

Nama : Khalisa Nadya Hakim

NIM : 24060123140149

Lab : C2

## 1. Screenshot Kode dan Output

### Pegawai.java

```
J Pegawai.java > ...
1 public class Pegawai {
2     protected String nama;
3     protected int gajiPokok = 5000000;
4
5     public void setNama(String nama) {
6         this.nama = nama;
7     }
8
9     public void tampilData() {
10        System.out.println("Nama : " + nama + ", Gaji pokok : " + gajiPokok);
11    }
12 }
13
```

### Manajer.java

```
J Manajer.java > ...
1 public class Manajer extends Pegawai {
2     private int tunjangan = 700000;
3
4     public Manajer(String nama) {
5         this.nama = nama;
6     }
7
8     @Override
9     public void tampilData() {
10        super.tampilData();
11        System.out.println("Tunjangan : " + tunjangan);
12    }
13 }
14
```

## Programmer.java

```
J Programmer.java > ...
1  public class Programmer extends Pegawai {
2      private int bonus = 450000;
3
4      public Programmer(String nama) {
5          this.nama = nama;
6      }
7
8      @Override
9      public void tampilData() {
10         super.tampilData();
11         System.out.println("Bonus : " + bonus);
12     }
13 }
14
15
```

## TestPolimorfisme.java dan Output

```
J TestPolimorfisme.java > TestPolimorfisme
1  import java.util.ArrayList;
2
3  public class TestPolimorfisme {
4      public static void main(String[] args) {
5          Pegawai pegawai1 = new Programmer(nama:"Mira");
6          Pegawai pegawai2 = new Manajer(nama:"Joko");
7          Manajer pegawai3 = new Manajer(nama:"Argo");
8
9          ArrayList<Pegawai> emps = new ArrayList<>();
10         emps.add(pegawai1);
11         emps.add(pegawai2);
12         emps.add(pegawai3);
13
14         for (Pegawai emp : emps) {
15             emp.tampilData();
16             System.out.println();
17         }
18     }
19 }
20
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Khalisa Nadya\POST TEST PBO 6> cd "c:\Users\Khalisa Nadya\POST TEST PBO 6\" ; if ($?) { javac TestPolimorfisme.java } ; if ($?) { java T
estPolimorfisme }
Nama : Mira, Gaji pokok : 5000000
Bonus : 450000

Nama : Joko, Gaji pokok : 5000000
Tunjangan : 700000

Nama : Argo, Gaji pokok : 5000000
Tunjangan : 700000

PS C:\Users\Khalisa Nadya\POST TEST PBO 6>
```

2. Jelaskan manfaat polimorfisme pada kasus ini.

Polimorfisme pada kasus ini bermanfaat untuk menyederhanakan pengolahan objek berbeda (Programmer dan Manajer) dalam satu struktur yang sama, yaitu `ArrayList<Pegawai>`. Dengan polimorfisme, kita bisa memanggil metode `tampilData()` tanpa harus tahu jenis objeknya, karena Java secara otomatis menjalankan metode yang sesuai dengan tipe asli objek tersebut. Ini membuat kode lebih sederhana, fleksibel, mudah dikembangkan, dan memudahkan penambahan jenis pegawai baru di masa depan tanpa perlu banyak perubahan pada program utama.

3. Apabila pada main program perlu menambahkan pegawai4 dan pegawai5! Apa permasalahan yang muncul jika diterapkan tanpa polimorfisme (inclusion)?

Tanpa menggunakan polimorfisme, penambahan pegawai4 dan pegawai5 akan menimbulkan masalah dalam hal fleksibilitas dan efisiensi kode. Kita harus mengecek tipe masing-masing objek secara manual, misalnya dengan if-else atau instanceof, untuk menentukan apakah objek tersebut merupakan Manajer, Programmer, atau Pegawai biasa, sebelum memanggil metode yang sesuai. Hal ini membuat kode menjadi lebih panjang, sulit dibaca, dan rentan terhadap kesalahan, terutama jika jumlah jenis pegawai terus bertambah. Akibatnya, program menjadi tidak modular dan lebih sulit untuk dikembangkan atau dimodifikasi di masa depan.