

LAPORAN PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK
MODUL 3
MATERI



Oleh :

Ardzaricho Revansyah

2311103137

SI07C

PROGRAM STUDI S1 SISTEM INFORMASI
FAKULTAS REKAYASA INDUSTRI
UNIVERSITAS TELKOM PURWOKERTO
2024

LATIHAN KONSEP PBO

A. Studi Kasus

Sistem Informasi Akademik Universitas

B. Deskripsi

Universitas ingin mengembangkan sistem informasi akademik yang memungkinkan pengelolaan data akademik mahasiswa, dosen, matakuliah, dan pengelolaan KRS (Kartu Rencana Studi) mahasiswa. Fitur utama dari sistem ini adalah sebagai berikut:

1. Manajemen Mahasiswa: Setiap mahasiswa memiliki informasi dasar seperti nama, NIM, prodi, dan daftar KRS yang sudah diambil.
2. Manajemen Dosen: Setiap dosen memiliki nama, NIP, dan daftar matakuliah yang diajarkan.
3. Manajemen Matakuliah: Setiap matakuliah memiliki kode, nama, dan jumlah SKS.
4. Pengelolaan KRS (Kartu Rencana Studi): Mahasiswa dapat memilih dan menghapus matakuliah yang ingin mereka ambil setiap semester.

C. Guided

1. Buat project baru dengan LatihanP7**Nama**. Nama diganti dengan nama kalian, contoh : *LatihanP7SenaWijayanto*
2. Buatlah program struktur program dasar dengan kelas-kelas berikut:
 - **Person**: Kelas abstrak yang menjadi *superclass* bagi **Mahasiswa** dan **Dosen**.
 - **Mahasiswa**: Kelas turunan dari **Person** yang memiliki informasi KRS.
 - **Dosen**: Kelas turunan dari **Person** yang mengelola daftar matakuliah yang diajarkan.
 - **Matakuliah**: Kelas yang berisi informasi matakuliah, dengan metode **aturJadwal** yang di-overload.
 - **KRS**: Interface yang mengatur pengambilan dan penghapusan matakuliah oleh mahasiswa.

Kode

Person.java

```
package latihanp7Ardza;  
/**  
 *  
 * @author ARDZARICHO REVANSYAH  
 * 2311103137  
 * 07C
```

```
*/  
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();  
}
```

Person adalah kelas abstrak yang berfungsi sebagai kelas dasar bagi kelas Mahasiswa dan Dosen. Kelas ini memiliki dua atribut, yaitu nama dan id, yang disetel melalui konstruktor ketika objek dibuat. Metode abstrak showInfo() dideklarasikan dalam kelas ini dan wajib diimplementasikan oleh kelas turunan untuk menampilkan informasi spesifik dari setiap objek yang dibuat. Dengan menggunakan kelas abstrak, kelas Person dapat menjadi dasar umum bagi semua objek yang memiliki informasi dasar berupa nama dan identitas.

MataKuliah.java

```
package latihanp7Ardza;  
/**  
 *  
 * @author ARDZARICHO REVANSYAH  
 * 2311103137  
 * 07C  
 */  
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);
    }
}
```

MataKuliah adalah kelas yang merepresentasikan entitas mata kuliah dengan atribut-atribut seperti kode mata kuliah, namaMatakuliah, sks (jumlah Satuan Kredit Semester), jadwalHari, dan jadwalJam. Konstruktor kelas MataKuliah digunakan untuk menginisialisasi atribut kode, namaMatakuliah, dan sks dengan nilai awal saat objek mata kuliah dibuat. Metode aturJadwal disediakan dalam dua versi: satu metode menerima dua parameter yaitu hari dan jam untuk menetapkan jadwal lengkap, sedangkan metode lainnya hanya menerima hari. Dengan menggunakan metode showInfo(), informasi lengkap mengenai mata kuliah dapat ditampilkan, mencakup kode, nama, dan SKS mata kuliah tersebut.

KRS.java

```
package latihanp7Ardza;
/**
 *
 * @author ARDZARICHO REVANSYAH
 * 2311103137
 * 07C
 */
interface KRS {

    void tambahMatakuliah(MataKuliah mk);

    void hapusMatakuliah(MataKuliah mk);

}
```

KRS adalah antarmuka yang menyediakan metode dasar untuk mengelola mata kuliah dalam daftar KRS (Kartu Rencana Studi) mahasiswa. Antarmuka ini mendeklarasikan dua metode, yaitu tambahMatakuliah dan hapusMatakuliah, yang digunakan untuk menambah dan menghapus mata kuliah dalam daftar KRS mahasiswa. Kelas Mahasiswa mengimplementasikan antarmuka ini sehingga setiap mahasiswa dapat mengatur daftar mata kuliah yang mereka ambil. Dengan menerapkan KRS, kelas yang mengimplementasinya akan memiliki kemampuan untuk memanipulasi daftar mata kuliah secara dinamis.

Dosen.java

```
package latihanp7Ardza;

import java.util.ArrayList;
import java.util.List;
/**
 *
 * @author ARDZARICHO REVANSYAH
 * 2311103137
 * 07C
 */
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);
    }

}
```

Dosen adalah kelas turunan dari kelas abstrak Person yang merepresentasikan seorang dosen. Kelas ini memiliki atribut tambahan yaitu nip (Nomor Induk Pegawai) dan daftarMatakuliah, sebuah daftar yang menyimpan mata kuliah yang diajarkan oleh dosen tersebut. Dalam kelas ini, metode showInfo() diimplementasikan untuk menampilkan informasi dosen seperti nama dan NIP. Selain itu, ada metode tambahMatakuliah() yang memungkinkan dosen menambahkan mata kuliah yang diajarkannya ke dalam daftarMatakuliah. Dengan demikian, kelas Dosen dapat memfasilitasi penyimpanan daftar mata kuliah yang diampu oleh dosen terkait.

Mahasiswa.java

```
package latihanp7Ardza;

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

```
/**
 *
 * @author ARDZARICHO REVANSYAH
 * 2311103137
 * 07C
 */
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);
    }
}
```

Mahasiswa adalah kelas yang merepresentasikan entitas mahasiswa, turunan dari kelas abstrak Person, serta mengimplementasikan antarmuka KRS. Kelas ini memiliki atribut tambahan berupa nim (Nomor Induk Mahasiswa) dan prodi (Program Studi), serta daftarKRS, yaitu daftar yang menyimpan mata kuliah yang diambil oleh mahasiswa. Metode showInfo() diimplementasikan untuk menampilkan informasi mahasiswa seperti nama, NIM, dan program studi. Metode tambahMatakuliah() menambahkan mata kuliah ke dalam daftar daftarKRS, sedangkan metode hapusMatakuliah() menghapus mata kuliah dari daftar tersebut. Dengan mengimplementasikan antarmuka KRS, kelas Mahasiswa memiliki fungsi untuk mengelola daftar mata kuliah yang diambil dalam semester tertentu.

Main.java

```
package latihanp7Ardza;
/**
 *
 * @author ARDZARICHO REVANSYAH
 * 2311103137
 * 07C
 */
public class Main {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        Mahasiswa mhs = new Mahasiswa("Revansyah", "123", "2211109001",
"Sistem Informasi");
        Dosen dosen = new Dosen("Ardza Richo", "456", "D001");

        MataKuliah mk1 = new MataKuliah("SI101", "PBO", 3);
        MataKuliah mk2 = new MataKuliah("SI102", "Basis Data", 3);

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

        mhs.tambahMatakuliah(mk1);
        mhs.tambahMatakuliah(mk2);

        mhs.showInfo();
        dosen.showInfo();
        mk1.showInfo();
        mk2.showInfo();
    }
}
```

Main adalah kelas utama dalam program ini yang berfungsi untuk menjalankan aplikasi dengan metode main(). Di dalam metode main(), objek Mahasiswa, Dosen, dan beberapa objek MataKuliah dibuat dan diinisialisasi dengan data yang diberikan. Mata kuliah ditambahkan ke dalam daftar KRS mahasiswa dengan memanggil metode tambahMatakuliah. Setelah objek diinisialisasi dan daftar KRS diperbarui, informasi mengenai mahasiswa, dosen, dan mata kuliah ditampilkan di layar. Kelas ini memberikan gambaran umum tentang cara program beroperasi dan bagaimana setiap entitas berinteraksi satu sama lain dalam sistem akademik ini.

```
Output - LatihanP7Ardza (run) x
run:
Nama Mahasiswa: Budi, NIM: 2211109001, Prodi: Sistem Informasi
Nama Dosen: Sena Wijayanto, NIP: D001
Kode MK: SI101, Nama: PBO, SKS: 3
Kode MK: SI102, Nama: Basis Data, SKS: 3
BUILD SUCCESSFUL (total time: 0 seconds)
```

menampilkan informasi lengkap tentang mahasiswa yang mengambil mata kuliah, dosen yang mengajar, serta detail masing-masing mata kuliah yang telah ditambahkan ke daftar KRS mahasiswa.

D. 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.

Mahasiswa.java

```
package latihanp7Ardza;

import java.util.ArrayList;
import java.util.List;
/**
 *
 * @author ARDZARICHO REVANSYAH
 * 2311103137
 * 07C
 */
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("\n===== Daftar KRS Mahasiswa " + nama + " =====");
    if (daftarKRS.isEmpty()) {
        System.out.println("Tidak ada mata kuliah yang diambil.");
    } else {
        for (MataKuliah mk : daftarKRS) {
            mk.showInfo();
        }
    }
}
}
```

Menambahkan fungsi baru didalam kelas mahasiswa yaitu fungsi showKRS() yang outputnya nanti adalah daftar krs mahasiswa jika mahasiswa tidak mengambil mata kuliah maka outputnya “tidak ada mata kuliah yang diambil”.

Main.java

```
package latihanp7Ardza;

/**
 *
 * @author ARDZARICHO REVANSYAH
 * 2311103137
 * 07C
 */
public class Main {
```

```
/**
 * Main method
 * @param args the command line arguments
 */
public static void main(String[] args) {
    // Membuat objek Mahasiswa dan Dosen
    System.out.println("===== Informasi Mahasiswa =====");
    Mahasiswa mhs = new Mahasiswa("Revansyah", "123", "2211109001",
    "Sistem Informasi");
    Dosen dosen = new Dosen("Ardza Richo", "456", "D001");

    // Membuat objek MataKuliah
    MataKuliah mk1 = new MataKuliah("SI101", "PBO", 3);
    MataKuliah mk2 = new MataKuliah("SI102", "Basis Data", 3);
    MataKuliah mk3 = new MataKuliah("SI103", "Jaringan Komputer", 3);

    // Mengatur jadwal MataKuliah (Overloading)
    mk1.aturJadwal("Senin", "08:00");
    mk2.aturJadwal("Selasa");

    // Menambahkan MataKuliah ke KRS Mahasiswa
    mhs.tambahMatakuliah(mk1);
    mhs.tambahMatakuliah(mk2);
    mhs.tambahMatakuliah(mk3);

    // Menampilkan informasi
    mhs.showInfo();    // Menampilkan informasi Mahasiswa (Polimorfisme)
    dosen.showInfo(); // Menampilkan informasi Dosen (Polimorfisme)
    mhs.showKRS();    // Menampilkan daftar MataKuliah
}
}
```

Menambahkan pemanggilan fungsi showKRS() pada file main sehingga Ketika file dijalankan akan menampilkan daftar mata kuliah yang diambil mahasiswa tadi.

```
Output - LatihanP7Ardza (run) ×
run:
===== Informasi Mahasiswa =====
Nama Mahasiswa: Revansyah, NIM: 2211109001, Prodi: Sistem Informasi
Nama Dosen: Ardza Richo, NIP: D001

===== Daftar KRS Mahasiswa Revansyah =====
Kode MK: SI101, Nama: PBO, SKS: 3
Kode MK: SI102, Nama: Basis Data, SKS: 3
Kode MK: SI103, Nama: Jaringan Komputer, SKS: 3
BUILD SUCCESSFUL (total time: 0 seconds)
```

Jika sebelumnya hanya menampilkan nama mahasiswa dan dosen saja sekarang bisa melihat apa saja yang mahasiswa ambil pada KRS.

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.

Matakuliah.java

```
package latihanp7Ardza;

/**
 *
 * @author ARDZARICHO REVANSYAH
 * 2311103137
 * 07C
 */
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);
}

// Metode untuk mendapatkan jumlah SKS
public int getSks() {
    return sks;
}
}
```

Kelas `MataKuliah` merepresentasikan informasi tentang mata kuliah yang ditawarkan dalam program akademik. Kelas ini memiliki atribut `kode`, `namaMatakuliah`, dan `sks` untuk menyimpan kode mata kuliah, nama mata kuliah, dan jumlah satuan kredit semester (SKS) yang diberikan. Kelas ini juga memiliki dua metode `aturJadwal` yang di-overload, yang memungkinkan pengguna untuk mengatur jadwal mata kuliah dengan menyertakan hari dan jam, atau hanya hari saja. Metode `showInfo()` digunakan untuk menampilkan informasi lengkap tentang mata kuliah, termasuk kode, nama, dan jumlah SKS. Terakhir, terdapat metode `getSks()` yang mengembalikan nilai SKS dari mata kuliah, memungkinkan kelas lain untuk mengakses informasi tersebut.

Mahasiswa.java

```
package latihanp7Ardza;

import java.util.ArrayList;
import java.util.List;
/**
 *
 * @author ARDZARICHO REVANSYAH
 * 2311103137
 * 07C
 */
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);
}

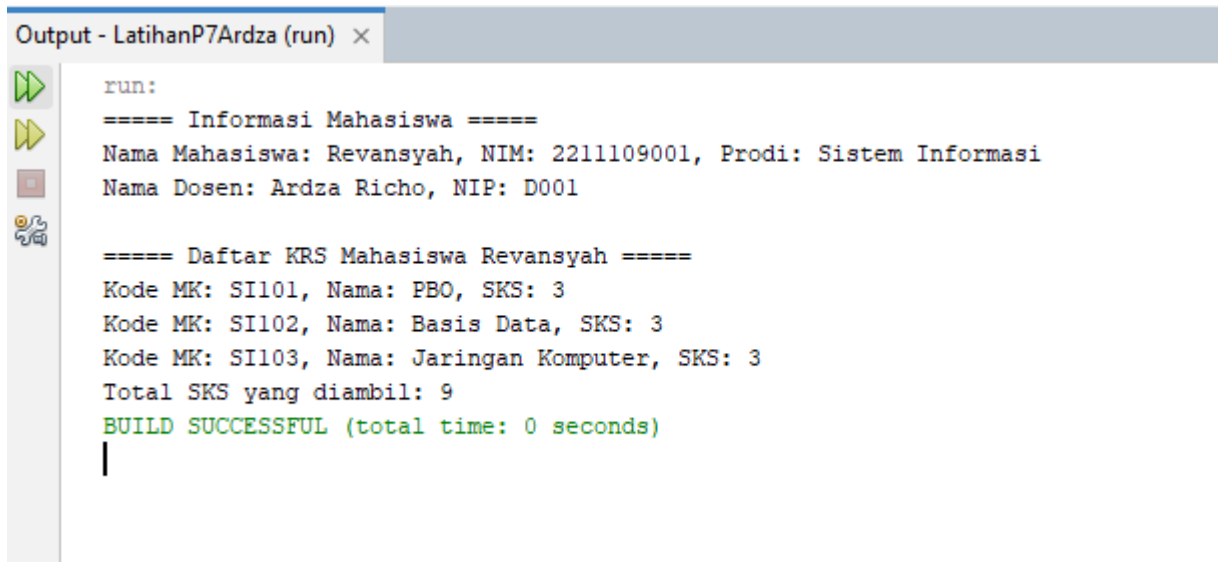
@Override
public void hapusMatakuliah(MataKuliah mk) {
    daftarKRS.remove(mk);
}

public void showKRS() {
    System.out.println("\n===== Daftar KRS Mahasiswa " + nama + " =====");
    if (daftarKRS.isEmpty()) {
        System.out.println("Tidak ada mata kuliah yang diambil.");
    } else {
        for (MataKuliah mk : daftarKRS) {
            mk.showInfo();
        }
        System.out.println("Total SKS yang diambil: " + hitungTotalSKS());
    }
}

// Metode untuk menghitung total SKS
public int hitungTotalSKS() {
    int totalSKS = 0;
    for (MataKuliah mk : daftarKRS) {
        totalSKS += mk.getSks();
    }
    return totalSKS;
}
```

```
}
```

Kelas `Mahasiswa` adalah kelas yang mewarisi dari kelas `Person` dan mengimplementasikan antarmuka `KRS`, sehingga dapat mengelola daftar mata kuliah yang diambil. Kelas ini memiliki atribut `nim` (nomor induk mahasiswa) dan `prodi` (program studi) yang diinisialisasi melalui konstruktor, serta daftar mata kuliah yang diambil yang disimpan dalam `daftarKRS`. Metode `showInfo()` menampilkan informasi mahasiswa, termasuk nama, NIM, dan program studi. Kelas ini juga mengimplementasikan metode dari antarmuka `KRS` untuk menambahkan dan menghapus mata kuliah dari daftar KRS mahasiswa. Metode `showKRS()` mencetak daftar mata kuliah yang diambil, dan jika daftar tersebut kosong, akan menampilkan pesan bahwa tidak ada mata kuliah yang diambil. Terakhir, metode `hitungTotalSKS()` menghitung total SKS dari semua mata kuliah yang telah diambil mahasiswa, dengan menjumlahkan SKS setiap mata kuliah menggunakan metode `getSks()` dari kelas `MataKuliah`.



```
Output - LatihanP7Ardza (run) x
run:
===== Informasi Mahasiswa =====
Nama Mahasiswa: Revansyah, NIM: 2211109001, Prodi: Sistem Informasi
Nama Dosen: Ardza Richo, NIP: D001

===== Daftar KRS Mahasiswa Revansyah =====
Kode MK: SI101, Nama: PBO, SKS: 3
Kode MK: SI102, Nama: Basis Data, SKS: 3
Kode MK: SI103, Nama: Jaringan Komputer, SKS: 3
Total SKS yang diambil: 9
BUILD SUCCESSFUL (total time: 0 seconds)
|
```

Outpunya dapat memberikan informasi lengkap tentang mahasiswa yang mengambil mata kuliah, dosen yang mengajar, serta detail mata kuliah yang telah diambil dan total SKS yang diperoleh mahasiswa.