

Jobsheet 7: INHERITANCE (PEWARISAN)

Object Oriented Programming



**Arranged by :
Shofwah Kanaka Ebsa Anargya
2241720254 / 22
2I**

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

2023

Percobaan 1

1. Class Karyawan

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

2. Class Manager

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

3. Class Staff

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

4. Class Inheritance

```
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 }
```

5. Output

```
/Library/Java/JavaVirtualMachines/jdk-  
↑  
↓  
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  
  
Process finished with exit code 0
```

Pertanyaan

1. Sebutkan class mana yang termasuk super class dan sub class dari percobaan 1 diatas!
Super class : Karyawan
Sub classnya : 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\t\t= "+(super.gaji+tunjangan));
```


Kata kunci super digunakan untuk merujuk ke atribut gaji yang ada di class Karyawan, karena gaji adalah atribut yang diwarisi oleh class Manager dari class Karyawan. Super digunakan untuk merujuk ke atribut yang ada di super class (class induk) dari class saat ini.
5. Program pada percobaan 1 diatas termasuk dalam jenis inheritance apa? Jelaskan alasannya!
Termasuk dalam jenis inheritance "Single Inheritance". Ini karena class Manager dan class Staff mewarisi (inherit) atribut dan metode dari satu class, yaitu class Karyawan. Dalam single inheritance, suatu class hanya memiliki satu class induk (super class).

Percobaan 2

1. Class StaffTetap

```
3 public class StaffTetap extends Staff{
4     2 usages
5     public String golongan;
6     3 usages
7     public int asuransi;
8     no usages
9     public StaffTetap () {
10    }
11    1 usage
12    public StaffTetap (String nama, String alamat, String jk, int umur,
13        int gaji, int lembur, int potongan, String golongan, int asuransi) {
14        super (nama, alamat, jk, umur, gaji, potongan, lembur);
15        this.golongan = golongan;
16        this.asuransi = asuransi;
17    }
18    1 usage
19    public void tampilStaffTetap () {
20        System.out.println("=====Data Staff Tetap=====");
21        super.tampilDataStaff();
22        System.out.println("Golongan\t\t\t= "+golongan);
23        System.out.println("Jumlah Asuransi\t\t= "+asuransi);
24        System.out.println("Gaji Bersih\t\t\t= "+(gaji+lembur-potongan-asuransi));
25    }
26 }
```

2. Class StaffHarian

```
3 public class StaffHarian extends Staff{
4     3 usages
5     public int jmlJamKerja;
6     no usages
7     public StaffHarian () {
8     }
9     1 usage
10    public StaffHarian (String nama, String alamat, String jk, int umur,
11        int gaji, int lembur, int potongan, int jmlJamKerja) {
12        super(nama, alamat, jk, umur, gaji, lembur, potongan);
13        this.jmlJamKerja = jmlJamKerja;
14    }
15    1 usage
16    public void tampilStaffHarian() {
17        System.out.println("=====Data Staff Harian=====");
18        super.tampilDataStaff();
19        System.out.println("Jumlah Jam Kerja\t= "+jmlJamKerja);
20        System.out.println("Gaji Bersih\t\t\t= "+(gaji*jmlJamKerja+lembur-potongan));
21    }
22 }
```

3. Class Inheritance1

```
3 public class Inheritance1 {
4     public static void main(String[] args) {
5         StaffTetap ST= new StaffTetap ( nama: "Budi", alamat: "MaLang", jk: "Lakilaki", umur: 20,
6             gaji: 2000000, lembur: 250000, potongan: 200000, golongan: "2A", asuransi: 100000);
7         ST. tampilStaffTetap();
8         StaffHarian SH= new StaffHarian ( nama: "Indah", alamat: "MaLang", jk: "Perempuan", umur: 27,
9             gaji: 10000, lembur: 100000, potongan: 50000, jmlJamKerja: 100);
10        SH. tampilStaffHarian();
11    }
12 }
```

4. Output

```
/Library/Java/JavaVirtualMachines/jdk-19.jdk/Content
=====Data Staff Tetap=====
Nama           = Budi
Alamat         = Malang
Jenis Kelamin  = Laki-laki
Umur           = 20
Gaji           = 2000000
Lembur         = 200000
Potongan       = 250000
Total Gaji     = 1950000
Golongan       = 2A
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    = 1050000
```

Pertanyaan

1. Berdasarkan class diatas manakah yang termasuk single inheritance dan mana yang termasuk multilevel inheritance?
Class StaffTetap dan Class StaffHarian termasuk dalam single inheritance, karena hanya mewarisi atribut dan metode dari satu class, yaitu 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!
 - Atribut yang dimiliki oleh class StaffTetap adalah: golongan, asuransi
 - Atribut yang diwarisi dari class Staff ke class StaffTetap adalah:
Nama, alamat, jk, umur, gaji, lembur, potongan
 - Atribut yang dimiliki oleh class StaffHarian adalah: jmlJamKerja
 - Atribut yang diwarisi dari class Staff ke class StaffHarian adalah:
Nama, alamat, jk, umur, gaji, lembur, potongan
3. Apakah fungsi potongan program berikut pada class StaffHarian

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

Untuk memanggil konstruktor dari class Staff dengan parameter-parameter yang sesuai. Ini berguna untuk menginisialisasi atribut-atribut yang diwarisi dari class Staff.

4. Apakah fungsi potongan program berikut pada class StaffHarian

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

Untuk memanggil metode tampilDataStaff() dari class Staff. Dengan ini, class StaffHarian bisa menampilkan data yang dimiliki oleh class Staff dan menambahkan data yang dimilikinya sendiri.

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

```
System.out.println("Gaji Bersih\t\t\t= "+(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?

atribut gaji, lembur, dan potongan dapat diakses langsung karena class StaffTetap adalah sub class dari Staff. I class StaffTetap mewarisi atribut-atribut tersebut dari class Staff. Dengan inheritance, class sub (subclass) dapat mengakses atribut dan metode yang diwarisi dari class super (superclass) tanpa perlu mendeklarasikannya ulang.

Tugas

1. Class Komputer

```
3 public class Komputer {
    2 usages
4     String merk;
    2 usages
5     int kecProsesor;
    2 usages
6     int sizeMemory;
    2 usages
7     String jnsProsesor;
8
9     public Komputer() {}
10
11     public Komputer(String merk, int 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\t\t\t: " + merk);
20         System.out.println("Kecepatan Prosesor\t: " + kecProsesor + " GHz");
21         System.out.println("Size Memory\t\t\t: " + sizeMemory + " GB");
22         System.out.println("Jenis Prosesor\t\t: " + jnsProsesor);
23     }
24 }
```

2. Class Laptop

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

3. Class PC

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


4. Class Mac

```
3 public class Mac extends Laptop {
4     2 usages
5     String security;
6
7     no usages
8     public Mac() {}
9
10    1 usage
11    public Mac(String merk, int kecProsesor, int sizeMemory, String jnsProsesor, String jnsBatrei, String security) {
12        super(merk, kecProsesor, sizeMemory, jnsProsesor, jnsBatrei);
13        this.security = security;
14    }
15
16    1 usage
17    public void tampilMac() {
18        tampilLaptop();
19        System.out.println("Security\t\t\t: " + security);
20    }
21 }
```

5. Class Windows

```
3 public class Windows extends Laptop{
4     2 usages
5     String fitur;
6
7     no usages
8     public Windows() {}
9
10    1 usage
11    public Windows(String merk, int kecProsesor, int sizeMemory, String jnsProsesor, String jnsBatrei, String fitur) {
12        super(merk, kecProsesor, sizeMemory, jnsProsesor, jnsBatrei);
13        this.fitur = fitur;
14    }
15
16    1 usage
17    public void tampilWindows() {
18        tampilLaptop();
19        System.out.println("Fitur\t\t\t\t: " + fitur);
20    }
21 }
```

6. Class Main

```
3 public class Main {
4     public static void main(String[] args) {
5         Laptop laptop = new Laptop( merk: "HP", kecProsesor: 2, sizeMemory: 8, jnsProsesor: "Intel i5",
6             jnsBatrei: "Lithium-ion");
7         Pc pc = new Pc( merk: "Dell", kecProsesor: 3, sizeMemory: 16, jnsProsesor: "AMD Ryzen", ukuranMonitor: 24);
8
9         System.out.println("Informasi Laptop:");
10        laptop.tampilLaptop();
11
12        System.out.println("\nInformasi PC:");
13        pc.tampilPc();
14
15        Mac macbook = new Mac( merk: "Apple", kecProsesor: 2, sizeMemory: 16, jnsProsesor: "Intel i7",
16            jnsBatrei: "Lithium-ion", security: "Touch ID");
17        Windows surface = new Windows( merk: "Microsoft", kecProsesor: 1, sizeMemory: 8, jnsProsesor: "Intel i5",
18            jnsBatrei: "Lithium-polymer", fitur: "Surface Pen");
19
20        System.out.println("Informasi Macbook:");
21        macbook.tampilMac();
22
23        System.out.println("\nInformasi Surface:");
24        surface.tampilWindows();
25    }
26 }
```

7. Output

```
↑ /Library/Java/JavaVirtualMachines/jdk-19.jdk/Con
↓
↳ Informasi Laptop:
↳ Merk : HP
↳ Kecepatan Prosesor : 2 GHz
↳ Size Memory : 8 GB
↳ Jenis Prosesor : Intel i5
↳ Jenis Baterai : Lithium-ion

↳ Informasi PC:
↳ Merk : Dell
↳ Kecepatan Prosesor : 3 GHz
↳ Size Memory : 16 GB
↳ Jenis Prosesor : AMD Ryzen
↳ Ukuran Monitor : 24 inch

↳ Informasi Macbook:
↳ Merk : Apple
↳ Kecepatan Prosesor : 2 GHz
↳ Size Memory : 16 GB
↳ Jenis Prosesor : Intel i7
↳ Jenis Baterai : Lithium-ion
↳ Security : Touch ID

↳ Informasi Surface:
↳ Merk : Microsoft
↳ Kecepatan Prosesor : 1 GHz
↳ Size Memory : 8 GB
↳ Jenis Prosesor : Intel i5
↳ Jenis Baterai : Lithium-polymer
↳ Fitur : Surface Pen

Process finished with exit code 0
```