

**JOBSHEET 7**  
**OBJECT ORIENTED PROGRAMMING**



Arranged By :

Lenka Melinda Florianka

2241720074

Class 2I

**INFORMATION TECHNOLOGY**  
**D-IV INFORMATICS ENGINEERING**  
**MALANG STATE POLYTECHNIC**  
**2023**

## Practicum 1

Code :

```
1 package jobsheet7;
2
3 public class Inheritance1 {
4     public static void main(String[] args) {
5         Manager M = new Manager();
6         M.nama = "Vivin";
7         M.alamat = "Jl. Vinolia";
8         M.umur = 25;
9         M.jk = "Perempuan";
10        M.gaji = 3000000;
11        M.tunjangan = 1000000;
12        M.tampilDataManager();
13
14        Staff S = new Staff();
15        S.nama = "Lestari";
16        S.alamat = "Malang";
17        S.umur = 25;
18        S.jk = "Perempuan";
19        S.gaji = 2000000;
20        S.lembur = 500000;
21        S.potongan = 250000;
22        S.tampilDataStaff();
23    }
24 }
25
```

```
1 package jobsheet7;
2 public class Staff extends Karyawan {
3     public int lembur, potongan;
4
5     public Staff() {
6
7     }
8     public Staff(String nama, String alamat, String jk, int umur, int gaji, int lembur, int potongan) {
9         super(nama, alamat, jk, umur, gaji);
10        this.lembur = lembur;
11        this.potongan = potongan;
12    }
13    public void tampilDataStaff() {
14        super.tampilDataKaryawan();
15        System.out.println("Lembur      = " + lembur);
16        System.out.println("Potongan    = " + potongan);
17        System.out.println("Total Gaji   = " + (gaji + lembur - potongan));
18    }
19 }
20
```

```

1 package jobsheet7;
2 public class Manager extends Karyawan{
3     public int tunjangan;
4
5     public Manager () {
6
7     }
8
9     public void tampilDataManager() {
10         super.tampilDataKaryawan();
11         System.out.println("Tunjangan      = " + tunjangan);
12         System.out.println("Total Gaji      = " + (super.gaji + tunjangan));
13     }
14 }
15

```

```

1 package jobsheet7;
2 public class Karyawan {
3     public String nama, alamat, jk;
4     public int umur, gaji;
5
6     public Karyawan () {
7
8     }
9
10    public Karyawan (String nama, String alamat, String jk, int umur, int gaji) {
11        this.nama = nama;
12        this.alamat = alamat;
13        this.jk = jk;
14        this.umur = umur;
15        this.gaji = gaji;
16    }
17
18    public void tampilDataKaryawan () {
19        System.out.println("Nama          = " + nama);
20        System.out.println("Alamat        = " + alamat);
21        System.out.println("Jenis Kelamin = " + jk);
22        System.out.println("Umur          = " + umur);
23        System.out.println("Gaji          = " + gaji);
24    }
25 }
26

```

Output :

```

Nama          = Vivin
Alamat        = Jl. Vinolia
Jenis Kelamin = Perempuan
Umur          = 25
Gaji          = 3000000
Tunjangan     = 1000000
Total Gaji    = 4000000
Nama          = Lestari
Alamat        = Malang
Jenis Kelamin = Perempuan
Umur          = 25
Gaji          = 2000000
Lembur        = 500000
Potongan      = 250000
Total Gaji    = 2250000
PS C:\Dev\00P>

```

## Questions

1. Sebutkan class mana yang termasuk super class dan sub class dari percobaan 1 diatas!

Super class dalam percobaan 1 di atas adalah class Karyawan, sedangkan sub class-nya adalah Manager dan Staff.

2. Kata kunci apakah yang digunakan untuk menurunkan suatu class ke class yang lain?

Kata kunci yang digunakan untuk menurunkan suatu class ke class yang lain adalah 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!

Atribut yang dimiliki oleh class Manager adalah:

- tunjangan

Atribut yang diwarisi dari class Karyawan ke class Manager adalah:

- nama
- alamat
- jk
- umur
- 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 pada potongan program di bawah ini digunakan untuk merujuk ke superclass, yaitu class Karyawan. Dalam konteks ini, super.gaji mengakses atribut gaji yang diwarisi dari superclass Karyawan, dan tunjangannya adalah atribut dari class Manager.

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

Program pada percobaan 1 di atas termasuk dalam jenis inheritance "Single Inheritance". Ini karena setiap subclass (dalam hal ini, Manager dan Staff) hanya memiliki satu superclass, yaitu Karyawan. Dalam single inheritance, suatu class hanya dapat mewarisi atribut dan metode dari satu superclass.

## Practicum 2

### Code :

```
1 package jobsheet7;
2
3 public class Inheritance1 {
4     public static void main(String[] args) {
5         StaffTetap ST = new StaffTetap("Budi", "Malang", "Lakilaki", 20, 2000000, 250000, 200000, "2S", 100000);
6         ST.tampilStaffTetap();
7
8         StaffHarian SH = new StaffHarian("Indah", "Malang", "Perempuan", 27, 10000, 100000, 50000, 100);
9         SH.tampilStaffHarian();
10    }
11 }
```

```
1 package jobsheet7;
2
3 public class StaffHarian extends Staff{
4     public int jmlJamKerja;
5
6     public StaffHarian() {
7
8     }
9     public StaffHarian(String nama, String alamat, String jk, int umur, int gaji, int lembur, int potongan, int jmlJamKerja) {
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 }
```

```
1 public class StaffTetap extends Staff{
2     public String golongan;
3     public int asuransi;
4
5     public StaffTetap() {
6
7     }
8     public StaffTetap(String nama, String alamat, String jk, int umur, int gaji, int lembur, int potongan, String golongan, int asuransi) {
9         super(nama, alamat, jk, umur, gaji, potongan, lembur);
10         this.golongan = golongan;
11         this.asuransi = asuransi;
12     }
13     public void tampilStaffTetap() {
14         System.out.println("=====Data Staff Tetap=====");
15         super.tampilDataStaff();
16         System.out.println("Golongan      = " + golongan);
17         System.out.println("Jumlah Asuransi = " + asuransi);
18         System.out.println("Gaji Bersih    = " + (gaji + lembur - potongan - asuransi));
19     }
20 }
```

## Output :

```
=====Data Staff Tetap=====
Nama      = Budi
Alamat    = Malang
Jenis Kelamin = Laki-laki
Umur      = 20
Gaji      = 2000000
Lembur    = 200000
Potongan  = 250000
Total Gaji = 1950000
Golongan  = 25
Jumlah Asuransi = 100000
Gaji Bersih = 1850000
=====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 = 60100
PS C:\Dev\OOP>
```

## Questions

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

- Single Inheritance: Tidak ada single inheritance dalam kode program yang diberikan. Single inheritance terjadi saat sebuah class hanya mewarisi dari satu class lainnya. Dalam kode ini, tidak ada class yang mewarisi dari class lain.
- Multilevel Inheritance: Tidak ada multilevel inheritance dalam kode program yang diberikan. Multilevel inheritance terjadi ketika sebuah class mewarisi dari class lain, dan kemudian class lain mewarisi dari class yang sebelumnya diwarisi. Dalam kode ini, hierarki pewarisan hanya berlangsung satu level, yaitu StaffTetap dan StaffHarian mewarisi dari class Staff, tetapi tidak ada class yang mewarisi dari StaffTetap atau StaffHarian.

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!

Atribut dalam class StaffTetap dan StaffHarian serta atribut yang diwarisi dari class Staff:

StaffTetap:

- Atribut yang dimiliki: golongan, asuransi
- Atribut yang diwarisi dari class Staff: nama, alamat, jk, umur, gaji, lembur, potongan

StaffHarian:

- Atribut yang dimiliki: jmlJamKerja
- Atribut yang diwarisi dari class Staff: nama, alamat, jk, umur, gaji, lembur, potongan

3. Apakah fungsi potongan program berikut pada class StaffHarian

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

Pernyataan `super(nama, alamat, jk, umur, gaji, lembur, potongan);` dalam konstruktor class `StaffHarian` digunakan untuk memanggil konstruktor dari class induk (yaitu class `Staff`) dengan mengirim argumen-argumen yang sesuai ke konstruktor class `Staff`. Hal ini digunakan untuk menginisialisasi atribut-atribut yang diwarisi dari class `Staff`.

4. Apakah fungsi potongan program berikut pada class StaffHarian

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

Pernyataan `super.tampilDataStaff();` dalam method `tampilStaffHarian()` class `StaffHarian` digunakan untuk memanggil method `tampilDataStaff()` dari class `Staff`. Ini memungkinkan class `StaffHarian` untuk menggunakan atau "mewarisi" fungsi-fungsi yang telah didefinisikan dalam 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`?

Pada program `System.out.println("Gaji Bersih = " + (gaji + lembur - potongan - asuransi));`, atribut `gaji`, `lembur`, dan `potongan` adalah bagian dari class `StaffTetap`, tetapi tidak perlu dideklarasikan kembali dalam class `StaffTetap` karena class `StaffTetap` adalah subkelas (subclass) dari class `Staff`. Ini berarti class `StaffTetap` secara otomatis mewarisi atribut-atribut dari class `Staff`, termasuk `gaji`, `lembur`, dan `potongan`. Sehingga, atribut-atribut ini dapat diakses langsung dalam class `StaffTetap` tanpa perlu dideklarasikan kembali. Ini adalah salah satu konsep dasar dalam pewarisan (inheritance) dalam pemrograman berorientasi objek, di mana subclass dapat mengakses atribut dan method yang diwarisi dari superclass.