

LAPORAN INHERITANCE



Disusun Oleh :

5210411174_VERATINA FRIDAYANTI

**PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS & TEKNOLOGI
UNIVERSITAS TEKNOLOGI YOGYAKARTA**

1. Pengertian Inheritance (Pewarisan Class)

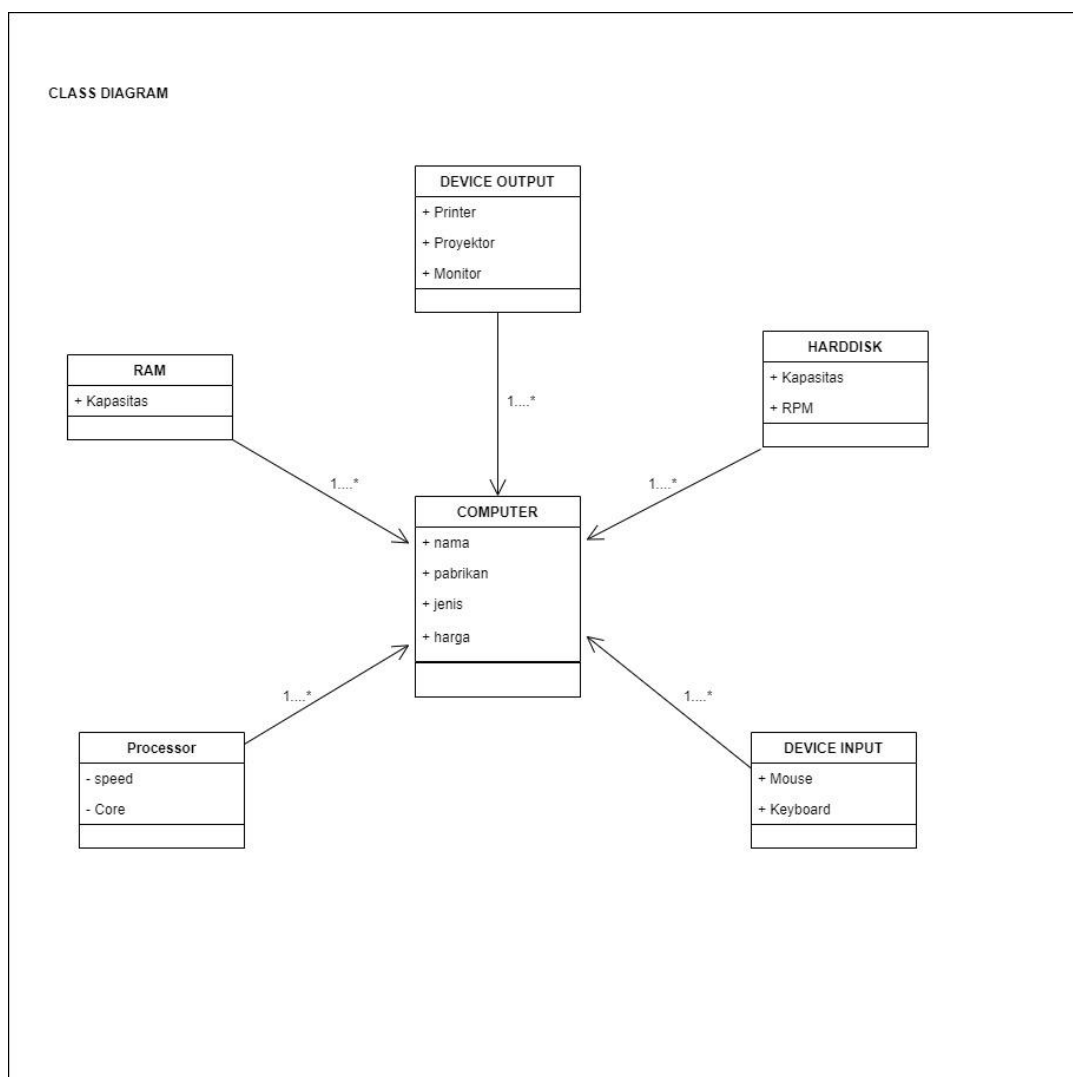
Inheritance adalah konsep OOP dimana sebuah class dapat menurunkan data member dan member function yang dimilikinya kepada class lain. Konsep inheritance dipakai untuk memanfaatkan fitur code reuse, yakni menghindari terjadinya duplikasi kode program.

Konsep inheritance membuat sebuah struktur atau hierarki class dalam kode program. Class yang akan diturunkan bisa disebut sebagai class induk (parent class), super class, atau base class. Sedangkan class yang menerima penurunan bisa disebut sebagai class anak (child class), sub class, derived class atau heir class.

2. Tujuan Inheritance (Pewarisan Class)

konsep pemrograman dimana sebuah class dapat 'menurunkan' property dan method yang dimilikinya kepada class lain. Konsep inheritance digunakan untuk memanfaatkan fitur 'code reuse' untuk menghindari duplikasi kode program.

3. Class Diagram



4. Program Hirarki

```
Get Started  hirarki.py U X  multiple.py U  multilevel.py U
hirarki.py > ...
1 class ComputerPart:
2     def __init__(self,pabrikan,nama,jenis,harga):
3         self.pabrikan = pabrikan
4         self.nama      = nama
5         self.jenis     = jenis
6         self.harga     = harga
7
8 class Processor(ComputerPart):
9     def __init__(self,pabrikan,nama,harga,jumlah_core,speed):
10        super().__init__(pabrikan,nama,"Processor",harga)
11        self.jumlah_core = jumlah_core
12        self.speed      = speed
13
14 class RandomAccessMemory(ComputerPart):
15     def __init__(self,pabrikan,nama,harga,kapasitas):
16        super().__init__(pabrikan,nama,"RAM",harga)
17        self.kapasitas = kapasitas
18
19 class HardDiskSATA(ComputerPart):
20     def __init__(self,pabrikan,nama,harga,kapasitas,rpm):
21        super().__init__(pabrikan,nama,"SATA",harga)
22        self.kapasitas = kapasitas
23        self.rpm       = rpm
24
25 p = Processor("Intel","Core i5 3250U",3500000,2,"2.6 GHz")
26 r = RandomAccessMemory("Corsaire","DDR4 SODIMM PC19200/2400 MHz",990000,"8 GB")
27 hdd = HardDiskSATA("Toshiba","HDD 2.5 inch",295000,"500 GB",7200)
28
29 part = [p,r,hdd]
30 for x in part:
31     print('\n{} {} produksi {} berharga {}'.format(x.jenis,x.nama,x.pabrikan,x.harga))
```

- **Output**

```
Processor Core i5 3250U produksi Intel berharga 3500000
RAM DDR4 SODIMM PC19200/2400 MHz produksi Corsaire berharga 990000
SATA HDD 2.5 inch produksi Toshiba berharga 295000
PS D:\KULIAH\Semester 2\Pemrograman Berorientasi Objek Praktikum\Tugas_PBO\Tugas\INHERITANCE> |
```

- **Penjelasan**

Suatu kelas dapat dibuat sebagai kelas yang benar-benar baru atau dapat dibentuk dari kelas yang sudah ada. Proses pembentukan kelas baru dari sebuah kelas yang sudah ada disebut sebagai pewarisan (inheritance). Kelas yang menjadi basis bagi pembentukan kelas lainnya disebut kelas super. Sedangkan kelas yang terbentuk dinamakan kelas sub.

Proses pewarisan inilah yang akan membentuk hirarki kelas. Kelas yang terletak pada puncak hirarki disebut sebagai kelas abstrak. Dimana sifat dari kelas abstrak hanya berisi deskripsi data dan method yang sangat umum. Sehingga tidak mungkin diimplementasikan secara terperinci. Oleh karena itu kelas abstrak ini tidak dapat langsung di instansiasi menjadi objek.

5. Program Multiple

```
Get Started hirarki.py U multiple.py X multilevel.py U
multiple.py > ...
9 class Storage():
10     def __init__(self, hdd) -> None:
11         self.hdd = hdd
12 class Monitor():
13     def __init__(self, mntr) -> None:
14         self.monitor = mntr
15 #child
16 class Computerpart(processor, Storage, RandomAccessMemory, Monitor) :
17     def __init__(self, processor, ram, hdd, mntr) -> None:
18         processor.__init__(self, processor)
19         Storage.__init__(self, hdd)
20         RandomAccessMemory.__init__(self, ram)
21         Monitor.__init__(self, mntr)
22 this_pc = Computerpart("Intel® Core™ i9-11900H", "DOMINATOR® PLATINUM RGB 32 GB (2 x 16 GB)"
23 ("DDR4 DRAM 3200MHz C16 Memory Kit", "HDD Toshiba 1 TB", "Monitor 24 inch Samsung S24F350FHE")
24 print("\n",
25 f'''
26 Saya mempunyai PC/Komputer dengan spesifikasi sebagai berikut :
27
28 Processor          : {this_pc.processor}
29 RandomAccessMemory : {this_pc.ram}
30 Storage             : {this_pc.hdd}
31 Monitor             : {this_pc.mntr}
32 ''')
```

- **Output**

```
Saya mempunyai PC/Komputer dengan Spesifikasi Sebagai Berikut :

Processor          : Intel® Core™ i9-11900H
Random Acces Memory : DOMINATOR® PLATINUM RGB 32 GB (2 x 16 GB) DDR4
DRAM 3200MHz C16 Memory Kit
Storage             : HDD Toshiba 1 TB
Monitor             : Monitor 24 inch Samsung S24F350FHE
```

- **Penjelasan**

Pada konsep multiple inheritance, sebuah kelas turunan akan memiliki semua atribut dan fitur dari semua kelas yang menjadi induknya.

6. Program MultiLevel

```
Get Started hirarki.py U multiple.py U multilevel.py X
multilevel.py > ...
1 class computerPart:
2     def __init__(self, nama, harga):
3         self.nama = nama
4         self.harga = harga
5 class storage(computerPart):
6     def __init__(self, nama, harga, jenis):
7         super().__init__(nama, harga)
8         self.jenis = jenis
9 class hdd(storage):
10     def __init__(self, nama, harga, kapasitas, rpm):
11         super().__init__(nama, harga, 'SSD')
12         self.kapasitas = kapasitas
13         self.rpm = rpm
14 pc = hdd('Seagate Harddisk Internal Barracuda', 750000, '1 TB', '7200 rpm')
15 print(f'Puji Tuhan Saya Mampu Memiliki Perangkat Computer dengan penyimpanan {pc.jenis} yang berkapasitas {pc.kapasitas} dan m
16
```

- **Output**

```
PS D:\KULIAH\Semester 2\Pemrograman Berorientasi Objek Praktik\GitHub\Tugas_PBO\Tugas\INHERITANCE> & "C:/Users/ASUS ExprtBook/AppData/Local/Microsoft/WindowsApps/python3.10.exe" "d:/KULIAH/Semester 2/Pemrograman Berorientasi Objek Praktik/GitHub/Tugas_PBO/Tugas/INHERITANCE/multilevel.py"
Puji Tuhan Saya Mampu Memiliki Perangkat Computer dengan penyimpanan SSD yang berkapasitas 1 TB dan memiliki RPM 7200 rpm
PS D:\KULIAH\Semester 2\Pemrograman Berorientasi Objek Praktik\GitHub\Tugas_PBO\Tugas\INHERITANCE>
```

- **Penjelasan**

penurunan class berjenjang dari satu class ke class lain. Misalkan kita memiliki class A yang diturunkan kepada class B. Kemudian class B ini juga diturunkan kepada class C, sehingga sudah terjadi sebuah multilevel inheritance.

7. Kesimpulan

Kesimpulannya, boleh dikatakan bahwa suatu subclass adalah tidak lain hanya memperluas (extend) parent class-nya.

8. Sumber

- <https://www.google.com/search?q=Prasetyo%2C+Didik+Dwi.+2007.+150+Rahasia+Pemrograman.+Bojonegoro+%3A+PT.+ElexMedia+Komputindo.3.+Tim+Asisten+Praktikum.+2014.&oq=Prasetyo%2C+Didik+Dwi.+2007.+150+Rahasia+Pemrograman.+Bojonegoro+%3A+PT.+ElexMedia+Komputindo.3.+Tim+Asisten+Praktikum.+2014.&aqs=chrome..69i57.606j0j7&sourceid=chrome&ie=UTF-8>
- https://www.academia.edu/37214408/LAPORAN_PRAKTIKUM_PBO_MODUL_3_INHERITANCE