

LAPORAN PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK
MODUL 3
Latihan Konsep PBO



Oleh :
Riska Vany Dewi
2311103111
S1SI-07-C

PROGRAM STUDI S1 SISTEM INFORMASI
FAKULTAS REKAYASA INDUSTRI
UNIVERSITAS TELKOM PURWOKERTO

2024



I. GUIDED

Kode:

Person.java

```
package latihanp7riskavanydewi;

/**
 * @author RISK A VANY DEWI
 * 2311103111
 * S1SI07C
 */
abstract class Person {
    protected String nama;
    protected String id;

    public Person(String nama, String id){
        this.nama = nama;
        this.id = id;
    }

    public abstract void showInfo();
}
```

Penjelasan:

Kelas Person adalah kelas abstrak yang mendefinisikan entitas dasar dengan atribut nama dan id, serta sebuah konstruktor untuk menginisialisasi kedua atribut tersebut. Kelas ini juga memiliki metode abstrak showInfo(), yang harus diimplementasikan oleh kelas-kelas turunan yang mewarisi Person.

MataKuliah.java

```
package latihanp7riskavanydewi;
```

```
/**
 * @author RISK A VANY DEWI
 * 2311103111
 * S1SI07C
 */

public class MataKuliah {
    private String kode;
    private String namaMatakuliah;
    private int sks;
    private String jadwalHari;
    private String jadwalJam;

    public MataKuliah(String kode, String namaMatakuliah, int sks){
        this.kode = kode;
        this.namaMatakuliah = namaMatakuliah;
        this.sks = sks;
    }

    public void aturJadwal(String hari, String jam){
        this.jadwalHari = hari;
        this.jadwalJam = jam;
    }

    public void aturJadwal(String hari){
        this.jadwalHari = hari;
    }

    public void showInfo(){
```

```
        System.out.println("Kode MK: " +kode+ ", Nama: " +namaMatakuliah+ ", SKS: " +sks);  
    }  
}
```

Penjelasan:

Kelas MataKuliah mendefinisikan struktur untuk mata kuliah di dalam suatu sistem akademik, dengan atribut privat seperti kode, namaMatakuliah, sks, jadwalHari, dan jadwalJam. Konstruktor kelas ini digunakan untuk menginisialisasi kode, namaMatakuliah, dan sks, sementara metode aturJadwal memungkinkan penentuan jadwal kuliah dengan overload untuk menerima satu atau dua parameter (hari dan jam). Metode showInfo digunakan untuk menampilkan informasi terkait mata kuliah, seperti kode, nama, dan jumlah SKS, ke konsol.

KRS.java

```
package latihanp7riskavanydewi;  
  
/**  
 * @author RISKA VANY DEWI  
 * 2311103111  
 * S1SI07C  
 */  
interface KRS {  
    void tambahMataKuliah(MataKuliah mk);  
    void hapusMataKuliah(MataKuliah mk);  
}
```

Penjelasan:

Interface KRS mendefinisikan kontrak untuk pengelolaan mata kuliah, dengan dua metode: tambahMataKuliah(MataKuliah mk) untuk menambahkan mata kuliah dan hapusMataKuliah(MataKuliah mk) untuk menghapus mata kuliah. Kelas yang mengimplementasikan interface ini diharuskan untuk menyediakan logika spesifik untuk kedua operasi tersebut.

Dosen.java

```
package latihanp7riskavanydewi;
```

```
/**
 * @author RISK A VANY DEWI
 * 2311103111
 * S1SI07C
 */

import java.util.ArrayList;
import java.util.List;

class Dosen extends Person {
    private String nip;
    private List<MataKuliah> daftarMatakuliah = new ArrayList<>();

    public Dosen(String nama, String id, String nip){
        super(nama, id);
        this.nip = nip;
    }

    @Override
    public void showInfo(){
        System.out.println("Nama Dosen: " + nama + ", NIP: " + nip);
    }

    public void tambahMataKuliah(MataKuliah mk){
        daftarMatakuliah.add(mk);
    }
}
```

Penjelasan:

Kelas Dosen adalah subclass dari Person yang merepresentasikan seorang dosen dengan atribut tambahan nip dan daftar mata kuliah yang diajarkan (daftarMatakuliah). Konstruktor kelas ini menginisialisasi nama, ID, dan NIP dosen menggunakan konstruktor superclass. Metode showInfo() dioverride untuk menampilkan informasi dosen, sementara metode tambahMataKuliah(MataKuliah mk) digunakan untuk menambahkan mata kuliah ke dalam daftar yang dimiliki oleh dosen.

Mahasiswa.java

```
package latihanp7riskavanydewi;

/**
 * @author RISKA VANY DEWI
 * 2311103111
 * S1SI07C
 */

import java.util.ArrayList;
import java.util.List;

class Mahasiswa extends Person implements KRS {
    private String nim;
    private String prodi;
    private List<MataKuliah> daftarKRS = new ArrayList<>();

    public Mahasiswa(String nama, String id, String nim, String prodi){
        super(nama, id);
        this.nim = nim;
        this.prodi = prodi;
    }
}
```

```
@Override
public void showInfo(){
    System.out.println("Nama Mahasiswa: " +nama+ ", NIM: " +nim+ ", Prodi: " +prodi);
}

@Override
public void tambahMataKuliah(MataKuliah mk){
    daftarKRS.add(mk);
}

@Override
public void hapusMataKuliah(MataKuliah mk) {
    daftarKRS.remove(mk);
}
}
```

Penjelasan:

Kelas Mahasiswa adalah subclass dari Person yang juga mengimplementasikan interface KRS, sehingga berfungsi untuk merepresentasikan seorang mahasiswa. Kelas ini memiliki atribut tambahan nim, prodi, dan daftar mata kuliah yang diambil (daftarKRS). Konstruktor kelas ini menginisialisasi nama, ID, NIM, dan program studi. Metode showInfo() dioverride untuk menampilkan informasi mahasiswa, sementara metode dari interface KRS, yaitu tambahMataKuliah(MataKuliah mk) dan hapusMataKuliah(MataKuliah mk), digunakan untuk menambah dan menghapus mata kuliah dari daftar KRS.

Main.java

```
package latihanp7riskavanydewi;

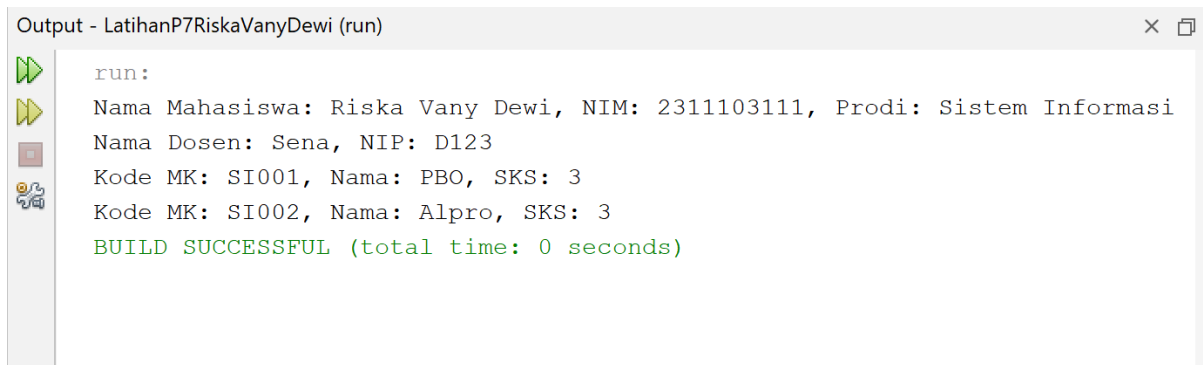
/**
 * @author RISKA VANY DEWI
 * 2311103111
 * S1SI07C
 */
```

```
public class Main {  
    public static void main(String[] args) {  
        Mahasiswa mhs = new Mahasiswa ("Riska Vany Dewi", "123", "2311103111", "Sistem  
Informasi");  
        Dosen dosen = new Dosen ("Sena", "111", "D123");  
  
        MataKuliah mk1 = new MataKuliah ("SI001", "PBO", 3);  
        MataKuliah mk2 = new MataKuliah ("SI002", "Alpro", 3);  
  
        mk1.aturJadwal ("Senin", "08:00");  
        mk2.aturJadwal ("Selasa");  
  
        mhs.tambahMataKuliah(mk1);  
        mhs.tambahMataKuliah(mk2);  
  
        mhs.showInfo();  
        dosen.showInfo();  
        mk1.showInfo();  
        mk2.showInfo();  
    }  
}
```

Penjelasan:

Kelas Main berfungsi sebagai titik masuk untuk menjalankan program. Di dalam metode main, objek Mahasiswa dan Dosen diciptakan dengan memberikan informasi dasar seperti nama, ID, NIM, dan NIP. Dua objek MataKuliah juga dibuat, dan jadwal untuk masing-masing mata kuliah diatur menggunakan metode aturJadwal. Mahasiswa kemudian menambahkan mata kuliah tersebut ke dalam daftar KRS mereka dengan memanggil metode tambahMataKuliah. Terakhir, program menampilkan informasi tentang mahasiswa, dosen, dan mata kuliah yang telah dibuat dengan memanggil metode showInfo() untuk masing-masing objek.

Output:



```
run:
Nama Mahasiswa: Riska Vany Dewi, NIM: 2311103111, Prodi: Sistem Informasi
Nama Dosen: Sena, NIP: D123
Kode MK: SI001, Nama: PBO, SKS: 3
Kode MK: SI002, Nama: Alpro, SKS: 3
BUILD SUCCESSFUL (total time: 0 seconds)
```

II. UNGUIDED

Kembangkan Sistem Informasi Akademik dengan menambahkan fitur berikut:

1. Menampilkan Daftar Mata Kuliah yang Diambil oleh Mahasiswa.
Tambahkan metode **showKRS()** dalam kelas **Mahasiswa** yang menampilkan seluruh mata kuliah yang ada dalam daftar KRS mahasiswa tersebut.
2. Menghitung Total SKS yang Diambil oleh Mahasiswa.
Tambahkan metode **hitungTotalSKS()** dalam kelas **Mahasiswa** untuk menghitung total SKS dari seluruh mata kuliah yang telah diambil.

Kode:

```
MataKuliah.java

package latihanp7riskavanydewi;

/**
 * @author RISKA VANY DEWI
 * 2311103111
 * S1SI07C
 */

public class MataKuliah {
    private String kode;
    private String namaMatakuliah;
    private int sks;
    private String jadwalHari;
    private String jadwalJam;

    public MataKuliah(String kode, String namaMatakuliah, int sks){
        this.kode = kode;
        this.namaMatakuliah = namaMatakuliah;
        this.sks = sks;
    }
}
```

```
public void aturJadwal(String hari, String jam){  
    this.jadwalHari = hari;  
    this.jadwalJam = jam;  
}  
  
public void aturJadwal(String hari){  
    this.jadwalHari = hari;  
}  
  
public void showInfo(){  
    System.out.println("Kode MK: " +kode+ ", Nama: " +namaMatakuliah+ ", SKS: " +sks);  
}  
  
public int getSks(){  
    return sks;  
}  
}
```

Penjelasan:

Modifikasi kelas MataKuliah dengan menambahkan metode getter di kelas MataKuliah untuk mengakses atribut sks yang bersifat privat.

Mahasiswa.java

```
package latihanp7riskavanydewi;  
  
/**  
 * @author RISK A VANY DEWI  
 * 2311103111  
 * S1SI07C
```

```
import java.util.ArrayList;
import java.util.List;

class Mahasiswa extends Person implements KRS {
    private String nim;
    private String prodi;
    private List<MataKuliah> daftarKRS = new ArrayList<>();

    public Mahasiswa(String nama, String id, String nim, String prodi){
        super(nama, id);
        this.nim = nim;
        this.prodi = prodi;
    }

    @Override
    public void showInfo(){
        System.out.println("Nama Mahasiswa: " +nama+ ", NIM: " +nim+ ", Prodi: " +prodi);
    }

    @Override
    public void tambahMataKuliah(MataKuliah mk){
        daftarKRS.add(mk);
    }

    @Override
    public void hapusMataKuliah(MataKuliah mk) {
```

```
        daftarKRS.remove(mk);
    }

    public void showKRS() {
        System.out.println("Daftar Mata Kuliah yang Diambil oleh " + nama + ":");
        for (MataKuliah mk : daftarKRS) {
            mk.showInfo();
        }
    }

    public int hitungTotalSKS() {
        int totalSKS = 0;
        for (MataKuliah mk : daftarKRS) {
            totalSKS += mk.getSks();
        }
        System.out.println("Total SKS: " +totalSKS);
        return totalSKS;
    }
}
```

Penjelasan:

Metode `showKRS()` dalam kelas Mahasiswa berfungsi untuk menampilkan daftar mata kuliah yang diambil oleh mahasiswa, dengan mencetak judul yang mencantumkan nama mahasiswa dan kemudian mengiterasi melalui `daftarKRS` untuk memanggil metode `showInfo()` dari setiap objek `MataKuliah`, sehingga semua detail mata kuliah ditampilkan. Selanjutnya, metode `hitungTotalSKS()` menghitung total SKS dari semua mata kuliah dalam `daftarKRS` dengan menjumlahkan nilai sks setiap objek `MataKuliah`. Setelah menghitung total, metode ini mencetak hasilnya dengan pesan "Total SKS:" untuk memberikan konteks yang jelas sebelum mengembalikan nilai total SKS sebagai integer.

Main.java

```
package latihanp7riskavanydewi;

/**
 * @author RISK A VANY DEWI
 *
 * 2311103111
 * S1SI07C
 */

public class Main {
    public static void main(String[] args) {
        Mahasiswa mhs = new Mahasiswa ("Riska Vany Dewi", "123", "2311103111", "Sistem
Informasi");
        Dosen dosen = new Dosen ("Sena", "111", "D123");

        MataKuliah mk1 = new MataKuliah ("SI001", "PBO", 3);
        MataKuliah mk2 = new MataKuliah ("SI002", "Alpro", 3);

        mk1.aturJadwal ("Senin", "08:00");
        mk2.aturJadwal ("Selasa");

        mhs.tambahMataKuliah(mk1);
        mhs.tambahMataKuliah(mk2);

        mhs.showInfo();
        dosen.showInfo();
        mk1.showInfo();
    }
}
```

```
        mk2.showInfo();  
        mhs.showKRS();  
        mhs.hitungTotalSKS();  
    }  
}
```

Penjelasan:

Pada kelas Main hanya menambahkan metode showKRS() dan hitungTotalSKS() dari kelas Mahasiswa yang sudah dimodifikasi.

Output:



```
run:  
Nama Mahasiswa: Riska Vany Dewi, NIM: 2311103111, Prodi: Sistem Informasi  
Nama Dosen: Sena, NIP: D123  
Kode MK: SI001, Nama: PBO, SKS: 3  
Kode MK: SI002, Nama: Alpro, SKS: 3  
Daftar Mata Kuliah yang Diambil oleh Riska Vany Dewi:  
Kode MK: SI001, Nama: PBO, SKS: 3  
Kode MK: SI002, Nama: Alpro, SKS: 3  
Total SKS: 6  
BUILD SUCCESSFUL (total time: 0 seconds)
```