

Object Oriented Programming
Jobsheet 6
Inheritance 2



BY :

D4 INFORMATICS ENGINEERING (2I)
Shofa Yasmin Fauziah (22/2241720108)

State Polytechnic of Malang
Soekarno Hatta street No.9, Malang, East Java 65141
2023/2024

PERCOBAAN 1

```
1 package Percobaan1;
2 public class Karyawan {
3     public String nama, alamat, jk;
4     public int umur, gaji;
5
6     public Karyawan() {
7     }
8
9     public Karyawan(String nama, String alamat, String jk, int umur, int gaji) {
10         this.nama = nama;
11         this.alamat = alamat;
12         this.jk = jk;
13         this.umur = umur;
14         this.gaji = gaji;
15     }
16
17     public void tampilDataKaryawan() {
18         System.out.println("Nama          =" + nama);
19         System.out.println("Alamat          =" + alamat);
20         System.out.println("Jeanis Kelamin =" + jk);
21         System.out.println("Umur           =" + umur);
22         System.out.println("Gaji           =" + gaji);
23     }
24
25 }
```

```
1 package Percobaan1;
2 public class Manager extends Karyawan{
3     public int tunjangan;
4
5     public Manager() {
6     }
7
8     public void tampilDataManager() {
9         super.tampilDataKaryawan();
10        System.out.println("Tunjangan          =" + tunjangan);
11        System.out.println("Total Gaji         =" + (super.gaji + tunjangan));
12    }
13 }
14
```

```

1 package Percobaan1;
2 public class Staff extends Karyawan{
3     public int lembur, potongan;
4
5     public Staff() {
6     }
7
8     public Staff(int lembur, int potongan, String nama, String alamat, String jk, int umur, int gaji) {
9         super(nama, alamat, jk, umur, gaji);
10        this.lembur = lembur;
11        this.potongan = potongan;
12    }
13
14    public void tampilDataStaff(){
15        super.tampilDataKaryawan();
16        System.out.println("Lembur      ="+lembur);
17        System.out.println("Potongan    ="+potongan);
18        System.out.println("Total Gaji   ="+(gaji+lembur-potongan));
19    }
20 }

```

```

Nama      =Vivin
Alamat    =Jl. Vinolia
Jeanis Kelamin =Perempuan
Umur      =25
Gaji      =3000000
Tunjangan =1000000
Total Gaji =4000000
Nama      =Lestari
Alamat    =Malang
Jeanis Kelamin =Perempuan
Umur      =25
Gaji      =2000000
Lembur    =500000
Potongan  =250000
Total Gaji =2250000

```

PERTANYAAN

1. Sebutkan class mana yang termasuk super class dan sub class dari percobaan 1 diatas!
Class Karyawan termasuk superclass, class Staff dan class Manager termasuk subclass.
2. Kata kunci apakah yang digunakan untuk menurunkan suatu class ke class yang lain?
Extends
3. Perhatikan kode program pada class Manager, atribut apa saja yang dimiliki oleh class tersebut? Sebutkan atribut mana saja yang diwarisi dari class Karyawan!
Atributte yang dimiliki class Manager: tunjangan dengan datatype int.
Atributte yang diwarisi: 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 merujuk atributte yang berada di parents class.

5. Program pada percobaan 1 diatas termasuk dalam jenis inheritance apa? Jelaskan alasannya!

Hierarchical Inheritance, karena class Staff dan class Manager memiliki induk class yang sama, yaitu class Karyawan.

PERCOBAAN 2

```
1 package Percobaan1;
2 public class StaffHarian extends Staff{
3     public int jmlJamKerja;
4
5     public StaffHarian() {
6     }
7
8     public StaffHarian(String nama, String alamat, String jk, int umur,
9         int gaji, int jmlJamKerja, int lembur, int potongan) {
10         super(nama, alamat, jk, umur, gaji, lembur, potongan);
11         this.jmlJamKerja = jmlJamKerja;
12     }
13     public void tampilStaffHarian(){
14         System.out.println("-----Data Staff Harian-----");
15         super.tampilDataStaff();
16         System.out.println("Jumlah Jam Kerja="+jmlJamKerja);
17         System.out.println("Gaji Bersih      "+(gaji*jmlJamKerja+lembur-potongan));
18     }
19 }
20
21 package Percobaan1;
22 public class StaffTetap extends Staff {
23     public String golongan;
24     public int asuransi;
25
26     public StaffTetap() {
27     }
28
29     public StaffTetap(String nama, String alamat, String jk, int umur,
30         int gaji,int lembur, int potongan, String golongan, int asuransi) {
31         super(nama, alamat, jk, umur, gaji, lembur, potongan);
32         this.golongan = golongan;
33         this.asuransi = asuransi;
34     }
35
36     public void tampilStaffTetap(){
37         System.out.println("-----Data Staff Tetap-----");
38         super.tampilDataStaff();
39         System.out.println("Golongan      "+golongan);
40         System.out.println("Jumlah asuransi "+asuransi);
41         System.out.println("Gaji Bersih      "+(gaji+lembur-potongan-asuransi));
42     }
43 }
```

```

1 package Percobaan1;
2 public class Inheritance1 {
3     public static void main(String[] args) {
4         StaffTetap ST = new StaffTetap(nama: "Budi", alamat: "Malang", jk: "Laki-Laki", umur: 20,
5             gaji: 2000000, lembur: 200000, potongan: 2500000, golongan: "2A", asuransi: 100000);
6         ST.tampilStaffTetap();
7
8         StaffHarian SH = new StaffHarian(nama: "Indah", alamat: "Malang", jk: "Perempuan", umur: 27,
9             gaji: 10000, jmlJamKerja: 100, lembur: 100000, potongan: 50000);
10        SH.tampilStaffHarian();
11    }
12 }

```

-----Data Staff Tetap-----

```

Nama          =Budi
Alamat        =Malang
Jenis Kelamin =Laki-Laki
Umur          =20
Gaji          =2000000
Lembur        =200000
Potongan      =2500000
Total Gaji    =-300000
Golongan      =2A
Jumlah asuransi =100000
Gaji Bersih   =-400000

```

-----Data Staff Harian-----

```

Nama          =Indah
Alamat        =Malang
Jenis Kelamin =Perempuan
Umur          =27
Gaji          =10000
Lembur        =100000
Potongan      =50000
Total Gaji    =60000
Jumlah Jam Kerja=100
Gaji Bersih   =1050000

```

PERTANYAAN

1. Berdasarkan class diatas manakah yang termasuk single inheritance dan mana yang termasuk multilevel inheritance?

Single heritage: class Karyawan superclass dari class Manager

Multilevel heritage:

- class StaffHarian merupakan subclass class Staff, sehingga class Staff menjadi superclass dari class StaffHarian. Lalu class Staff merupakan subclass dari class Karyawan, sehingga class Karyawan merupakan super class dari class Staff.

- class StaffTetap merupakan subclass class Staff, sehingga class Staff menjadi superclass dari class StaffTetap. Lalu class Staff merupakan subclass dari class Karyawan, sehingga class Karyawan merupakan super class dari class Staff.

class StaffHarian dan class StaffTetap sama-sama merupakan subclass dari class Staff

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!

Attribute yang dimiliki: String golongan, int asuransi

Attribute yang diwarisi: String nama, String alamat, String jk, int umur, int gaji, int lembur, int potongan

3. Apakah fungsi potongan program berikut pada class StaffHarian

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

Untuk memanggil konstruktor class Staff yang merupakan superclass untuk menginisialisasi konstruktor class StaffHarian.

4. Apakah fungsi potongan program berikut pada class StaffHarian

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

Untuk memanggil method tampilDataStaff() yang berada di class Staff lalu di cetak di dalam method tampilStaffHarian().

5. Perhatikan kode program dibawah ini yang terdapat pada class StaffTetap

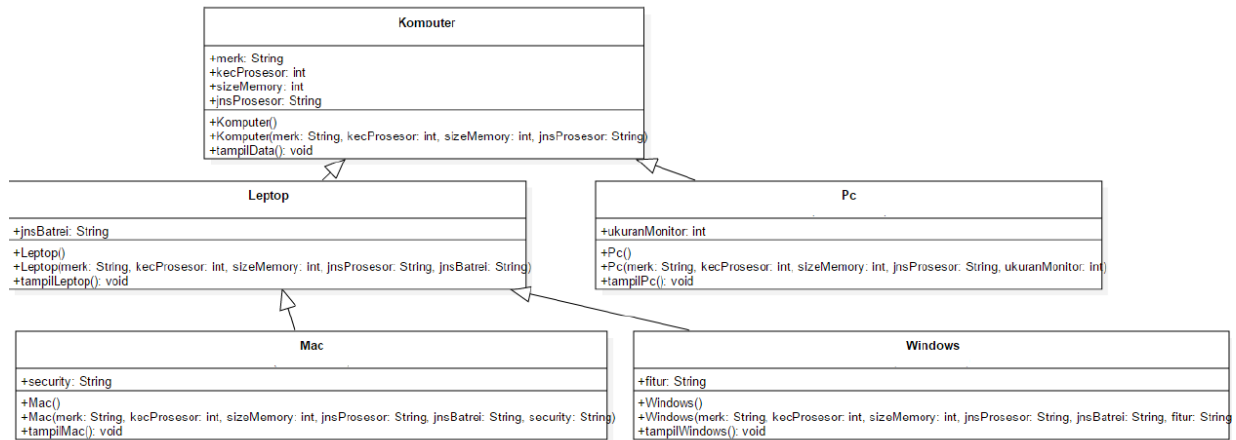
```
System.out.println("Gaji Bersih      =" + (gaji + lembur - potongan - asuransi));
```

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?

Karena di dalam class terdapat kode program StaffTetap extends Staff yang menjadikan class StaffTetap merupakan turunan atau sub class dari class Staff, sehingga kita bisa memanggil method yang beradda di class Staff ke dalam class StaffTetap.

TUGAS

Buatlah sebuah program dengan konsep pewarisan seperti pada class diagram berikut ini. Kemudian buatlah instansiasi objek untuk menampilkan data pada class Mac, Windows dan Pc!.




```

1 package Tugas;
2 public class Komputer {
3     public String merk;
4     public double kecProsesor;
5     public int sizeMemory;
6     public String jnsProsesor;
7
8     public Komputer() {
9     }
10
11     public Komputer(String merk, double kecProsesor, int
sizeMemory, String jnsProsesor) {
12         this.merk = merk;
13         this.kecProsesor = kecProsesor;
14         this.sizeMemory = sizeMemory;
15         this.jnsProsesor = jnsProsesor;
16     }
17
18     public void tampilData(){
19         System.out.println("Merk          : "+merk);
20         System.out.println("Kecepatan Prosesor   :
"+kecProsesor+ "GHz");
21         System.out.println("Size Memory        :
"+sizeMemory+ "GB RAM");
22         System.out.println("Jenis Prosesor     :
"+jnsProsesor);
23     }
24 }
  
```




```
1 package Tugas;
2 public class Laptop extends Komputer{
3     public String jnsBaterai;
4
5     public Laptop() {
6     }
7
8     public Laptop( String merk, double kecProsesor, int
sizeMemory, String jnsProsesor, String jnsBaterai) {
9         super(merk, kecProsesor, sizeMemory, jnsProsesor);
10        this.jnsBaterai = jnsBaterai;
11    }
12
13    public void tampilLaptop(){
14        super.tampilData();
15        System.out.println("Jenis Baterai      :
"+jnsBaterai);
16    }
17
18 }
```

```
1 package Tugas;
2 public class Mac extends Laptop{
3     public String security;
4
5     public Mac() {
6     }
7
8     public Mac(String merk, double kecProsesor, int
sizeMemory, String jnsProsesor, String jnsBaterai, String
security) {
9         super(merk, kecProsesor, sizeMemory, jnsProsesor,
jnsBaterai);
10        this.security = security;
11    }
12
13    public void tampilMac(){
14        super.tampilLaptop();
15        System.out.println("Security      :
"+security);
16    }
17 }
```

```
1 package Tugas;
2 public class Windows extends Laptop{
3     public String fitur;
4
5     public Windows() {
6     }
7
8     public Windows(String merk, double kecProsesor, int
sizeMemory, String jnsProsesor, String jnsBaterai, String
fitur) {
9         super(merk, kecProsesor, sizeMemory, jnsProsesor,
jnsBaterai);
10        this.fitur = fitur;
11    }
12
13    public void tampilWindows(){
14        super.tampilLaptop();
15        System.out.println("Fitur          : "+fitur);
16    }
17 }
```



```
1 package Tugas;
2 public class Pc extends Komputer{
3     public int ukuranMonitor;
4
5     public Pc() {
6     }
7
8     public Pc(String merk, double kecProsesor, int
sizeMemory, String jnsProsesor, int ukuranMonitor) {
9         super(merk, kecProsesor, sizeMemory, jnsProsesor);
10        this.ukuranMonitor = ukuranMonitor;
11    }
12
13    public void tampilPc(){
14        super.tampilData();
15        System.out.println("Ukuran Monitor      :
"+ukuranMonitor);
16    }
17 }
```



```
1 package Tugas;
2 public class TugasMain {
3     public static void main(String[] args) {
4         Komputer komputer = new Komputer("HP Envy Desktop",
5         2.9,
6             16, "Intel Core i7-10700");
7
8         Laptop laptop1 = new Laptop("Dell XPS 13", 2.8,
9             16, "Intel Core i7-1165G7", "lithium-ion");
10
11         Mac mac1 = new Mac("MacBook Pro", 2.4, 16,
12             "Intel Core i9-9980HK ", "lithium-ion", "T2
13             Security Chip");
14
15         Windows windows1 = new Windows("HP Spectre x360",
16             2.8,
17             16, "Core i7-1165G7", "lithium-ion", "FHD
18             touchscreen");
19
20         Pc pc1 = new Pc("HP Pavilion Desktop", 3.6, 8,
21             "AMD Ryzen 5 3600", 23);
22
23         komputer.tampilData();
24         System.out.println("- - - - -");
25
26         laptop1.tampilLaptop();
27         System.out.println("- - - - -");
28
29         mac1.tampilMac();
30         System.out.println("- - - - -");
31
32         windows1.tampilWindows();
33         System.out.println("- - - - -");
34
35         pc1.tampilPc();
36         System.out.println("- - - - -");
37
38     }
39 }
```