

# LAPORAN PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK

## Pertemuan 3

Nama: Aditya Alfandy

NIM: H1D024103

### A. Alur Kerja Program

Program ini mensimulasikan sistem data pegawai untuk startup "TechMaju" menggunakan konsep Inheritance (Pewarisan). Tujuannya adalah membedakan perlakuan data antara karyawan biasa dan manajer tanpa perlu menulis ulang kode dasar. Alur kerjanya adalah sebagai berikut:

#### 1. Hierarki Class (Pewarisan):

- Class Parent (Karyawan): Menyimpan data dasar yang dimiliki semua pegawai, yaitu Nama dan Gaji Pokok.
- Class Child (Manajer): Mewarisi sifat-sifat Karyawan menggunakan keyword extends. Manajer memiliki data tambahan berupa Tunjangan yang tidak dimiliki karyawan biasa.

#### 2. Inisialisasi Objek:

Pada class main UjiKaryawan, program membuat dua jenis objek:

- Objek Karyawan (Budi): Dibuat hanya dengan data Nama dan Gaji Pokok.
- Objek Manajer (Siti): Dibuat dengan data Nama, Gaji Pokok, dan Tunjangan. Constructor Manajer menggunakan super(...) untuk mengirim data nama dan gaji ke constructor induknya.

#### 3. Polimorfisme (Overriding):

Program menampilkan informasi menggunakan method tampilInfo().

- Untuk Karyawan Biasa, method ini hanya menampilkan Nama dan Gaji Pokok.
- Untuk Manajer, terjadi proses Overriding (penimpaan method). Method tampilInfo() di class Manajer memodifikasi perilaku aslinya untuk menampilkan tambahan Tunjangan dan menghitung Total Pendapatan (Gaji Pokok + Tunjangan).

### B. Fungsi dan Method yang Digunakan

Berikut adalah penjelasan method yang diimplementasikan:

No	Nama Method	Class	Fungsi / Kegunaan

1.	Karyawan(...)	Karyawan	<i>Constructor</i> dasar untuk menginisialisasi nama dan gaji pokok pegawai.
2.	Manajer(...)	Manajer	<i>Constructor</i> khusus Manajer. Menggunakan keyword <i>super</i> untuk memanggil constructor Karyawan, lalu menginisialisasi variabel tunjangan.
3.	tampilInfo()	Karyawan	Menampilkan data standar pegawai (Nama dan Gaji Pokok) ke layar konsol.
4.	tampilInfo()	Manajer	Override Method: Menimpa method milik parent. Berfungsi menampilkan data standar, ditambah data Tunjangan, dan hasil perhitungan Total Gaji.

### C. Implementasi Kode (Source Code)

#### 1. File: Karyawan.java (Parent Class)

```
class Karyawan {
    protected String nama;
    protected double gajiPokok;

    public Karyawan(String nama, double gajiPokok) {
        this.nama = nama;
        this.gajiPokok = gajiPokok;
    }

    public void tampilInfo() {
        System.out.println("Nama: " + nama + " | Gaji Pokok: Rp " + gajiPokok);
    }
}
```

#### 2. File: Manajer.java (Child Class)

```
class Manajer extends Karyawan {
    double tunjangan;

    public Manajer(String nama, double gajiPokok, double tunjangan) {
        super(nama, gajiPokok);
        this.tunjangan = tunjangan;
    }

    @Override
    public void tampilInfo() {
```

```

        System.out.println("Nama: " + nama + " | Gaji Pokok: Rp " + gajiPokok +
            " | Tunjangan: Rp " + tunjangan);

        double totalGaji = gajiPokok + tunjangan;
        System.out.println("Total Pendapatan: Rp " + totalGaji);
    }
}

```

### 3. File: UjiKaryawan.java (Main Class)

```

public class UjiKaryawan {
    public static void main(String[] args) {
        System.out.println("=== DATA KARYAWAN TECHMAJU ===");

        Karyawan karyawan1 = new Karyawan("Budi Santoso", 4000000);
        System.out.println("Status: Karyawan Biasa");
        karyawan1.tampilInfo();

        System.out.println();

        Manajer manajer1 = new Manajer("Siti Aminah", 6000000, 2500000);
        System.out.println("Status: Manajer");
        manajer1.tampilInfo();
    }
}

```

### D. Hasil Output Program

```

=== DATA KARYAWAN TECHMAJU ===
Status: Karyawan Biasa
Nama: Budi Santoso | Gaji Pokok: Rp 4000000.0

Status: Manajer
Nama: Siti Aminah | Gaji Pokok: Rp 6000000.0 | Tunjangan: Rp 2500000.0
Total Pendapatan: Rp 8500000.0

```