

Object Oriented Programming



Malik Abdul Azis

2241720240

2I Class

13

STUDY PROGRAM D-IV INFORMATIC ENGINEER

INFORMATION TECHNOLOGY DEPARTMENT

MALANG STATE POLYTECHNIC

Soekarno Hatta Street No.9, Jatimulyo, Lowokwaru District, Malang City, East Java

65141

Percobaan 1

1. Sebutkan class mana yang termasuk super class dan sub class dari percobaan 1 diatas!
Class Karyawan merupakan super class dari Class Manager dan Class Staff yang merupakan sub class
2. Kata kunci apakah yang digunakan untuk menurunkan suatu class ke class yang lain?
Extends adalah kata kunci yang ditulis setelah deklarasi nama class, kata kunci inilah yang digunakan untuk menurunkan suatu class ke class yang memiliki kata kunci ini.
3. Perhatikan kode program pada class Manager, atribut apa saja yang dimiliki oleh class tersebut?
Sebutkan atribut mana saja yang diwarisi dari class Karyawan!
Pada class Manager() memiliki atribut tunjangan, serta beberapa atribut yang diwarisi dari class Karyawan() yaitu nama, alamat, umur, jk, dan gaji.
4. Jelaskan kata kunci super pada potongan program dibawah ini yang terdapat pada class Manager!

```
System.out.println("Total Gaji      =" + (super.gaji + tunjangan));
```


Kata kunci super digunakan untuk mengakses attribute atau method yang ada pada class utama (super class)
5. Program pada percobaan 1 diatas termasuk dalam jenis inheritance apa? Jelaskan alasannya!
Program pada percobaan 1 termasuk pada jenis hierarchical inheritance karena super class diwariskan ke beberapa class / sub class.

Percobaan 2

1. Berdasarkan class diatas manakah yang termasuk single inheritance dan mana yang termasuk multilevel inheritance?
Class Karyawan dengan class Manager merupakan single inheritance, sedangkan class Karyawan-Staff-StaffTetap atau Karyawan-Staff-StaffHarian merupakan 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!
Pada staffTetap, class tersebut memiliki attribute yaitu golongan dan asuransi, sedangkan pada class StaffHarian, memiliki attribute jmlJamKerja. Attribute dari kedua class yang diwarisi dari class Staff adalah attribute nama, Alamat, umur, jk, gaji, lembur, dan potongan.
3. Apakah fungsi potongan program berikut pada class StaffHarian

```
super(nama, alamat, jk, umur, gaji, lembur, potongan);
```


Potongan kode tersebut berfungsi untuk melakukan assign value kepada constructor parent class yaitu Staff
4. Apakah fungsi potongan program berikut pada class StaffHarian

```
super.tampilDataStaff();
```


Potongan kode tersebut digunakan untuk mengakses method tampilDataStaff() yang dimiliki oleh parent class yaitu class Staff()
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?

Sebelumnya, attribute tersebut berasal dari class Staff serta class Karyawan. Attribute tersebut dapat diakses secara langsung karena attribute tersebut bersifat public serta karena class Staff mewarisi attribute gaji dari class Karyawan maka class StaffTetap dapat mengakses attribute gaji.

Tugas

```
package oop_java.week7;

public class Komputer {
    public String merk, jnsProsesor;
    public int kecProsesor, sizeMemory;

    public Komputer() {
    }

    public Komputer(String merk, String jnsProsesor, int kecProsesor, int sizeMemory) {
        this.merk = merk;
        this.jnsProsesor = jnsProsesor;
        this.kecProsesor = kecProsesor;
        this.sizeMemory = sizeMemory;
    }

    public void tampilData() {
        System.out.println("Merk\t\t: " + merk);
        System.out.println("Jenis Prosesor\t: " + jnsProsesor);
        System.out.println("Kecepatan Prosesor\t: " + kecProsesor);
        System.out.println("Size Memory\t: " + sizeMemory);
    }
}
```

```
package oop_java.week7;

public class Pc extends Komputer{
    public int ukuranMonitor;

    public Pc() {
    }

    public Pc(String merk, String jnsProsesor, int kecProsesor, int sizeMemory, int ukuranMonitor) {
        super(merk, jnsProsesor, kecProsesor, sizeMemory);
        this.ukuranMonitor = ukuranMonitor;
    }

    public void tampilPc() {
        super.tampilData();
        System.out.println("Ukuran Monitor\t: " + ukuranMonitor);
    }
}
```

```
package oop_java.week7;

public class Leptop extends Komputer{
    public String jnsBatrei;

    public Leptop() {
    }

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

    public void tampilLeptop() {
        super.tampilData();
        System.out.println("Jenis Baterai\t: " + jnsBatrei);
    }
}
```

```

package oop_java.week7;

public class Mac extends Leptop{
    public String security;

    public Mac() {
    }

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

    public void tampilMac() {
        super.tampilLeptop();
        System.out.println("Security\t: " + security);
    }
}

```

```

package oop_java.week7;

public class Windows extends Leptop{
    String fitur;

    public Windows() {
    }

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

    public void tampilWindows() {
        super.tampilLeptop();
        System.out.println("Fitur\t: " + fitur);
    }
}

```

```

package oop_java.week7;

public class tugasMain {
    Run | Debug
    public static void main(String[] args) {
        Mac m1 = new Mac(merk:"Macbook Pro 2020", jnsProsesor:"Intel i7", kecProsesor:3, sizeMemory:16, jnsBatrei:"Li-Ion", security:"Apple T2 Chip");
        m1.tampilMac();
        System.out.println();
        Windows w1 = new Windows(merk:"Asus Zenbook", jnsProsesor:"Intel i7", kecProsesor:3, sizeMemory:16, jnsBatrei:"Li-Ion", fitur:"Windows Hello");
        w1.tampilWindows();
        System.out.println();
        Pc gaming = new Pc(merk:"Asus ROG", jnsProsesor:"Intel i7", kecProsesor:3, sizeMemory:16, ukuranMonitor:32);
        gaming.tampilPc();
    }
}

```

```

Merk      : Macbook Pro 2020
Jenis Prosesor : Intel i7
Kecepatan Prosesor : 3
Size Memory : 16
Jenis Baterai : Li-Ion
Security   : Apple T2 Chip

```

```

Merk      : Asus Zenbook
Jenis Prosesor : Intel i7
Kecepatan Prosesor : 3
Size Memory : 16
Jenis Baterai : Li-Ion
Fitur     : Windows Hello
Merk      : Asus ROG
Jenis Prosesor : Intel i7
Kecepatan Prosesor : 3
Size Memory : 16
Ukuran Monitor : 32

```