

LAPORAN PRAKTIKUM

PEMROGRAMAN BERORIENTASI OBJEK LANJUT

2023



Prepared By:

NAMA : ZIAROTUN NIMAH
NIM : 221511016
KELAS : TI21K (K2 KONVERSI)

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:

1. Single Inheritance

A. Single1 Inheritance

- Source Code

```
1 class Hero:
2     def __init__(self,nama,damage):
3         self.nama = nama
4         self.damage = damage
5
6     def bergerak(self):
7         print(self.damage, "damage")
8
9 class Fighter(Hero):
10    def __init__(self, nama, damage, item ):
11        super().__init__(nama,damage)
12        self.item = item
13
14    def menyerang(self):
15        print(self.nama, "menyakitkan")
16
17 fighterA = Fighter("Aldos", 4000, "Cursed Helmet")
18 fighterA.bergerak()
19 fighterA.menyerang()
```

- Output

```
PS C:\KULIAH UMC\PBO LANJUT\INHERITANCE\praktikum2> py single1.py
4000 damage
Aldos menyakitkan
```

B. Single2 Inheritance

- Source Code

```
1 class Televisi:
2     def __init__(self, merk):
3         self.merk = merk
4
5     def ditonton(self):
6         print(f"{self.merk} sedang ditonton")
7
8 class Siaran(Televisi):
9     def __init__(self, merk, nama_siaran, lokasi, nama_program_acara ):
10        super().__init__(merk)
11        self.nama_siaran = nama_siaran
12        self.lokasi = lokasi
13        self.nama_program_acara = nama_program_acara
14
15    def menonton(self):
16        print(f"Merk Televisi: {self.merk}\n Siaran: {self.nama_siaran}\n Lokasi: {self.lokasi}\n Program Acara: {self.nama_program_acara}")
17
18 siaranA = Siaran("Televisi Merk Samsung","Trans 7","Jakarta","Lapor Pak")
19 siaranA.ditonton()
20 siaranA.menonton()
```

- Output

```
PS C:\KULIAH UMC\PBO LANJUT\INHERITANCE\praktikum2> py single2.py
Televisi Merk Samsung sedang ditonton
Merk Televisi: Televisi Merk Samsung
Siaran: Trans 7
Lokasi: Jakarta
Program Acara: Laporan Pak
```

2. Multiple Inheritance

A. Multiple

- Source Code

```
1 class Makeup:
2     def __init__(self, nama_cushion):
3         self.nama_cushion = nama_cushion
4
5     def digunakan(self):
6         print(f"{self.nama_cushion} sedang digunakan")
7
8 class Cushion:
9     def __init__(self, ukuran, shade):
10         self.ukuran = ukuran
11         self.shade = shade
12
13     def mengaplikasikan(self):
14         print(f"Mengaplikasikan Cushion {self.shade} ")
15
16 class MakeupCushion(Makeup, Cushion):
17     def __init__(self, nama_cushion, shade, ukuran):
18         Makeup.__init__(self, nama_cushion)
19         Cushion.__init__(self, ukuran, shade)
20
21     def memakai(self):
22         print(f"Memakai Cushion {self.nama_cushion}, Shade {self.shade}, Ukuran {self.ukuran}")
23
24 makeup_cushion = MakeupCushion("Pixy", "Shade 02", "Kecil")
25 makeup_cushion.digunakan()
26 makeup_cushion.mengaplikasikan()
27 makeup_cushion.memakai()
```

- Output

```
PS C:\KULIAH UMC\PBO LANJUT\INHERITANCE\praktikum2> py multiple1.py
Pixy sedang digunakan
Mengaplikasikan Cushion Shade 02
Memakai Cushion Pixy, Shade Shade 02, Ukuran Kecil
```

B. Multiple2

- Source Code

```
1 class Team:
2     def __init__(self, nama_team):
3         self.nama_team = nama_team
4
5     def bermain(self):
6         print(f"Team {self.nama_team} sedang bermain")
7
8 class TipeHero:
9     def __init__(self, tipe_hero):
10        self.tipe_hero = tipe_hero
11
12    def menggunakan(self):
13        print(f"Menggunakan {self.tipe_hero}")
14
15 class Hero(Team, TipeHero):
16     def __init__(self, nama_team, tipe_hero, damage):
17         Team.__init__(self, nama_team)
18         TipeHero.__init__(self, tipe_hero)
19         self.damage = damage
20
21     def dipilih(self):
22         print(f"Hero {self.tipe_hero} sedang dipilih dengan damage yang {self.damage}")
23
24 teamA = Hero("Barracuda", "Fighter", "Kuat")
25 teamA.bermain()
26 teamA.menggunakan()
27 teamA.dipilih()
```

- Output

```
PS C:\KULIAH UMC\PB0 LANJUT\INHERITANCE\praktikum2> py multiple2.py
Team Barracuda sedang bermain
Menggunakan Fighter
Hero Fighter sedang dipilih dengan damage yang Kuat
```

3. Hierarchical Inheritance

A. Hierarchical1

- Source Code

```
1 class Makeup:
2     def __init__(self, nama):
3         self.nama = nama
4     def get_nama(self):
5         return self.nama
6
7 class Cushion(Makeup):
8     def __init__(self, nama):
9         super().__init__(nama)
10    def get_nama(self):
11        return self.nama
12
13 class Lipstick(Makeup):
14     def __init__(self, nama, shade):
15         super().__init__(nama)
16         self.shade = shade
17     def get_shade(self):
18         return self.shade
19
20 #Hierarchical Inheritance
21 class LipstickWardah(Lipstick):
22     def __init__(self, nama, shade, harga):
23         super().__init__(nama, shade)
24         self.harga = harga
25     def get_harga(self):
26         return self.harga
27
28 lipstickA = LipstickWardah("Wardah", "12", "RP 30000")
29 print(f"Lipstick {Makeup.get_nama(lipstickA)} dengan shade {Lipstick.get_shade(lipstickA)} dan harganya {LipstickWardah.get_harga(lipstickA)}")
```

- Output

PS C:\KULIAH UMC\PBO LANJUT\INHERITANCE\praktikum2> py hierarchical1.py
Lipstick Wardah dengan shade 12 dan harganya RP 30000

B. Hierarchical2

- Source Code

```
1 class SmartPhone:
2     def __init__(self, nama):
3         self.nama = nama
4     def get_nama(self):
5         return self.nama
6
7 class Android(SmartPhone):
8     def __init__(self, nama, merk):
9         super().__init__(nama)
10        self.merk = merk
11    def get_merk(self):
12        return self.merk
13
14 class IOS(SmartPhone):
15     def __init__(self, nama, merk, kapasitas):
16         super().__init__(nama, merk)
17         self.kapasitas = kapasitas
18     def get_kapasitas(self):
19         return self.kapasitas
20
21 #Hierarchical Inheritance
22 class Samsung(Android):
23     def __init__(self, nama, merk, warna):
24         super().__init__(nama, merk)
25         self.warna = warna
26     def get_warna(self):
27         return self.warna
28
29 samsungA = Samsung("Samsung", "Galaxy J1", "Hitam")
30 print(f"{SmartPhone.get_nama(samsungA)} {Android.get_merk(samsungA)} Warna {Samsung.get_warna(samsungA)}")
```

- Output

PS C:\KULIAH UMC\PBO LANJUT\INHERITANCE\praktikum2> py hierarchical2.py
Samsung Galaxy J1 Warna Hitam

4. Multilevel Inheritance

A. Multilevel1

- Source Code

```
1 class Dosen:
2     def __init__(self, nama, umur, nip):
3         self.nama = nama
4         self.umur = umur
5         self.nip = nip
6
7     def informasi(self):
8         print(f>Nama: {self.nama}\nUmur: {self.umur}\nNIP: {self.nip}")
9
10 class Programmer(Dosen):
11     def __init__(self, nama, umur, nip, skill):
12         super().__init__(nama, umur, nip)
13         self.skill = skill
14
15     def keahlian(self):
16         print(f"Skill: {self.skill}")
17
18 class Salary(Programmer):
19     def __init__(self, nama, umur, nip, skill, gaji):
20         super().__init__(nama, umur, nip, skill)
21         self.gaji = gaji
22
23     def penghasilan(self):
24         print(f"Gaji: {self.gaji}")
25
26 salaryA = Salary("Lubis", "40 Tahun", "123456", "Web Developer", "RP 5000000")
27 salaryA.informasi()
28 salaryA.keahlian()
29 salaryA.penghasilan()
```

- Output

PS C:\KULIAH UMC\PBO LANJUT\INHERITANCE\praktikum2> py multilevel1.py

Nama: Lubis

Umur: 40 Tahun

NIP: 123456

Skill: Web Developer

Gaji: RP 5000000

B. Multilevel2

- Source Code

```
1 class Kendaraan:
2     def __init__(self, brand):
3         self.brand = brand
4     def get_brand(self):
5         return self.brand
6
7 class Sepeda(Kendaraan):
8     def __init__(self, brand, merk):
9         super().__init__(brand)
10        self.merk = merk
11    def get_merk(self):
12        return self.merk
13
14 class Motor(Kendaraan):
15     def __init__(self, brand, merk, tahun):
16         super().__init__(brand, merk)
17         self.tahun = tahun
18    def get_tahun(self):
19        return self.tahun
20
21 class SepedaListrik(Sepeda):
22     def __init__(self, brand, merk, tahun, kapasitas_baterai):
23         super().__init__(brand, merk)
24         self.kapasitas_baterai = kapasitas_baterai
25    def get_kapasitas_baterai(self):
26        return self.kapasitas_baterai
27
28 sepedalistrikA = SepedaListrik("Goda", "Sepeda Listrik Goda 140 New", "Tahun 2023", "12 Jam")
29 print(f"{Sepeda.get_merk(sepedalistrikA)} dengan kapasitas baterai {SepedaListrik.get_kapasitas_baterai(sepedalistrikA)}")
```

- Output

PS C:\KULIAH UMC\PBO LANJUT\INHERITANCE\praktikum2> py multilevel2.py

Sepeda Listrik Goda 140 New dengan kapasitas baterai 12 Jam

5. Hybrid Inheritance

A. Hybrid1

- Source Code

```
1  #Single Inheritance
2  class Team:
3      def __init__(self, nama_team):
4          self.nama_team = nama_team
5
6      def bermain(self):
7          print(f"Team {self.nama_team} sedang bermain")
8
9  #Single Inheritance
10 class TipeHero:
11     def __init__(self, tipe_hero):
12         self.tipe_hero = tipe_hero
13
14     def menggunakan(self):
15         print(f"Menggunakan {self.tipe_hero}")
16
17 #Multiple Inheritance
18 class Player(Team, TipeHero):
19     def __init__(self, nama_team, tipe_hero, jenis_permainan):
20         Team.__init__(self, nama_team)
21         TipeHero.__init__(self, tipe_hero)
22         self.jenis_permainan = jenis_permainan
23
24     def dipilih(self):
25         print(f"sedang dipilih untuk bermain di {self.jenis_permainan}")
26
27 teamA = Player("Barracuda", "Fighter", "Rank")
28 teamA.bermain()
29 teamA.menggunakan()
30 teamA.dipilih()
```

- Output

```
PS C:\KULIAH UMC\PBO LANJUT\INHERITANCE\praktikum2> py hybrid1.py
Team Barracuda sedang bermain
Menggunakan Fighter
sedang dipilih untuk bermain di Rank
```

B. Hybrid2

- Source Code

```
1  #Single Inheritance
2  class Kendaraan:
3      def __init__(self, brand):
4          self.brand = brand
5
6      def digunakan(self):
7          print(f"{self.brand} sedang digunakan")
8
9  #Single Inheritance
10 class Sepeda:
11     def __init__(self, merk):
12         self.merk = merk
13
14     def memilih(self):
15         print(f"Memilih sepeda {self.merk} ")
16
17 class SepedaListrik(Kendaraan, Sepeda):
18     def __init__(self, brand, merk, kapasitas_baterai):
19         Kendaraan.__init__(self, brand)
20         Sepeda.__init__(self, merk)
21         self.kapasitas_baterai = kapasitas_baterai
22
23     def memakai(self):
24         print(f"Memakai Sepeda Listrik {self.merk} dengan kapasitas baterai {self.kapasitas_baterai}")
25
26 sepedaA = SepedaListrik("Goda", "Sepeda Listrik Goda 140 New", "12 Jam")
27 sepedaA.digunakan()
28 sepedaA.memilih()
29 sepedaA.memakai()
```

- Output

```
PS C:\KULIAH UMC\PBO LANJUT\INHERITANCE\praktikum2> py hybrid2.py
Goda sedang digunakan
Memilih sepeda Sepeda Listrik Goda 140 New
Memakai Sepeda Listrik Sepeda Listrik Goda 140 New dengan kapasitas baterai 12 Jam
```