

Object Oriented Programming Inheritance 2

**Name**

Virza Aulia Rachman

NIM

2241720078

Class

1i

Department

Information Technology

Study Program

D4 Informatics Engineering

Practicum 1

```
package Practicum1;

2 usages
public class Staff extends Karyawan{
    4 usages
    public int lembur, potongan;
    1 usage
    public Staff(){

    }
    no usages
    public Staff(String nama, String alamat, String jk, int umur, int gaji, int lembur, int potongan){
        super(nama, alamat, jk, umur, gaji);
        this.lembur = lembur;
        this.potongan = potongan;
    }
    1 usage
    public void tampilDatastaff(){
        super.tampilDataKaryawan();
        System.out.println("Lembur"      = "+lembur);
        System.out.println("Potongan"    = "+potongan);
        System.out.println("Total gaji"   = "+(gaji+lembur - potongan));
    }
}
```

```
package Practicum1;

2 usages
public class Manager extends Karyawan {
    3 usages
    public int tunjangan;
    1 usage
    public Manager(){

    }
    1 usage
    public void tampilDatamanager(){
        super.tampilDataKaryawan();
        System.out.println("Tunjangan"   = "+tunjangan);
        System.out.println("Total Gaji"   = "+(super.gaji+tunjangan));
    }
}
```

```
package Practicum1;

public class inheritance1 {
    public static void main(String[] args){
        Manager M = new Manager();
        M.nama="Vira";
        M.alamat="Jl. Woilah";
        M.umur=26;
        M.jk="Perempuan";
        M.tunjangan=1000000;
        M.tampilDatamanager();

        Staff S = new Staff();
        S.nama = "Virzaa";
        S.alamat = "Jl. punkstreet";
        S.umur = 23;
        S.jk = "Laki";
        S.gaji = 2000000;
        S.lembur = 500000;
        S.potongan = 250000;
        S.tampilDatastaff();
    }
}
```

```
package Practicum1;

2 usages 2 inheritors
public class Karyawan {
    4 usages
    public String nama, alamat, jk;
    3 usages
    public int umur, gaji;
    2 usages
    public Karyawan(){
    }
    1 usage
    public Karyawan(String nama, String alamat, String jk, int umur, int gaji){
        this.nama = nama;
        this.alamat = alamat;
        this.jk = jk;
        this.umur = umur;
        this.gaji = gaji;
    }
    2 usages
    public void tampilDataKaryawan(){
        System.out.println("Nama"      = " + nama);
        System.out.println("Alamat"    = " + alamat);
        System.out.println("Jenis Kelamin" = " + jk);
        System.out.println("Gaji"      = " + gaji);
    }
}
```

Nama	=Vira
Alamat	=Jl. Woilah
Jenis Kelamin	=Perempuan
Gaji	=0
Tunjangan	=1000000
Total Gaji	=1000000
Nama	=Virzaa
Alamat	=Jl. punkstreet
Jenis Kelamin	=Laki
Gaji	=2000000
Lembur	= 500000
Potongan	= 250000
Total gaji	= 2250000

Question

1. Sebutkan class mana yang termasuk super class dan sub class dari percobaan 1 diatas!
2. Kata kunci apakah yang digunakan untuk menurunkan suatu class ke class yang lain?
3. Perhatikan kode program pada class Manager, atribut apa saja yang dimiliki oleh class tersebut? Sebutkan atribut mana saja yang diwarisi dari class Karyawan!
4. Jelaskan kata kuncisuper pada potongan program dibawah ini yang terdapat pada class Manager!

```
Manager:  
    System.out.println("Total Gaji      =" + (super.gaji+tunjangan));
```
5. Program pada percobaan 1 diatas termasuk dalam jenis inheritance apa? Jelaskan alasannya!

Answer

1. Class "Karyawan" Super class, Class Manager dan Staff merupakan subclass
2. Extends
3. Atribut yang dimiliki hanya tunjangan,yang diwariskan dari karyawan adalah gaji nama jenis kelamin,umur,Alamat
4. Super digunakan untuk mempresentasikan class induk
5. Termasuk jenis Single inheritance.

Practicum 2

```
package Practicum2;

import ...

public class inheritance1 {
    public static void main(String[] args){
        StaffTetap ST = new StaffTetap( nama: "Teo", alamat: "Malang", jk: "Laki", umur: 20, gaji: 2000000, lembur: 250000, potongan: 200000, golongan: "2A", asuransi: 100000);
        ST.tampilStafftetap();

        StaffHarian SH = new StaffHarian( nama: "Windi", alamat: "Malang", jk: "Perempuan", umur: 27, gaji: 10000, lembur: 100000, potongan: 50000, jmljamkerja: 100);
        SH.tampilDataStaffharian();
    }
}
```

```
package Practicum2;

2 usages
public class StaffTetap extends Staff{
    2 usages
    public String golongan;
    3 usages
    public int asuransi;

    no usages
    public StaffTetap(){

    }
    1 usage
    public StaffTetap(String nama,String alamat,String jk,int umur,int gaji,
        int lembur,int potongan,String golongan,int asuransi){
        super(nama,alamat,jk,gaji,umur,potongan,lembur);
        this.golongan = golongan;
        this.asuransi = asuransi;
    }
    1 usage
    public void tampilStafftetap(){
        System.out.println("+++++++=Data Staff Tetap++++++");
        super.tampilDatastaff();
        System.out.println("Golongan      ="+golongan);
        System.out.println("Jumlah Asuransi ="+asuransi);
        System.out.println("Gaji Bersih    ="+(gaji+lembur-potongan-asuransi));
    }
}
```

```
+++++++=Data Staff Tetap++++++
Nama           =Teo
Alamat         =Malang
Jenis Kelamin  =Laki
Gaji           =20
Lembur         = 200000
Potongan       = 250000
Total gaji     = -49980
Golongan       =2A
Jumlah Asuransi =100000
Gaji Bersih    =-149980
+++++++=Data Staff Harian++++++
Nama           =Windi
Alamat         =Malang
Jenis Kelamin  =Perempuan
Gaji           =10000
Lembur         = 100000
Potongan       = 50000
Total gaji     = 60000
Jumlah Jam kerja = 100
Gaji Bersih    = 1050000
```

```
package Practicum2;

2 usages
public class StaffHarian extends Staff{
    3 usages
    public int jmljamkerja;
    no usages
    public StaffHarian(){

    }
    1 usage
    public StaffHarian(String nama,String alamat,String jk,int umur,
        int gaji,int lembur,int potongan,int jmljamkerja){
        super(nama, alamat, jk, umur, gaji, lembur, potongan);
        this.jmljamkerja = jmljamkerja;
    }
    1 usage
    public void tampilDataStaffharian(){
        System.out.println("+++++++=Data Staff Harian++++++");
        super.tampilDatastaff();
        System.out.println("Jumlah Jam kerja = "+jmljamkerja);
        System.out.println("Gaji Bersih     = "+(gaji*jmljamkerja+lembur-potongan));
    }
}
```

Question

1. Berdasarkan class diatas manakah yang termasuk single inheritance dan mana yang termasuk multilevel inheritance?
2. Perhatikan kode program class StaffTetap dan StaffHarian, atribut apa saja yang dimiliki oleh class tersebut? Sebutkan atribut mana saja yang diwarisi dari class Staff!
3. Apakah fungsi potongan program berikut pada class StaffHarian
`super(nama,alamat,jk,umur,gaji,lembur,potongan);`
4. Apakah fungsi potongan program berikut pada class StaffHarian
`super.tampilDataStaff();`
5. Perhatikan kode program dibawah ini yang terdapat pada class StaffTetap
`System.out.println("Gaji Bersih =" + (gaji+lembur-potongan-asuransi));`
Terlihat dipotongan program diatas atribut gaji, lembur dan potongan dapat diakses langsung. Kenapa hal ini bisa terjadi dan bagaimana class StaffTetap memiliki atribut gaji, lembur, dan potongan padahal dalam class tersebut tidak dideklarasikan atribut gaji, lembur, dan potongan

Answer

1. Karyawan – Staff (Single inheritance), Karyawan – Staff – Stafftetap (Multilevel inheritance)
2. Stafftetap : golongan, asuransi, Staffharian : jmljamkerja, Yang diwarisi dari Staff : lembur, potongan
3. Untuk mengambil nilai dari induk
4. Untuk memanggil fungsi dari induk
5. Karena Class Stafftetap mewarisi atribut dari Karyawan melalui Class Staff.

Assingment

- Komputer

```
package Assingment;

2 usages 4 inheritors
public class Komputer {
    3 usages
    public String merk, jnsProsesor;
    3 usages
    public int kecProsesor, sizeMemory;
    2 usages
    public Komputer(){}
    2 usages
    public Komputer(String merk, String jnsProsesor, int kecProsesor, int sizeMemory){
        this.jnsProsesor = jnsProsesor;
        this.merk = merk;
        this.kecProsesor = kecProsesor;
        this.sizeMemory = sizeMemory;
    }
}
```

- Laptop

```
package Assingment;
```

2 usages 2 inheritors

```
public class Laptop extends Komputer {
```

2 usages

```
    public String jnsBatrei;
```

2 usages

```
    public Laptop(){} 
```

2 usages

```
    public Laptop(String merk,String jnsBatrei,String jnsProsesor,  
                    int kecProsesor,int sizeMemory){  
        super(merk, jnsProsesor, kecProsesor, sizeMemory);  
        this.jnsBatrei=jnsBatrei;  
    }
```

2 usages

```
    public void tampilkanLaptop(){
```

```
        System.out.println("merk          = " +merk);
```

```
        System.out.println("kecProsesor = " +kecProsesor);
```

```
        System.out.println("sizeMemory = " +sizeMemory);
```

```
        System.out.println("jnsProsesor = " +jnsProsesor);
```

```
        System.out.println("jnsBatrei   = " +jnsBatrei);
```

```
    }
```

```
}
```

- PC

```
package Assingment;

2 usages
public class PC extends Komputer{
    2 usages
    public int ukuranMonitor;
    no usages
    public PC (){};
    1 usage
    public PC(String merk,int kecProsesor,int sizeMemory,
        String jnsProsesor,int ukuranMonitor){
        super(merk, jnsProsesor, kecProsesor, sizeMemory);
        this.ukuranMonitor = ukuranMonitor;
    }
    1 usage
    public void tampilPC() {
        System.out.println("merk = " + merk);
        System.out.println("kecProsesor = " + kecProsesor);
        System.out.println("sizeMemory = " + sizeMemory);
        System.out.println("jnsProsesor = " + jnsProsesor);
        System.out.println("Monitor = " + ukuranMonitor + "inch");
    }
}
```


- Mac

```
package Assingment;

2 usages
public class Mac extends Laptop{
    2 usages
    public String security;
    1 usage
    public Mac(){
    2 usages
    public Mac(String merk,String jnsBatrei,String jnsProsesor,
        int kecProsesor,int sizeMemory,String security){
        super(merk, jnsBatrei, jnsProsesor, kecProsesor, sizeMemory);
        this.security = security;
    }
    1 usage
    public void tampilMac() {
        super.tampilLaptop();
        System.out.println("security    = " +security);
    }
}
```

- Windows

```
package Assingment;

2 usages
public class Windows extends Laptop{
    2 usages
    public String fitur;
    no usages
    public Windows(){
    1 usage
    public Windows(String merk,String jnsBatrei,String jnsProsesor,
        int kecProsesor,int sizeMemory,String fitur){
        super(merk, jnsBatrei, jnsProsesor, kecProsesor, sizeMemory);
        this.fitur = fitur;
    }
    1 usage
    public void tampilWindows() {
        super.tampilLaptop();
        System.out.println("fitur = " +fitur);
    }
}
```

- Result

```
merk      = Apple
kecProsesor = 100000
sizeMemory = 240000
jnsProsesor = m1
jnsBatrei  = Abc
security   = Avast
merk      = Asus
kecProsesor = 100000
sizeMemory = 240000
jnsProsesor = Ryzen
jnsBatrei  = Abc
fitur     = Leptop geming
merk      = Legion
kecProsesor = 1000000
sizeMemory = 2500000
jnsProsesor = Intel omegod
Monitor   = 100inch
```