

**LAPORAN PRAKTIKUM**  
**Pemrograman Berbasis Objek**



Nama : Ghoffar Abdul Ja'far

NIM : 2341720035

Kelas : 2F

**JURUSAN TEKNOLOGI INFORMASI**  
**POLITEKNIK NEGERI MALANG**

**2023/2024**

# Percobaan 1

## Code:

- Karyawan

```
public class Karyawan {
    public String nama, alamat, jk;
    public int umur, gaji;

    public Karyawan() {
    }

    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;
    }

    public void tampilDataKaryawan() {
        System.out.println("Nama      =" + nama);
        System.out.println("Alamat   =" + alamat);
        System.out.println("Jenis Kelamin =" + jk);
        System.out.println("Umur     =" + umur);
        System.out.println("Gaji     =" + gaji);
    }
}
```

- Manager

```
public class Manager extends Karyawan {
    public int tunjangan;

    public Manager() {
    }

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

- Staff

```
public class Staff extends Karyawan {
    public int lembur, potongan;

    public Staff() {
    }

    public Staff(int lembur, int potongan, String nama, String alamat, String jk, int umur, int gaji) {
        super(nama, alamat, jk, umur, gaji);
        this.lembur = lembur;
        this.potongan = potongan;
    }

    public void tampilDataStaff() {
        super.tampilDataKaryawan();
        System.out.println("Lembur      =" + lembur);
        System.out.println("Potongan    =" + potongan);
        System.out.println("Total Gaji  =" + (super.gaji + lembur - potongan));
    }
}
```

- Inheritance1

```
public class Inheritance1 {
    public static void main(String[] args) {
        Manager M = new Manager();
        M.nama = "Vivin";
        M.alamat = "Jl. Vinolia";
        M.umur = 25;
        M.jk = "Perempuan";
        M.gaji = 3000000;
        M.tunjangan = 1000000;
        M.tampilDataManager();

        Staff S = new Staff();
        S.nama = "Lestari";
        S.alamat = "Malang";
        S.umur = 25;
        S.jk = "Perempuan";
        S.gaji = 2000000;
        S.lembur = 500000;
        S.potongan = 250000;
        S.tampilDataStaff();
    }
}
```

## Hasil:

```

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
BUILD SUCCESSFUL (total time: 0 seconds)

```

## Pertanyaan:

1. Sebutkan class mana yang termasuk super class dan sub class dari percobaan 1 diatas!  
= **Superclass yaitu Class Karyawan. Subclass yaitu class Manager dan Staff**
2. Kata kunci apakah yang digunakan untuk menurunkan suatu class ke class yang lain?  
= **Yaitu kunci pada public class Manager extends Karyawan. Karna Manager adalah subclass yang diturunkan dari Karyawan sebagai Superclass menggunakan kata kunci 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!  
= **Tunjangan. Lalu atribut yang diwarisi dari class Karyawan 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));`  
= **Berfungsi untuk mengakses anggota (atribut atau metode) dari superclass karyawan dalam subclass Manager**
5. Program pada percobaan 1 diatas termasuk dalam jenis inheritance apa? Jelaskan alasannya!  
= **Termasuk pada Single inheritance ketika sebuah subclass hanya mewarisi dari satu superclass. Di Class Manager dan Staff masing masing hanya mewarisi dari satu superclass, yaitu Karyawan, Dengan kata lain Manager dan Staff merupakan subclass yang hanya memiliki satu superclass yaitu Karyawan**

## Percobaan 2

### Code:

- StaffTetap

```
public class StaffTetap extends Staff{
    public String golongan;
    public int asuransi;

    public StaffTetap() {
    }

    public StaffTetap(String golongan, int asuransi, int lembur, int potongan, String nama, String alamat, String jk, int umur, int gaji) {
        super(lembur, potongan, nama, alamat, jk, umur, gaji);
        this.golongan = golongan;
        this.asuransi = asuransi;
    }

    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));
    }
}
```

- StaffHarian

```
public class StaffHarian extends Staff{
    public int jmlJamKerja;

    public StaffHarian() {
    }

    public StaffHarian(int jmlJamKerja, int lembur, int potongan, String nama, String alamat, String jk, int umur, int gaji) {
        super(lembur, potongan, nama, alamat, jk, umur, gaji);
        this.jmlJamKerja = jmlJamKerja;
    }

    public void tampilSatffharian(){
        System.out.println("=====Data Staff Harian=====");
        super.tampilDataStaff();
        System.out.println("Golongan      =" + jmlJamKerja);
        System.out.println("Gaji Bersih    =" + (gaji*jmlJamKerja-asuransi-potongan));
    }
}
```

- Inheritance1

```
public class Inheritance1 {
    public static void main(String[] args) {
        StaffTetap ST = new StaffTetap("Budi", "Malang", "Lakilaki", 20, 2000000, 250000, 200000, "2A", 100000);
        ST.tampilStaffTetap ();

        StaffHarian SH = new StaffHarian ("Indah", "Malang", "Perempuan", 27, 10000, 100000, 50000, 100);
        SH.tampilStaffHarian();
    }
}
```

### Hasil:

```

=====Data Staff Tetap=====
Nama          =Malang
Alamat        =Lakilaki
Jenis Kelamin =2A
Umur          =200000
Gaji          =100000
Lembur        =2000000
Potongan      =250000
Total Gaji    =1850000
Golongan      =Budi
Jumlah Asuransi =20
Gaji Bersih   =1849980
=====Data Staff Harian=====
Nama          =Indah
Alamat        =Malang
Jenis Kelamin =Perempuan
Umur          =50000
Gaji          =100
Lembur        =10000
Potongan      =100000
Total Gaji    =-89900
Golongan      =27
Gaji Bersih   =-107300
BUILD SUCCESSFUL (total time: 0 seconds)

```

### Pertanyaan:

1. Berdasarkan class diatas manakah yang termasuk single inheritance dan mana yang termasuk multilevel inheritance?  
 = **Single Inheritance** adalah Staff dari subclass Karyawan, Dan Manager & StaffHarian serta StaffTetap masing masing adalah subclass Karyawan
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!  
 = **Dari Class StaffTetap atribut yang dimiliki adalah golongan & asuransi**  
**Atribut yang diwarisi class Staff adalah : nama, alamat, jk, umur, gaji, lembur, potongan**  
**Dari Class StaffHarian atribut yang dimiliki adalah jmlkerja**  
**Atribut yang diwarisi oleh 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);`  
 = **Fungsi nya memanggil Constructor Superclass. Menginisialisasi Atribut. Mewarisi isi dari Superclass**
4. Apakah fungsi potongan program berikut pada class StaffHarian  
`super.tampilDataStaff();`  
 = **Fungsi Memanggil Method dari Superclass, Menampilkan Data Staff, Mewarisi dan Menggabungkan Fungsionalitas**

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 Pewarisan Inheritance. Didalam class stafftetap adalah subclass dari class staff. Dan untuk**

**Gaji, Lembur dan potongan adalah atribut yang dideklarasikan secara eksplisit dalam class StaffTetap, dan Dapat mengaksesnya karena Mewarisi dari Staff**

**Yang kedua karena protected atau public dalam class staff dan dapat diakses secara langsung dalam Class StaffTetap**

**Jadi atribut gaji lembur dan potongan dapat diakses langsung dalam class stafftetap karena class tersebut adalah subclass dari staff yang akan mendeklarasikan atribut atribut tersebut**