

Nama : George Ryan Saputra

NPM : 5240311048

Kelas : Pemrograman Berorientasi Objek (B)

Studi : Sistem Informasi

## TUGAS PBO

### IMPLEMENTASI OVERLOADING, OVERRIDING, DAN ABSTRAKSI

Soal :

```
PenumpangReguler.java
1 import java.util.Scanner;
2
3 abstract class Penumpang {
4     protected String nama;
5     protected String noTiket;
6
7     public Penumpang(String nama, String noTiket) {
8         this.nama = nama;
9         this.noTiket = noTiket;
10    }
11
12    public abstract double hitungHargaTiket();
13
14    public void tampilkanData() {
15        System.out.println("Nama      : " + nama);
16        System.out.println("No Tiket   : " + noTiket);
17        System.out.println("Harga Tiket : Rp " + hitungHargaTiket());
18    }
19 }
20
21 class PenumpangReguler extends Penumpang {
22     public PenumpangReguler(String nama, String noTiket) {
23         super(nama, noTiket);
24     }
25 }
26
27 class PenumpangVIP extends Penumpang {
28     public PenumpangVIP(String nama, String noTiket) {
29         super(nama, noTiket);
30     }
31 }
32
33 class InputPenumpang {
34     protected Scanner input = new Scanner(System.in);
35 }
```

Jawaban :

#### 1. Lengkapi program di atas tanpa mengubah struktur kelas

Saya melengkapi program dengan menambahkan:

- Implementasi method di class PenumpangRegular dan PenumpangVIP
- Class InputPenumpang dengan method input
- Class Main sebagai entry point program
- Struktur kelas tetap sama: 4 class dari soal + 1 class Main baru

**Source Code**

```
package tiketbus;

/**
 *
 * @author George Ryan Saputra
 */
import java.util.Scanner;

abstract class Penumpang {
    protected String nama;
    protected String noTiket;

    public Penumpang(String nama, String noTiket) {
        this.nama = nama;
        this.noTiket = noTiket;
    }

    public abstract double hitungHargaTiket();

    public void tampilkanData() {
        System.out.println("\n==== DATA DASAR ====");
        System.out.println("Nama : " + nama);
        System.out.println("No Tiket : " + noTiket);
        System.out.println("Harga : Rp " + hitungHargaTiket());
    }

    public void tampilkanData(String pesan) {
        tampilkanData();
        System.out.println("Keterangan: " + pesan);
    }

    public void tampilkanData(boolean detail) {
        System.out.println("\n==== DATA LENGKAP ====");
        System.out.println("Nama Penumpang : " + nama);
        System.out.println("Nomor Tiket : " + noTiket);
        if (detail) {
            System.out.println("Harga Final : Rp " + hitungHargaTiket());
            System.out.println("(Termasuk semua biaya)");
        }
    }
}

package tiketbus;

/**
 *
 * @author George Ryan Saputra
 */
class PenumpangRegular extends Penumpang {
    private double hargaDasar = 150000;
```

```

private double biayaAdmin = 5000;

public PenumpangRegular(String nama, String noTiket) {
    super(nama, noTiket);
}

@Override
public double hitungHargaTiket() {
    return hargaDasar + biayaAdmin;
}

public void tampilkanData(String kursi, String jam) {
    System.out.println("\n==== TIKET REGULAR ====");
    System.out.println("Atas Nama : " + nama);
    System.out.println("Nomor : " + noTiket);
    System.out.println("Kursi : " + kursi);
    System.out.println("Jam Berangkat: " + jam);
    System.out.println("-----");
    System.out.println("Harga Dasar : Rp " + hargaDasar);
    System.out.println("Biaya Admin : Rp " + biayaAdmin);
    System.out.println("TOTAL : Rp " + hitungHargaTiket());
    System.out.println("=====");
}
}

package tiketbus;

/**
 *
 * @author George Ryan Saputra
 */
class PenumpangVIP extends Penumpang {
    private double hargaDasar = 350000;
    private double diskon = 0.20;
    private double biayaAdmin = 10000;

    public PenumpangVIP(String nama, String noTiket) {
        super(nama, noTiket);
    }

    @Override
    public double hitungHargaTiket() {
        double hargaSetelahDiskon = hargaDasar - (hargaDasar * diskon);
        return hargaSetelahDiskon + biayaAdmin;
    }

    public void tampilkanData(String fasilitas) {
        System.out.println("\n==== TIKET VIP ====");
        System.out.println("Atas Nama : " + nama);
        System.out.println("Nomor : " + noTiket);
        System.out.println("Fasilitas : " + fasilitas);
    }
}

```

```

        System.out.println("-----");
        System.out.println("Harga Dasar : Rp " + hargaDasar);
        System.out.println("Diskon 20% : -Rp " + (hargaDasar * diskon));
        System.out.println("Biaya Admin : Rp " + biayaAdmin);
        System.out.println("TOTAL : Rp " + hitungHargaTiket());
        System.out.println("=====");
    }

}

package tiketbus;

/**
 *
 * @author George Ryan Saputra
 */
import java.util.Scanner;

class InputPenumpang {
    protected Scanner input = new Scanner(System.in);

    public String[] ambilDataPenumpang() {
        String[] data = new String[3];

        System.out.print("Masukkan Nama Penumpang : ");
        data[0] = input.nextLine();

        System.out.print("Masukkan Nomor Tiket : ");
        data[1] = input.nextLine();

        String jenis;
        do {
            System.out.print("Jenis Tiket (R=Regular, V=VIP): ");
            jenis = input.nextLine().toUpperCase();
            if (!jenis.equals("R") && !jenis.equals("V")) {
                System.out.println("Error: Hanya R atau V yang diperbolehkan!");
            }
        } while (!jenis.equals("R") && !jenis.equals("V"));

        data[2] = jenis;
        return data;
    }

    public String ambilFasilitasVIP() {
        System.out.print("Fasilitas VIP yang diinginkan: ");
        return input.nextLine();
    }

    public String[] ambilDetailRegular() {
        String[] detail = new String[2];

        System.out.print("Nomor Kursi : ");

```

```
detail[0] = input.nextLine();

System.out.print("Jam Keberangkatan : ");
detail[1] = input.nextLine();

return detail;
}

}

package tiketbus;

/**
 *
 * @author George Ryan Saputra
 */
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        InputPenumpang input = new InputPenumpang();

        System.out.println("== SISTEM PEMESANAN TIKET BUS ==");

        // Input data
        String[] data = input.ambilDataPenumpang();
        String nama = data[0];
        String noTiket = data[1];
        String jenis = data[2];

        Penumpang penumpang;

        // Proses berdasarkan jenis
        if (jenis.equals("R")) {
            penumpang = new PenumpangRegular(nama, noTiket);
            System.out.println("\n> Tiket Regular Berhasil Dibuat");

            // Ambil detail kursi
            String[] detail = input.ambilDetailRegular();
            String kursi = detail[0];
            String jam = detail[1];

            // Tampilkan tiket regular
            ((PenumpangRegular) penumpang).tampilkanData(kursi, jam);
        } else {
            penumpang = new PenumpangVIP(nama, noTiket);
            System.out.println("\n> Tiket VIP Berhasil Dibuat");

            // Ambil fasilitas
            String fasilitas = input.ambilFasilitasVIP();
        }
    }
}
```

```

        ((PenumpangVIP) penumpang).tampilkanData(fasilitas);
    }

    // Tampilkan ringkasan akhir
    System.out.println("\n==== RINGKASAN PEMESANAN ====");
    System.out.println("Nama : " + penumpang.nama);
    System.out.println("No. Tiket : " + penumpang.noTiket);
    System.out.println("Total Bayar: Rp " + penumpang.hitungHargaTiket());
    System.out.println("=====");
}

// Tampilkan info tipe (opsional, bukti polimorfisme)
System.out.println("\n[Info: Objek bertipe " + penumpang.getClass().getSimpleName() + "]");
scanner.close();
}
}

```

## 2. Terapkan Abstraksi, Overriding, Overloading

### a. Abstraksi pada class Penumpang:

```

/*
 * @author George Ryan Saputra
 */
import java.util.Scanner;

abstract class Penumpang {
    protected String nama;
    protected String noTiket;

    public Penumpang(String nama, String noTiket) {
        this.nama = nama;
        this.noTiket = noTiket;
    }

    public abstract double hitungHargaTiket();
}

```

**Penjelasan:** Class Penumpang dibuat abstract sehingga tidak bisa diinstansiasi langsung.  
Method hitungHargaTiket() dideklarasikan abstract sehingga wajib diimplementasikan subclass.

### b. Overriding pada method hitungHargaTiket():

```

DI FILE PenumpangRegular
@Override
public double hitungHargaTiket() {
    return hargaDasar + biayaAdmin;
}

```

```

DI FILE PenumpangVIP
@Override
public double hitungHargaTiket() {
    double hargaSetelahDiskon = hargaDasar - (hargaDasar * diskon);
    return hargaSetelahDiskon + biayaAdmin;
}

```

**Penjelasan:** Method hitungHargaTiket() di-override di subclass dengan implementasi yang berbeda.

### c. Overloading pada method tampilkanData():

```

DI FILE Penumpang

```

```

    public void tampilanData() {
        System.out.println("\n==== DATA DASAR ====");
        System.out.println("Nama      : " + nama);
        System.out.println("No Tiket  : " + noTiket);
        System.out.println("Harga     : Rp " + hitungHargaTiket());
    }

    public void tampilanData(String pesan) {
        tampilanData();
        System.out.println("Keterangan: " + pesan);
    }
}

```

**Penjelasan:** Method tampilanData() punya 2 versi di class yang sama dengan parameter berbeda.

### 3.Gunakan Scanner untuk menerima input

*DI FILE InputPenumpang*

```

class InputPenumpang {
    protected Scanner input = new Scanner(System.in);

    public String[] ambilDataPenumpang() {
        String[] data = new String[3];

        System.out.print("Masukkan Nama Penumpang : ");
        data[0] = input.nextLine();
    }
}

```

**Penjelasan:** Scanner digunakan di class InputPenumpang untuk mengambil input dari user.

### 4.Buat class Main sendiri

*DI FILE Main*

```

package tiketbus;

/**
 * 
 * @author George Ryan Saputra
 */
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        InputPenumpang input = new InputPenumpang();

        System.out.println("==== SISTEM PEMESANAN TIKET BUS ====");

        // Input data
        String[] data = input.ambilDataPenumpang();
        String nama = data[0];
        String noTiket = data[1];
        String jenis = data[2];

        Penumpang penumpang;

        // Proses berdasarkan jenis
        if (jenis.equals("R")) {
            penumpang = new PenumpangRegular(nama, noTiket);
            System.out.println("\n>> Tiket Regular Berhasil Dibuat");
        }
    }
}

```

```

        // Ambil detail kursi
        String[] detail = input.ambilDetailRegular();
        String kursi = detail[0];
        String jam = detail[1];
        |
        // Tampilkan tiket regular
        ((PenumpangRegular) penumpang).tampilkanData(kursi, jam);

    } else {
        penumpang = new PenumpangVIP(nama, noTiket);
        System.out.println("\n>> Tiket VIP Berhasil Dibuat");

        // Ambil fasilitas
        String fasilitas = input.ambilFasilitasVIP();
        ((PenumpangVIP) penumpang).tampilkanData(fasilitas);
    }

    // Tampilkan ringkasan akhir
    System.out.println("\n==== RINGKASAN PEMESANAN ===");
    System.out.println("Nama : " + penumpang.nama);
    System.out.println("No. Tiket : " + penumpang.noTiket);
    System.out.println("Total Bayar: Rp " + penumpang.hitungHargaTiket());
    System.out.println("=====");

    // Tampilkan info tipe (opsional, bukti polimorfisme)
    System.out.println("\n[Info: Objek bertipe " + penumpang.getClass().getSimpleName() + "]");

    scanner.close();
}
}

```

**Penjelasan:** Class Main dibuat sebagai entry point yang menangani seluruh alur program.

## 5. Program harus dapat membedakan penumpang Reguler dan VIP

### DI FILE Main

```

// Proses berdasarkan jenis
if (jenis.equals("R")) {
    penumpang = new PenumpangRegular(nama, noTiket);
    System.out.println("\n>> Tiket Regular Berhasil Dibuat");

    // Ambil detail kursi
    String[] detail = input.ambilDetailRegular();
    String kursi = detail[0];
    String jam = detail[1];
    |
    // Tampilkan tiket regular
    ((PenumpangRegular) penumpang).tampilkanData(kursi, jam);

} else {
    penumpang = new PenumpangVIP(nama, noTiket);
    System.out.println("\n>> Tiket VIP Berhasil Dibuat");

    // Ambil fasilitas
    String fasilitas = input.ambilFasilitasVIP();
    ((PenumpangVIP) penumpang).tampilkanData(fasilitas);
}

```

**Penjelasan:** Program menggunakan if-else untuk membedakan input 'R' (Regular) dan 'V' (VIP).

## 6. Berikan penjelasan terhadap hasil penerapan polimorfisme dan abstract

### a. Polimorfisme:

- **Variabel polimorfik:** Penumpang penumpang; bisa merujuk ke objek PenumpangRegular atau PenumpangVIP
- **Late binding:** Saat memanggil penumpang.hitungHargaTiket(), Java akan memilih implementasi yang sesuai dengan objek aktual

- **Contoh:** Jika objek adalah PenumpangRegular, maka yang dipanggil adalah implementasi di class PenumpangRegular

**b. Abstraksi:**

- **Abstract class:** Penumpang tidak bisa diinstansiasi langsung (new Penumpang() error)
- **Abstract method:** hitungHargaTiket() hanya dideklarasikan, implementasinya di subclass
- **Keuntungan:** Menyembunyikan detail implementasi, fokus pada antarmuka

**7. Upload ke Git dan buat laporan**

[https://github.com/GeorgeRyanSaputra/5240311048\\_George-Ryan-Saputra.git](https://github.com/GeorgeRyanSaputra/5240311048_George-Ryan-Saputra.git)

**8. Hasil kode dan Hasil running**

**Hasil Kode :**

```
package tiketbus;

/**
 *
 * @author George Ryan Saputra
 */
import java.util.Scanner;

abstract class Penumpang {
    protected String nama;
    protected String noTiket;

    public Penumpang(String nama, String noTiket) {
        this.nama = nama;
        this.noTiket = noTiket;
    }

    public abstract double hitungHargaTiket();

    public void tampilkanData() {
        System.out.println("\n==== DATA DASAR ====");
        System.out.println("Nama : " + nama);
        System.out.println("No Tiket : " + noTiket);
        System.out.println("Harga : Rp " + hitungHargaTiket());
    }

    public void tampilkanData(String pesan) {
        tampilkanData();
        System.out.println("Keterangan: " + pesan);
    }

    public void tampilkanData(boolean detail) {
        System.out.println("\n==== DATA LENGKAP ====");
        System.out.println("Nama Penumpang : " + nama);
        System.out.println("Nomor Tiket : " + noTiket);
        if (detail) {
            System.out.println("Harga Final : Rp " + hitungHargaTiket());
        }
    }
}
```

```
        System.out.println("(Termasuk semua biaya)");
    }
}
}

package tiketbus;

/**
 *
 * @author George Ryan Saputra
 */
class PenumpangRegular extends Penumpang {
    private double hargaDasar = 150000;
    private double biayaAdmin = 5000;

    public PenumpangRegular(String nama, String noTiket) {
        super(nama, noTiket);
    }

    @Override
    public double hitungHargaTiket() {
        return hargaDasar + biayaAdmin;
    }

    public void tampilkanData(String kursi, String jam) {
        System.out.println("\n== TIKET REGULAR ==");
        System.out.println("Atas Nama : " + nama);
        System.out.println("Nomor : " + noTiket);
        System.out.println("Kursi : " + kursi);
        System.out.println("Jam Berangkat: " + jam);
        System.out.println("-----");
        System.out.println("Harga Dasar : Rp " + hargaDasar);
        System.out.println("Biaya Admin : Rp " + biayaAdmin);
        System.out.println("TOTAL : Rp " + hitungHargaTiket());
        System.out.println("=====");
    }
}

package tiketbus;

/**
 *
 * @author George Ryan Saputra
 */
class PenumpangVIP extends Penumpang {
    private double hargaDasar = 350000;
    private double diskon = 0.20;
    private double biayaAdmin = 10000;

    public PenumpangVIP(String nama, String noTiket) {
        super(nama, noTiket);
    }
```

```

}

@Override
public double hitungHargaTiket() {
    double hargaSetelahDiskon = hargaDasar - (hargaDasar * diskon);
    return hargaSetelahDiskon + biayaAdmin;
}

public void tampilkanData(String fasilitas) {
    System.out.println("\n==== TIKET VIP ===");
    System.out.println("Atas Nama : " + nama);
    System.out.println("Nomor : " + noTiket);
    System.out.println("Fasilitas : " + fasilitas);
    System.out.println("-----");
    System.out.println("Harga Dasar : Rp " + hargaDasar);
    System.out.println("Diskon 20% : -Rp " + (hargaDasar * diskon));
    System.out.println("Biaya Admin : Rp " + biayaAdmin);
    System.out.println("TOTAL : Rp " + hitungHargaTiket());
    System.out.println("=====");
}
}

package tiketbus;

/**
 *
 * @author George Ryan Saputra
 */
import java.util.Scanner;

class InputPenumpang {
    protected Scanner input = new Scanner(System.in);

    public String[] ambilDataPenumpang() {
        String[] data = new String[3];

        System.out.print("Masukkan Nama Penumpang : ");
        data[0] = input.nextLine();

        System.out.print("Masukkan Nomor Tiket : ");
        data[1] = input.nextLine();

        String jenis;
        do {
            System.out.print("Jenis Tiket (R=Regular, V=VIP): ");
            jenis = input.nextLine().toUpperCase();
            if (!jenis.equals("R") && !jenis.equals("V")) {
                System.out.println("Error: Hanya R atau V yang diperbolehkan!");
            }
        } while (!jenis.equals("R") && !jenis.equals("V"));
    }
}

```

```
        data[2] = jenis;
        return data;
    }

    public String ambilFasilitasVIP() {
        System.out.print("Fasilitas VIP yang diinginkan: ");
        return input.nextLine();
    }

    public String[] ambilDetailRegular() {
        String[] detail = new String[2];

        System.out.print("Nomor Kursi      : ");
        detail[0] = input.nextLine();

        System.out.print("Jam Keberangkatan  : ");
        detail[1] = input.nextLine();

        return detail;
    }
}

package tiketbus;

/**
 *
 * @author George Ryan Saputra
 */
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        InputPenumpang input = new InputPenumpang();

        System.out.println("== SISTEM PEMESANAN TIKET BUS ==");

        // Input data
        String[] data = input.ambilDataPenumpang();
        String nama = data[0];
        String noTiket = data[1];
        String jenis = data[2];

        Penumpang penumpang;

        // Proses berdasarkan jenis
        if (jenis.equals("R")) {
            penumpang = new PenumpangRegular(nama, noTiket);
            System.out.println("\n> Tiket Regular Berhasil Dibuat");

        // Ambil detail kursi
    }
}
```

```

String[] detail = input.ambilDetailRegular();
String kursi = detail[0];
String jam = detail[1];

// Tampilkan tiket regular
((PenumpangRegular) penumpang).tampilkanData(kursi, jam);

} else {
    penumpang = new PenumpangVIP(nama, noTiket);
    System.out.println("\n>> Tiket VIP Berhasil Dibuat");

    // Ambil fasilitas
    String fasilitas = input.ambilFasilitasVIP();
    ((PenumpangVIP) penumpang).tampilkanData(fasilitas);
}

// Tampilkan ringkasan akhir
System.out.println("\n== RINGKASAN PEMESANAN ==");
System.out.println("Nama : " + penumpang.nama);
System.out.println("No. Tiket : " + penumpang.noTiket);
System.out.println("Total Bayar: Rp " + penumpang.hitungHargaTiket());
System.out.println("=====");

// Tampilkan info tipe (opsional, bukti polimorfisme)
System.out.println("\n[Info: Objek bertipe " + penumpang.getClass().getSimpleName() + "]");

scanner.close();
}
}

```

### Hasil Running :

*SCREENSHOT 1: Running Tiket Regular*

```

run:
==== SISTEM PEMESANAN TIKET BUS ====
Masukkan Nama Penumpang : George Ryan Saputra
Masukkan Nomor Tiket : REG001
Jenis Tiket (R=Regular, V=VIP): R

>> Tiket Regular Berhasil Dibuat
Nomor Kursi : 23
Jam Keberangkatan : 09.00

==== TIKET REGULAR ===
Atas Nama : George Ryan Saputra
Nomor : REG001
Kursi : 23|
Jam Berangkat: 09.00
-----
Harga Dasar : Rp 150000.0
Biaya Admin : Rp 5000.0
TOTAL : Rp 155000.0
=====

==== RINGKASAN PEMESANAN ===
Nama : George Ryan Saputra
No. Tiket : REG001
Total Bayar: Rp 155000.0
=====

[Info: Objek bertipe PenumpangRegular]
BUILD SUCCESSFUL (total time: 2 minutes 38 seconds)

```

*SCREENSHOT 2: Running Tiket VIP*

```
run:  
==== SISTEM PEMESANAN TIKET BUS ====  
Masukkan Nama Penumpang : Isna Nurul  
Masukkan Nomor Tiket : VIP001  
Jenis Tiket (R=Regular, V=VIP) : V  
  
>> Tiket VIP Berhasil Dibuat  
Fasilitas VIP yang diinginkan: Lounge  
  
==== TIKET VIP ====  
Atas Nama : Isna Nurul  
Nomor : VIP001  
Fasilitas : Lounge  
-----  
Harga Dasar : Rp 350000.0  
Diskon 20% : -Rp 70000.0  
Biaya Admin : Rp 10000.0  
TOTAL : Rp 290000.0  
=====  
  
==== RINGKASAN PEMESANAN ====  
Nama : Isna Nurul  
No. Tiket : VIP001  
Total Bayar: Rp 290000.0  
=====  
  
[Info: Objek bertipe PenumpangVIP]  
BUILD SUCCESSFUL (total time: 2 minutes 10 seconds)
```