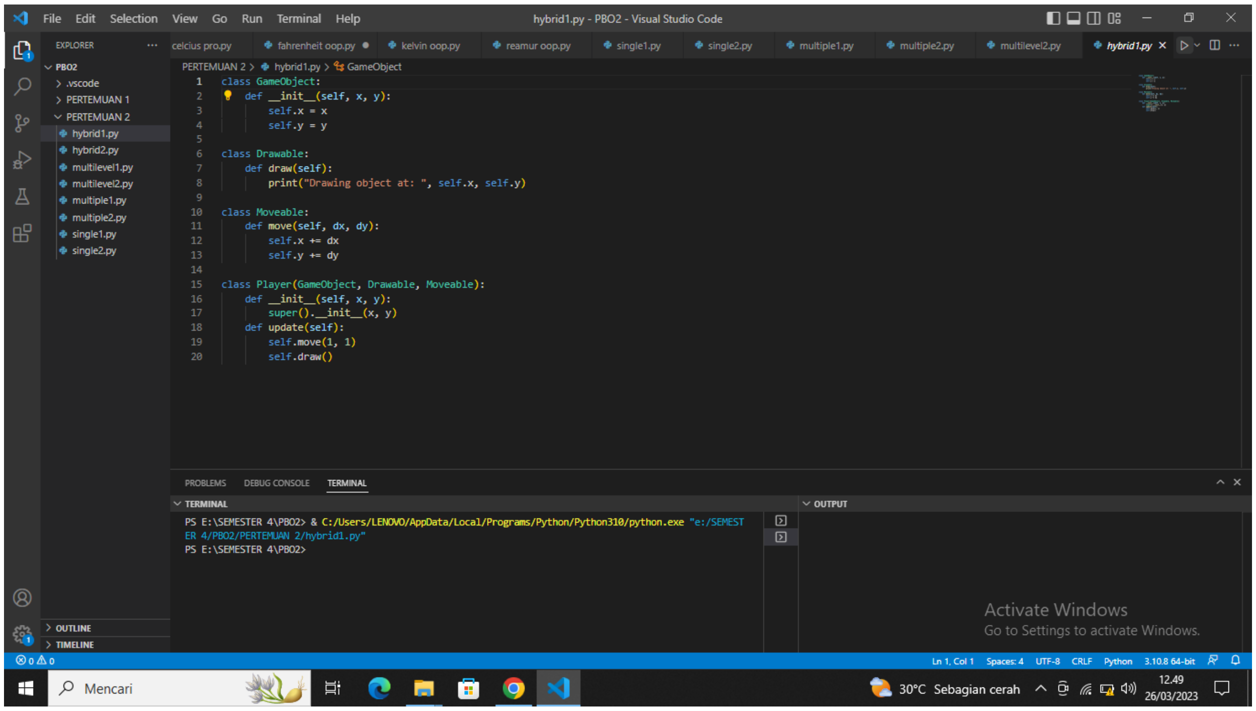
```python
class Child:
    def __init__(self, name, birth):
        self.name = name
        self.birth = birth
    def get_details(self):
        print(f"Name: {self.name}, Birth: {self.birth}")

class Hospital(Child):
    def __init__(self, name, birth, id, Mom):
        super().__init__(name, birth)
        self.id = id
        self.Mom = Mom
    def get_details(self):
        super().get_details()
        print(f"ID: {self.id}, Mom's of: {self.Mom}")

class Baby(Hospital):
    def __init__(self, name, birth, id, Mom, blood):
        super().__init__(name, birth, id, Mom)
        self.blood = blood
    def get_details(self):
        super().get_details()
        print(f"Blood: {self.blood}")

baby = Baby("Aini", 3501, 20151139, "Eni", "B")
baby.get_details()
```

```
PS E:\SEMESTER 4\PBO2> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python310/python.exe "e:/SEMEST
ER 4/PBO2/PERTEMUAN 2/multilevel2.py"
Name: Aini, Birth: 3501
ID: 20151139, Mom's of: Eni
Blood: B
PS E:\SEMESTER 4\PBO2>
```

5. **Hybrid Inheritance :**
   a. Contoh 1:

```python
class GameObject:
    def __init__(self, x, y):
        self.x = x
        self.y = y

class Drawable:
    def draw(self):
        print("Drawing object at: ", self.x, self.y)

class Moveable:
    def move(self, dx, dy):
        self.x += dx
        self.y += dy

class Player(GameObject, Drawable, Moveable):
    def __init__(self, x, y):
        super().__init__(x, y)
    def update(self):
        self.move(1, 1)
        self.draw()
```

Output:



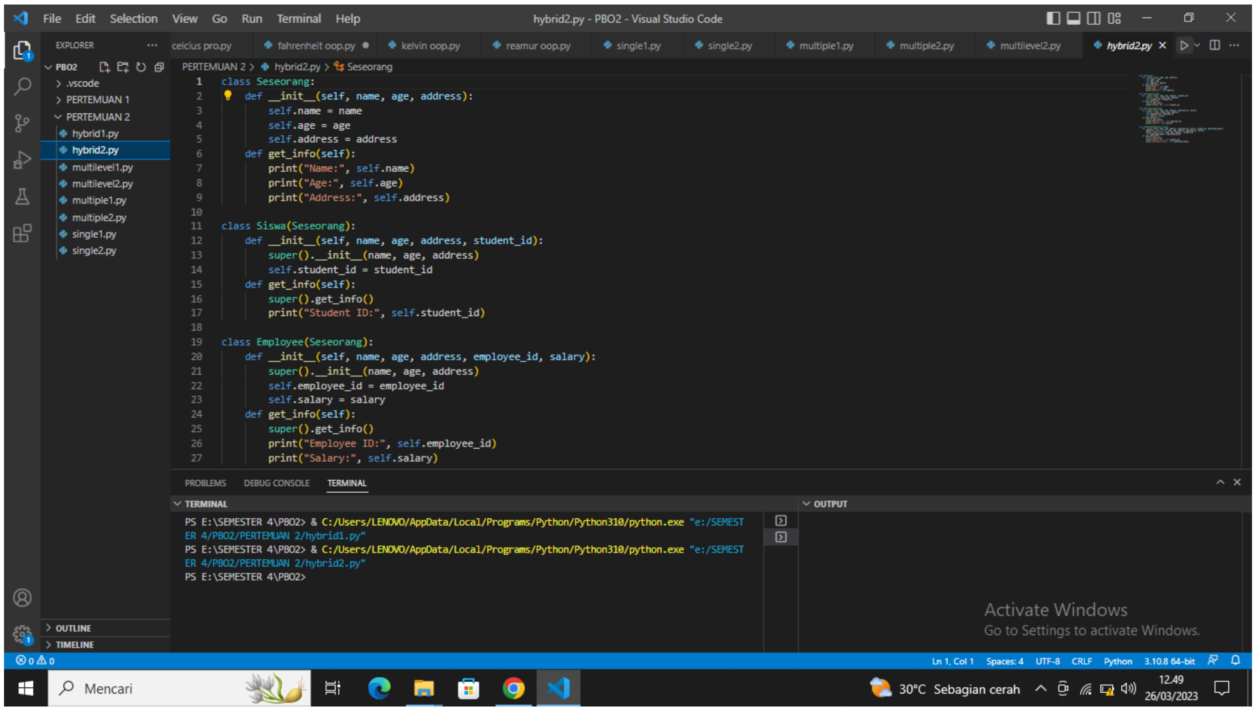
```
class GameObject:
    def __init__(self, x, y):
        self.x = x
        self.y = y

class Drawable:
    def draw(self):
        print("Drawing object at: ", self.x, self.y)

class Moveable:
    def move(self, dx, dy):
        self.x += dx
        self.y += dy

class Player(GameObject, Drawable, Moveable):
    def __init__(self, x, y):
        super().__init__(x, y)
    def update(self):
        self.move(1, 1)
        self.draw()
```

```
PS E:\SEMESTER 4\PBO2> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python310/python.exe "e:/SEMEST
ER 4/PBO2/PERTEMUAN 2/hybrid1.py"
PS E:\SEMESTER 4\PBO2>
```

b. Contoh 2:

```
class Seseorang:
    def __init__(self, name, age, address):
        self.name = name
        self.age = age
        self.address = address
    def get_info(self):
        print("Name:", self.name)
        print("Age:", self.age)
        print("Address:", self.address)

class Siswa(Seseorang):
    def __init__(self, name, age, address, student_id):
        super().__init__(name, age, address)
        self.student_id = student_id
    def get_info(self):
        super().get_info()
        print("Student ID:", self.student_id)

class Employee(Seseorang):
    def __init__(self, name, age, address, employee_id, salary):
        super().__init__(name, age, address)
        self.employee_id = employee_id
        self.salary = salary
    def get_info(self):
        super().get_info()
        print("Employee ID:", self.employee_id)
        print("Salary:", self.salary)

class Pengarang(Employee, Siswa):
    def __init__(self, name, age, address, employee_id, salary, student_id, published_books):
        Employee.__init__(self, name, age, address, employee_id, salary)
        Siswa.__init__(self, name, age, address, student_id)
        self.published_books = published_books
    def get_info(self):
        super().get_info()
        print("Student ID:", self.student_id)
        print("Published Books:", self.published_books)
```

Output:



```
class Seseorang:
    def __init__(self, name, age, address):
        self.name = name
        self.age = age
        self.address = address
    def get_info(self):
        print("Name:", self.name)
        print("Age:", self.age)
        print("Address:", self.address)

class Siswa(Seseorang):
    def __init__(self, name, age, address, student_id):
        super().__init__(name, age, address)
        self.student_id = student_id
    def get_info(self):
        super().get_info()
        print("Student ID:", self.student_id)

class Employee(Seseorang):
    def __init__(self, name, age, address, employee_id, salary):
        super().__init__(name, age, address)
        self.employee_id = employee_id
        self.salary = salary
    def get_info(self):
        super().get_info()
        print("Employee ID:", self.employee_id)
        print("Salary:", self.salary)
```

```
PS E:\SEMESTER 4\PBO2> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python310/python.exe "e:/SEMEST
ER 4/PBO2/PERTEMUAN 2/hybrid1.py"
PS E:\SEMESTER 4\PBO2> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python310/python.exe "e:/SEMEST
ER 4/PBO2/PERTEMUAN 2/hybrid2.py"
PS E:\SEMESTER 4\PBO2>
```