

LAPORAN PRAKTIKUM

PEMROGRAMAN BERORIENTASI OBJEK LANJUT

2023



Prepared By:

Dewi Alvi Nurfadilah
210511085
TI21B/R2

Soal :

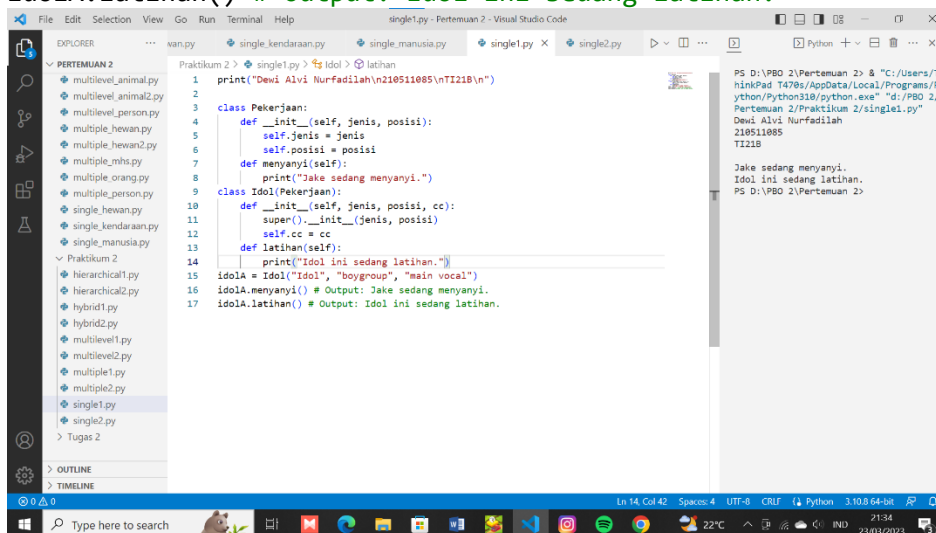
1. Buatlah masing-masing 2 contoh jenis pewarisan di luar dari contoh yang telah diberikan, beri nama:
single1.py, single2.py,
multiple1.py, multiple2.py,
hierarchical1.py, hierarchical2.py,
multilevel1.py, multilevel2,
hybrid1.py, hybrid2.py

Jawaban :

Single1 :

```
print("Dewi Alvi Nurfadilah\n210511085\nTI21B\n")
```

```
class Pekerjajaan:
    def __init__(self, jenis, posisi):
        self.jenis = jenis
        self.posisi = posisi
    def menyanyi(self):
        print("Jake sedang menyanyi.")
class Idol(Pekerjajaan):
    def __init__(self, jenis, posisi, cc):
        super().__init__(jenis, posisi)
        self.cc = cc
    def latihan(self):
        print("Idol ini sedang latihan.")
idola = Idol("Idol", "boygroup", "main vocal")
idola.menyanyi() # Output: Jake sedang menyanyi.
idola.latihan() # Output: Idol ini sedang latihan.
```



Single2 :

```
print("Dewi Alvi Nurfadilah\n210511085\nTI21B\n")
```

```
class Film:
```

```
    def __init__(self, genre, platform):
        self.genre = genre
        self.platform = platform
    def menonton(self):
        print("V sedang menonton.")
```

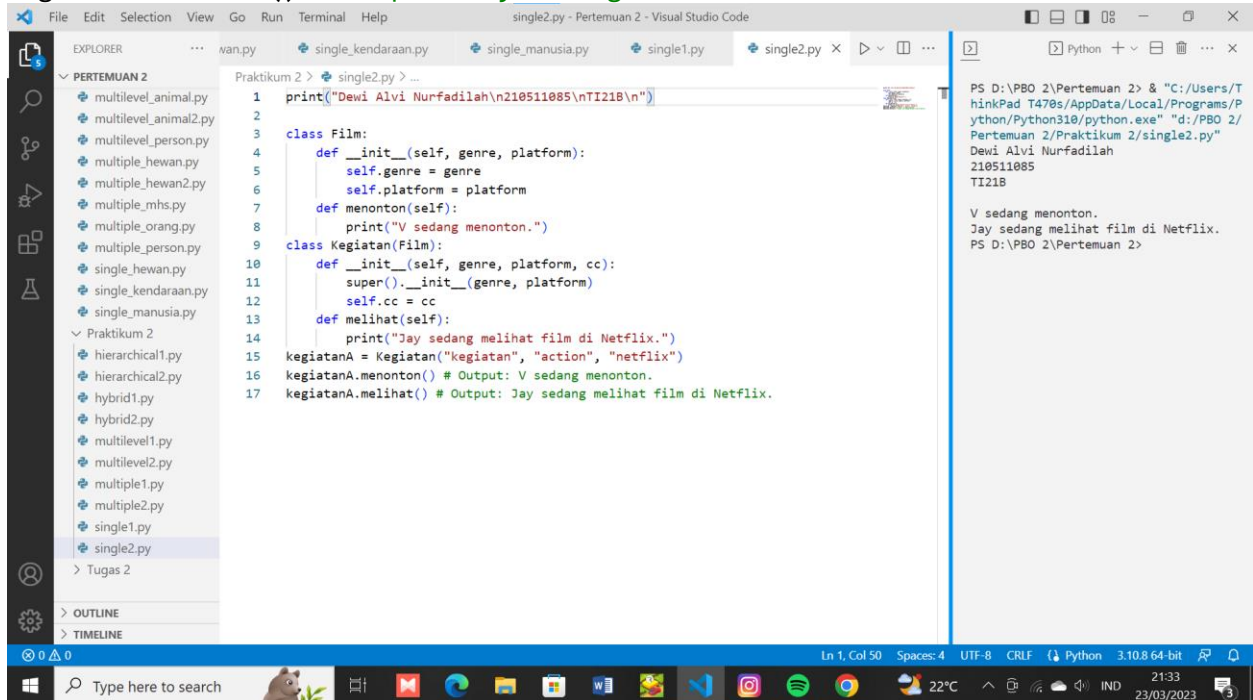
```
class Kegiatan(Film):
```

```
    def __init__(self, genre, platform, cc):
        super().__init__(genre, platform)
        self.cc = cc
    def melihat(self):
        print("Jay sedang melihat film di Netflix.")
```

```
kegiatanA = Kegiatan("kegiatan", "action", "netflix")
```

```
kegiatanA.menonton() # Output: V sedang menonton.
```

```
kegiatanA.melihat() # Output: Jay sedang melihat film di Netflix.
```



```
File Edit Selection View Go Run Terminal Help
single2.py - Pertemuan 2 - Visual Studio Code

EXPLORER
PERTEMUAN 2
  multilevel_animal.py
  multilevel_animal2.py
  multilevel_person.py
  multiple_hewan.py
  multiple_hewan2.py
  multiple_mhs.py
  multiple_orang.py
  multiple_person.py
  single_hewan.py
  single_kendaraan.py
  single_manusia.py
  Praktikum 2
    hierarchical1.py
    hierarchical2.py
    hybrid1.py
    hybrid2.py
    multilevel1.py
    multilevel2.py
    multiple1.py
    multiple2.py
    single1.py
    single2.py
  Tugas 2
  > OUTLINE
  > TIMELINE

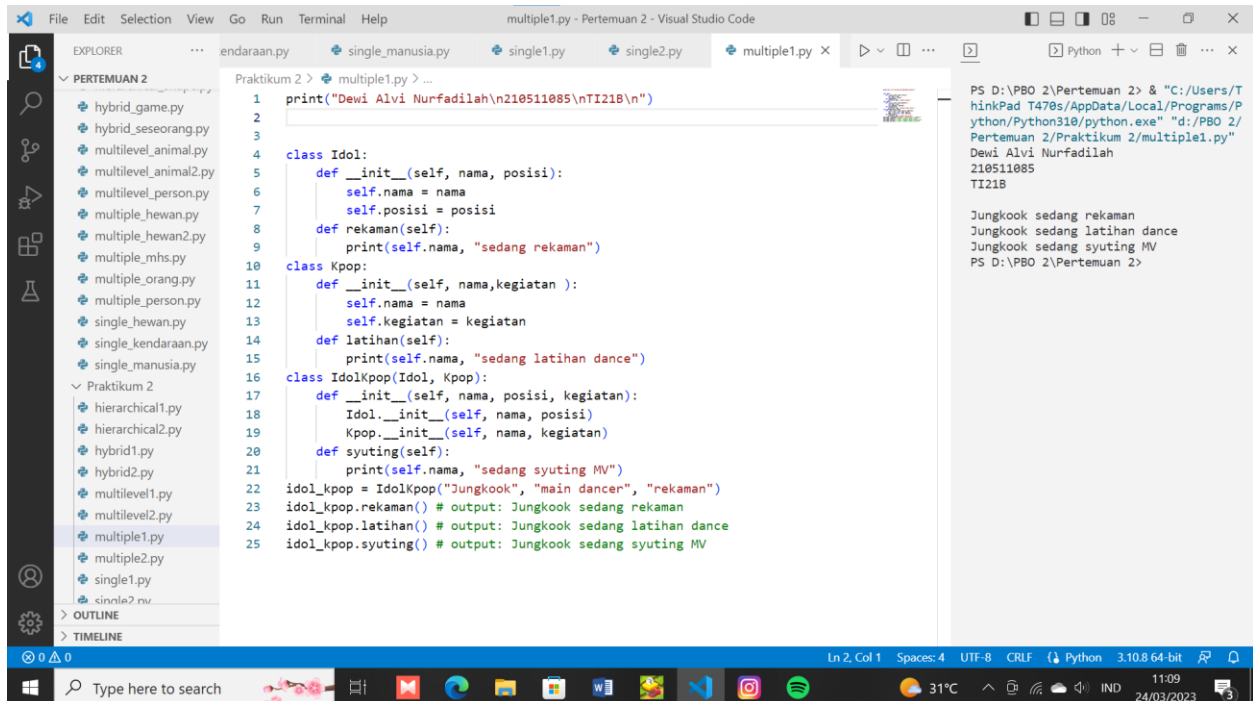
Praktikum 2 > single2.py > ...
1 print("Dewi Alvi Nurfadilah\n210511085\nTI21B\n")
2
3 class Film:
4     def __init__(self, genre, platform):
5         self.genre = genre
6         self.platform = platform
7     def menonton(self):
8         print("V sedang menonton.")
9 class Kegiatan(Film):
10    def __init__(self, genre, platform, cc):
11        super().__init__(genre, platform)
12        self.cc = cc
13    def melihat(self):
14        print("Jay sedang melihat film di Netflix.")
15 kegiatanA = Kegiatan("kegiatan", "action", "netflix")
16 kegiatanA.menonton() # Output: V sedang menonton.
17 kegiatanA.melihat() # Output: Jay sedang melihat film di Netflix.

PS D:\PBO 2\Pertemuan 2> & "C:/Users/T hinkPad T470s/AppData/Local/Programs/Python/Python310/python.exe" "d:/PBO 2/Pertemuan 2/Praktikum 2/single2.py"
Dewi Alvi Nurfadilah
210511085
TI21B

V sedang menonton.
Jay sedang melihat film di Netflix.
PS D:\PBO 2\Pertemuan 2>
```

Multiple1 :

```
print("Dewi Alvi Nurfadilah\n210511085\nTI21B\n")
class Idol:
    def __init__(self, nama, posisi):
        self.nama = nama
        self.posisi = posisi
    def rekaman(self):
        print(self.nama, "sedang rekaman")
class Kpop:
    def __init__(self, nama, kegiatan):
        self.nama = nama
        self.kegiatan = kegiatan
    def latihan(self):
        print(self.nama, "sedang latihan dance")
class IdolKpop(Idol, Kpop):
    def __init__(self, nama, posisi, kegiatan):
        Idol.__init__(self, nama, posisi)
        Kpop.__init__(self, nama, kegiatan)
    def syuting(self):
        print(self.nama, "sedang syuting MV")
idol_kpop = IdolKpop("Jungkook", "main dancer", "rekaman")
idol_kpop.rekaman() # output: Jungkook sedang rekaman
idol_kpop.latihan() # output: Jungkook sedang latihan dance
idol_kpop.syuting() # output: Jungkook sedang syuting MV
```



```
File Edit Selection View Go Run Terminal Help
multiple1.py - Pertemuan 2 - Visual Studio Code

EXPLORER
... endaraan.py single_manusia.py single1.py single2.py multiple1.py x
PERTEMUAN 2
  hybrid_game.py
  hybrid_seseorang.py
  multilevel_animal.py
  multilevel_animal2.py
  multilevel_person.py
  multiple_hewan.py
  multiple_hewan2.py
  multiple_mhs.py
  multiple_orang.py
  multiple_person.py
  single_hewan.py
  single_kendaraan.py
  single_manusia.py
  Praktikum 2
    hierarchical1.py
    hierarchical2.py
    hybrid1.py
    hybrid2.py
    multilevel1.py
    multilevel2.py
    multiple1.py
    multiple2.py
    single1.py
    sinle2 nu
  > OUTLINE
  > TIMELINE

Praktikum 2 > multiple1.py > ...
1 print("Dewi Alvi Nurfadilah\n210511085\nTI21B\n")
2
3
4 class Idol:
5     def __init__(self, nama, posisi):
6         self.nama = nama
7         self.posisi = posisi
8     def rekaman(self):
9         print(self.nama, "sedang rekaman")
10
11 class Kpop:
12     def __init__(self, nama, kegiatan):
13         self.nama = nama
14         self.kegiatan = kegiatan
15     def latihan(self):
16         print(self.nama, "sedang latihan dance")
17
18 class IdolKpop(Idol, Kpop):
19     def __init__(self, nama, posisi, kegiatan):
20         Idol.__init__(self, nama, posisi)
21         Kpop.__init__(self, nama, kegiatan)
22     def syuting(self):
23         print(self.nama, "sedang syuting MV")
24
25 idol_kpop = IdolKpop("Jungkook", "main dancer", "rekaman")
26 idol_kpop.rekaman() # output: Jungkook sedang rekaman
27 idol_kpop.latihan() # output: Jungkook sedang latihan dance
28 idol_kpop.syuting() # output: Jungkook sedang syuting MV

PS D:\PBO 2\Pertemuan 2> & "C:/Users/T hinkPad T470s/AppData/Local/Programs/Python/Python310/python.exe" "d:/PBO 2/Pertemuan 2/Praktikum 2/multiple1.py"
Dewi Alvi Nurfadilah
210511085
TI21B

Jungkook sedang rekaman
Jungkook sedang latihan dance
Jungkook sedang syuting MV
PS D:\PBO 2\Pertemuan 2>
```

Multiple2 :

```
print("Dewi Alvi Nurfadilah\n210511085\nTI21B\n")
```

```
class Person:
```

```
    def __init__(self, nama, umur):
```

```
        self.nama = nama
```

```
        self.umur = umur
```

```
    def display_info(self):
```

```
        print(f>Nama: {self.nama}")
```

```
        print(f"Umur: {self.umur}")
```

```
class Pekerjaan(Person):
```

```
    def __init__(self, nama, umur, pekerjaan):
```

```
        super().__init__(nama, umur, pekerjaan)
```

```
        self.pekerjaan = pekerjaan
```

```
    def display_info(self):
```

```
        super().display_info()
```

```
        print(f"Pekerjaan: {self.pekerjaan}")
```

```
class Posisi(Person):
```

```
    def __init__(self, nama, umur, posisi):
```

```
        super().__init__(nama, umur)
```

```
        self.posisi = posisi
```

```
    def display_info(self):
```

```
        super().display_info()
```

```
        print(f"Posisi: {self.posisi}")
```

```
class PekerjaanPosisi(Pekerjaan, Posisi):
```

```
    def __init__(self, nama, umur, pekerjaan, posisi):
```

```
        Pekerjaan.__init__(self, nama, umur, pekerjaan)
```

```
        Posisi.__init__(self, nama, umur, posisi)
```

```
    def display_info(self):
```

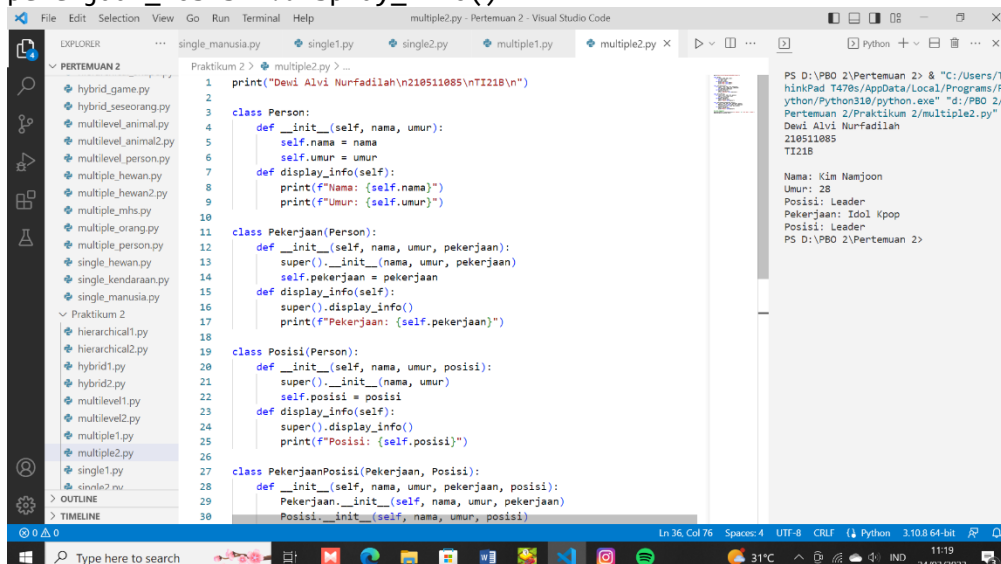
```
        super().display_info()
```

```
        print(f"Posisi: {self.posisi}")
```

contoh penggunaan

```
pekerjaan_PosisiA = PekerjaanPosisi("Kim Namjoon", 28, "Idol Kpop", "Leader")
```

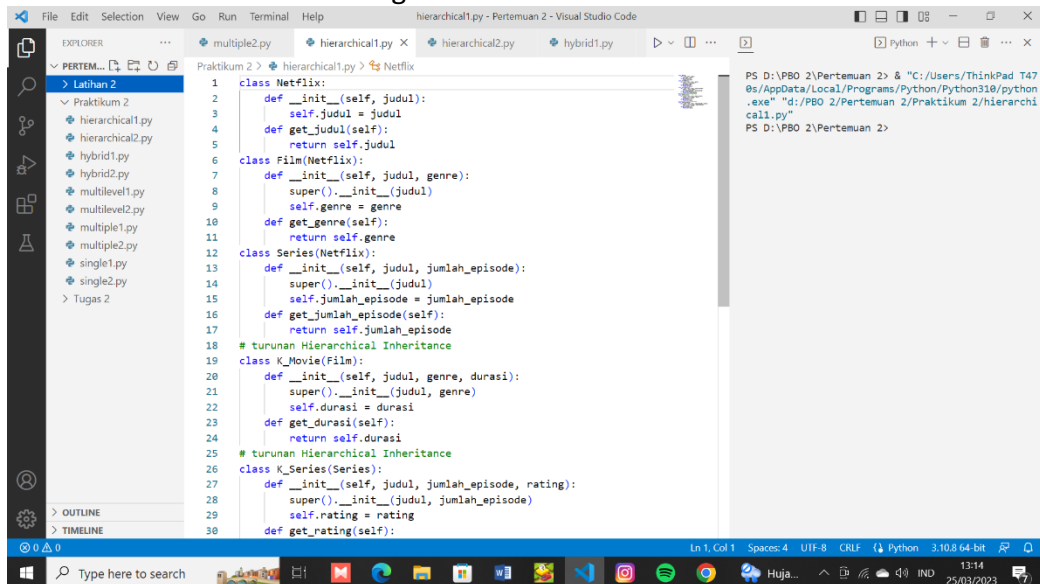
```
pekerjaan_PosisiA.display_info()
```



The screenshot shows a Visual Studio Code window with a file explorer on the left, a code editor in the center, and a terminal on the right. The code editor displays the Python code from the previous blocks. The terminal shows the output of the script, which is the same as the output shown in the previous blocks. The status bar at the bottom indicates the file is at line 36, column 78, with 4 spaces, using the UTF-8 encoding and CRLF line endings. The Python version is 3.10.8 64-bit.

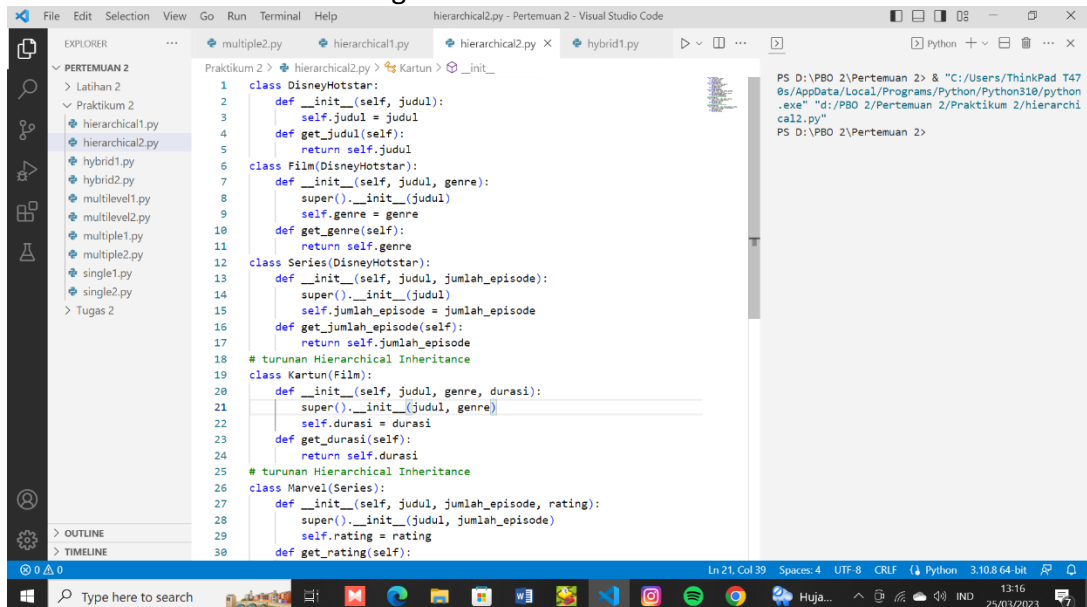
Hierarchical1 :

```
class Netflix:
    def __init__(self, judul):
        self.judul = judul
    def get_judul(self):
        return self.judul
class Film(Netflix):
    def __init__(self, judul, genre):
        super().__init__(judul)
        self.genre = genre
    def get_genre(self):
        return self.genre
class Series(Netflix):
    def __init__(self, judul, jumlah_episode):
        super().__init__(judul)
        self.jumlah_episode = jumlah_episode
    def get_jumlah_episode(self):
        return self.jumlah_episode
# turunan Hierarchical Inheritance
class K_Movie(Film):
    def __init__(self, judul, genre, durasi):
        super().__init__(judul, genre)
        self.durasi = durasi
    def get_durasi(self):
        return self.durasi
# turunan Hierarchical Inheritance
class K_Series(Series):
    def __init__(self, judul, jumlah_episode, rating):
        super().__init__(judul, jumlah_episode)
        self.rating = rating
    def get_rating(self):
        return self.rating
```



Hierarchical2 :

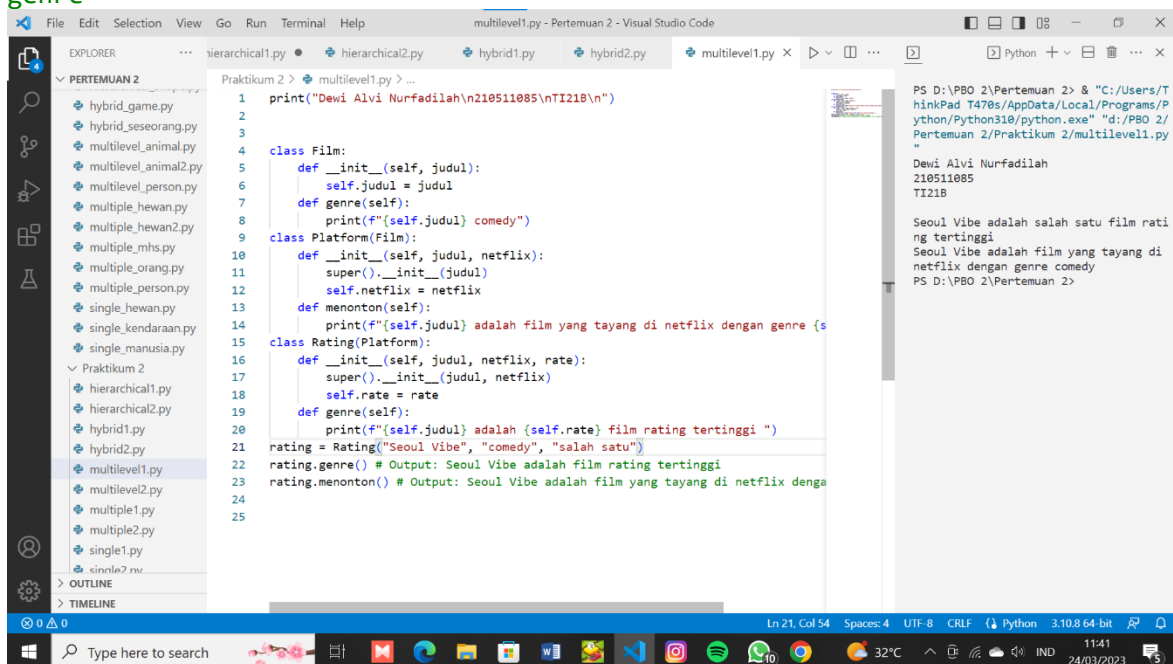
```
class DisneyHotstar:
    def __init__(self, judul):
        self.judul = judul
    def get_judul(self):
        return self.judul
class Film(DisneyHotstar):
    def __init__(self, judul, genre):
        super().__init__(judul)
        self.genre = genre
    def get_genre(self):
        return self.genre
class Series(DisneyHotstar):
    def __init__(self, judul, jumlah_episode):
        super().__init__(judul)
        self.jumlah_episode = jumlah_episode
    def get_jumlah_episode(self):
        return self.jumlah_episode
# turunan Hierarchical Inheritance
class Kartun(Film):
    def __init__(self, judul, genre, durasi):
        super().__init__(judul, genre)
        self.durasi = durasi
    def get_durasi(self):
        return self.durasi
# turunan Hierarchical Inheritance
class Marvel(Series):
    def __init__(self, judul, jumlah_episode, rating):
        super().__init__(judul, jumlah_episode)
        self.rating = rating
    def get_rating(self):
        return self.rating
```



Multilevel1 :

```
print("Dewi Alvi Nurfadilah\n210511085\nTI21B\n")
```

```
class Film:
    def __init__(self, judul):
        self.judul = judul
    def genre(self):
        print(f"{self.judul} comedy")
class Platform(Film):
    def __init__(self, judul, netflix):
        super().__init__(judul)
        self.netflix = netflix
    def menonton(self):
        print(f"{self.judul} adalah film yang tayang di netflix dengan genre {self.netflix}")
class Rating(Platform):
    def __init__(self, judul, netflix, rate):
        super().__init__(judul, netflix)
        self.rate = rate
    def genre(self):
        print(f"{self.judul} adalah {self.rate} film rating tertinggi ")
rating = Rating("Seoul Vibe", "comedy", "salah satu")
rating.genre() # Output: Seoul Vibe adalah film rating tertinggi
rating.menonton() # Output: Seoul Vibe adalah film yang tayang di netflix dengan genre
```



The screenshot shows a Visual Studio Code window with a file explorer on the left, a code editor in the center, and a terminal on the right. The file explorer shows a project named 'PERTEMUAN 2' with several Python files. The code editor displays the Python script from the previous block. The terminal shows the output of the script, which includes the file path, the name 'Dewi Alvi Nurfadilah', the ID '210511085', the course 'TI21B', and the genre 'comedy' for the film 'Seoul Vibe'.

```
PS D:\PBO 2\Pertemuan 2> & "C:/Users/T  
hinkPad T470s/AppData/Local/Programs/P  
ython/Python310/python.exe" "d:/PBO 2/  
Pertemuan 2/Praktikum 2/multilevel1.py"  
Dewi Alvi Nurfadilah  
210511085  
TI21B  
  
Seoul Vibe adalah salah satu film rati  
ng tertinggi  
Seoul Vibe adalah film yang tayang di  
netflix dengan genre comedy  
PS D:\PBO 2\Pertemuan 2>
```


Multilevel2 :

```
print("Dewi Alvi Nurfadilah\n210511085\nTI21B\n")
```

```
class Drakor:  
    def drakor_info(self):  
        print("Drakor merupakan serial drama dari Korea Selatan.")
```

```
class Judul:  
    def judul_info(self):  
        print("The Glory adalah drakor dengan rating tertinggi saat ini.")
```

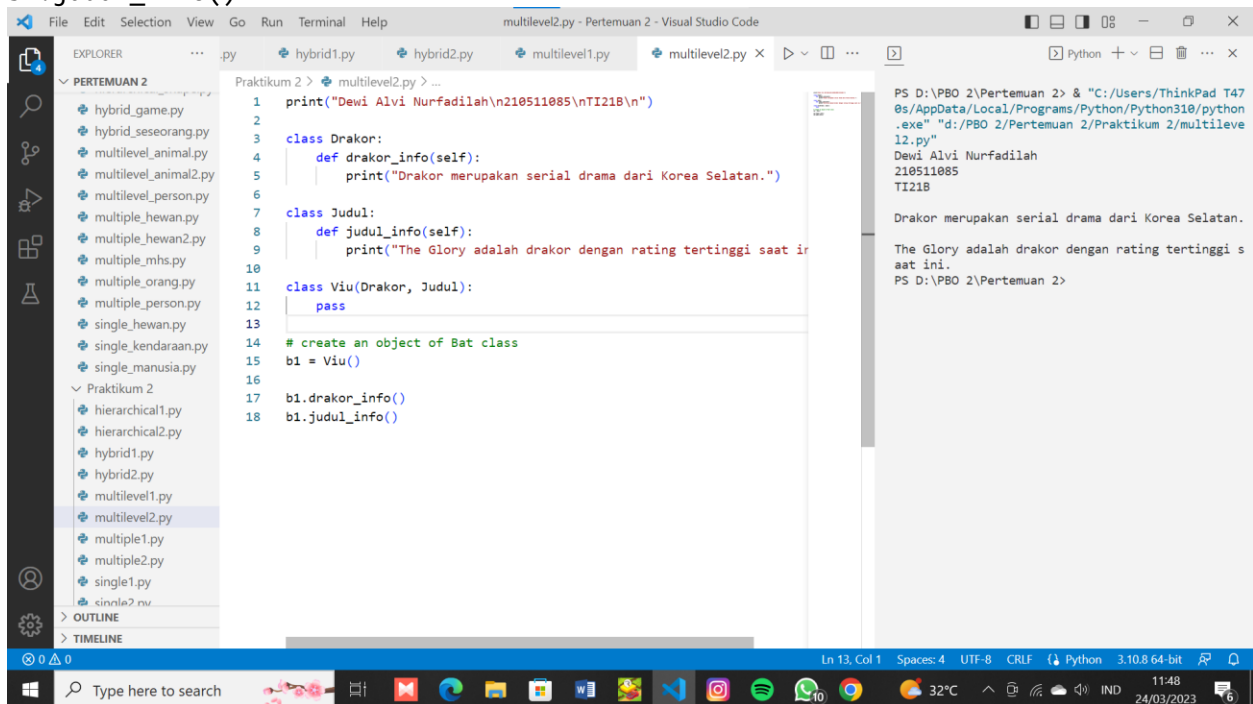
```
class Viu(Drakor, Judul):  
    pass
```

create an object of Bat class

```
b1 = Viu()
```

```
b1.drakor_info()
```

```
b1.judul_info()
```



```
1 print("Dewi Alvi Nurfadilah\n210511085\nTI21B\n")  
2  
3 class Drakor:  
4     def drakor_info(self):  
5         print("Drakor merupakan serial drama dari Korea Selatan.")  
6  
7 class Judul:  
8     def judul_info(self):  
9         print("The Glory adalah drakor dengan rating tertinggi saat ini.")  
10  
11 class Viu(Drakor, Judul):  
12     pass  
13  
14 # create an object of Bat class  
15 b1 = Viu()  
16  
17 b1.drakor_info()  
18 b1.judul_info()
```

PS D:\PBO 2\Pertemuan 2> & "C:/Users/ThinkPad T470s/AppData/Local/Programs/Python/Python310/python.exe" "d:/PBO 2/Pertemuan 2/Praktikum 2/multilevel2.py"
Dewi Alvi Nurfadilah
210511085
TI21B

Drakor merupakan serial drama dari Korea Selatan.

The Glory adalah drakor dengan rating tertinggi saat ini.
PS D:\PBO 2\Pertemuan 2>

Hybrid1 :

```
class Youtube:
    def __init__(self, nama, jenis, ):
        self.nama = nama
        self.jenis = jenis
    def get_info(self):
        print("nama:", self.nama)
        print("jenis:", self.jenis)
# Single Inheritance
class Channel(Youtube):
    def __init__(self, nama, jenis, subscriber):
        super().__init__(nama, jenis, )
        self.subscriber = subscriber
    def get_info(self):
        super().get_info()
        print("Jumlah:", self.subscriber)
# Single Inheritance
class Konten(Youtube):
    def __init__(self, nama, jenis, konten, viewers):
        super().__init__(nama, jenis, )
        self.konten = konten
        self.viewers = viewers
    def get_info(self):
        super().get_info()
        print("konten:", self.konten)
        print("viewers:", self.viewers)
# Multiple Inheritance
class Trending(Konten, Channel):
    def __init__(self, nama, jenis, konten, viewers, subscriber,
trending):
        Konten.__init__(self, nama, jenis, konten, viewers)
        Channel.__init__(self, nama, jenis, subscriber)
        self.trending = trending
    def get_info(self):
        super().get_info()
        print("Berapa subscriber:", self.subscriber)
        print("Trending ke:", self.trending)
```

Hybrid2 :

```
class Makanan:
    def __init__(self, nama, jenis, ):
        self.nama = nama
        self.jenis = jenis
    def get_info(self):
        print("nama:", self.nama)
        print("jenis:", self.jenis)
# Single Inheritance
class Asal(Makanan):
    def __init__(self, nama, jenis, daerah):
        super().__init__(nama, jenis, )
        self.daerah = daerah
    def get_info(self):
        super().get_info()
        print("Berasal dari:", self.daerah)
# Single Inheritance
class Variasi(Makanan):
    def __init__(self, nama, jenis, variasi, harga):
        super().__init__(nama, jenis, )
        self.variasi = variasi
        self.harga = harga
    def get_info(self):
        super().get_info()
        print("Variasi:", self.variasi)
        print("Harga:", self.harga)
# Multiple Inheritance
class Rasa(Variasi, Asal):
    def __init__(self, nama, jenis, variasi, harga, daerah,
rasa):
        Variasi.__init__(self, nama, jenis, variasi, harga)
        Asal.__init__(self, nama, jenis, daerah)
        self.rasa = rasa
    def get_info(self):
        super().get_info()
        print("Berasal dari Daerah:", self.daerah)
        print("Rasanya:", self.rasa)
```

Hybrid 1 & 2 :

The image displays two screenshots of a Visual Studio Code editor window, showing Python code for class inheritance. The top screenshot shows the 'hybrid1.py' file, and the bottom screenshot shows the 'hybrid2.py' file.

Top Screenshot (hybrid1.py):

```
Praktikum 2 > hybrid1.py > Trending > get_info
1 class Youtube:
2     def __init__(self, nama, jenis, ):
3         self.nama = nama
4         self.jenis = jenis
5     def get_info(self):
6         print("nama:", self.nama)
7         print("jenis:", self.jenis)
8     # Single Inheritance
9     class Channel(Youtube):
10        def __init__(self, nama, jenis, subscriber):
11            super().__init__(nama, jenis, )
12            self.subscriber = subscriber
13        def get_info(self):
14            super().get_info()
15            print("Jumlah:", self.subscriber)
16    # Single Inheritance
17    class Konten(Youtube):
18        def __init__(self, nama, jenis, konten, viewers):
19            super().__init__(nama, jenis, )
20            self.konten = konten
21            self.viewers = viewers
22        def get_info(self):
23            super().get_info()
24            print("konten:", self.konten)
25            print("viewers:", self.viewers)
26    # Multiple Inheritance
27    class Trending(Konten, Channel):
28        def __init__(self, nama, jenis, konten, viewers, subscriber,
29            trending):
30            Konten.__init__(self, nama, jenis, konten, viewers)
```

Bottom Screenshot (hybrid2.py):

```
Praktikum 2 > hybrid2.py > Asal > __init__
1 class Makanan:
2     def __init__(self, nama, jenis, ):
3         self.nama = nama
4         self.jenis = jenis
5     def get_info(self):
6         print("nama:", self.nama)
7         print("jenis:", self.jenis)
8     # Single Inheritance
9     class Asal(Makanan):
10        def __init__(self, nama, jenis, daerah):
11            super().__init__(nama, jenis, )
12            self.daerah = daerah
13        def get_info(self):
14            super().get_info()
15            print("Berasal dari:", self.daerah)
16    # Single Inheritance
17    class Variasi(Makanan):
18        def __init__(self, nama, jenis, variasi, harga):
19            super().__init__(nama, jenis, )
20            self.variasi = variasi
21            self.harga = harga
22        def get_info(self):
23            super().get_info()
24            print("Variasi:", self.variasi)
25            print("Harga:", self.harga)
26    # Multiple Inheritance
27    class Rasa(Variasi, Asal):
28        def __init__(self, nama, jenis, variasi, harga, daerah,
29            rasa):
30            Variasi.__init__(self, nama, jenis, variasi, harga)
```