

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 LatihanP7Nama. 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 LatihanP7Hanafiilham;

/**
 *
 * @author hanaf
 */
abstract class Person {

    protected String nama;
    protected String id;

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

    // Abstract Method (Implementasi khusus di subclass)
    public abstract void showInfo();
}
```

Penjelasan :

Class Person berfungsi untuk dasar class abstrak yang lainnya, class ini memiliki dua atribut utama yaitu : nama dan id, yang menyimpan informasi identitas seseorang.

MataKuliah.java

```
package LatihanP7Hanafiilham;

/**
 *
 * @author hanaf
 */
```

```

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;
    }

    // Overloading: Metode aturJadwal dengan dua versi
    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 :

Pada mata class MataKuliah terdapat lima atribut, yaitu “String kode”, “String namaMatakuliah”, “int SKS”, “String jadwalHari”, “String jadwalJam”. Yang digunakan untuk menyimpan informasi seputar matakuliah.

KRS.java

```
package LatihanP7Hanafiilham;

/**
 *
 * @author hanaf
 */
interface KRS {

    void tambahMatakuliah(MataKuliah mk);

    void hapusMatakuliah(MataKuliah mk);
}
```

Penjelasan :

Interface yang mendefinisikan kontrak untuk mengelola matakuliah, dan mengimplementasikan tambah dan hapus matakuliah.

Dosen.java

```
package LatihanP7Hanafiilham;

/**
 *
 * @author hanaf
 */
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;
    }

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

    // Tambah Matakuliah untuk Dosen
    public void tambahMatakuliah(MataKuliah mk) {
        daftarMatakuliah.add(mk);
    }
}

```

Penjelasan :

Kelas Dosen berfungsi sebagai representasi dari dosen dalam sistem akademik, dengan kemampuan untuk menyimpan informasi dasar (nama, id, dan nip) dan mengelola mata kuliah yang diajarkan. Dengan mengextends Person, Dosen mewarisi perilaku umum yang berlaku untuk semua orang, tetapi juga menambahkan fungsionalitas spesifik yang berkaitan dengan peran dosen.

Mahasiswa.java

```

package LatihanP7Hanafiilham;

/**
 *
 * @author hanaf
 */
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;
}

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

// Implementasi metode interface untuk tambah dan hapus matakuliah
@Override
public void tambahMatakuliah(MataKuliah mk) {
    daftarKRS.add(mk);
}

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

```

Penjelasan :

Class ini digunakan untuk menambah dan menghapus matakuliah, serta metode “ShowInfo” berfungsi untuk memberikan informasi sesuai kebutuhan.

Main.java

```
package LatihanP7Hanafiilham;

/**
 *
 * @author hanaf
 */
public class Main {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // Membuat Objek Mahasiswa dan Dosen
        Mahasiswa mhs = new Mahasiswa("Budi", "123", "2211109001", "Sistem
Informasi");
        Dosen dosen = new Dosen("Sena Wijayanto", "456", "D001");

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

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

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

        // Menampilkan Informasi
        mhs.showInfo(); // Menampilkan informasi Mahasiswa (Polimorfisme)
        dosen.showInfo(); // Menampilkan informasi Dosen (Polimorfisme)
        mk1.showInfo(); // Informasi Matakuliah 1
```

```
mk2.showInfo(); // Informasi Matakuliah 2
    }
}
```

Penjelasan :

File utama untuk menjalankan program,

Output

```
.
--- exec:3.1.0:exec (default-cli) @ Pbo_inheritance2 ---
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 SUCCESS
-----
Total time: 4.945 s
Finished at: 2024-11-03T17:08:08+07:00
```

Penjelasan :

Jika program berjalan, output program ini menampilkan pesan berupa, informasi seorang mahasiswa, dosen, dan dua mata kuliah yang diambil oleh mahasiswa tersebut.

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.

Code :

```
public void showKRS() {
    System.out.println("Daftar Mata Kuliah yang Diambil oleh " + nama +
        ":");
    for (MataKuliah mk : daftarKRS) {
        mk.showInfo(); // Menampilkan informasi setiap mata kuliah
    }
}
```


Untuk menampilkan daftar matakuliah, tambahkan metode showKRS(), pada class Mahasiswa.java, serta menggunakan metode looping untuk menampilkan seluruh daftar matakuliah.

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.

Code :

```
public int hitungTotalSKS() {  
    int totalSKS = 0;  
    for (MataKuliah mk : daftarKRS) {  
        totalSKS += mk.getSks(); // Menjumlahkan SKS dari setiap mata  
        kuliah  
    }  
    return totalSKS;  
}
```

Untuk menghitung total sks, tambahkan metode hitungTotalSKS(), pada class Mahasiswa.java, serta menggunakan metode looping yang didalamnya terdapat penjumlahan untuk menjumlahkan setiap sks.

Code :

```
// Menambahkan getter  
public int getSks() {  
    return sks;  
}
```

Menambahkan getter pada class Matakuliah.java, untuk memberikan akses pada atribut sks yang privat.

Code :

```
mhs.showKRS();  
  
// Menghitung dan Menampilkan Total SKS  
int totalSKS = mhs.hitungTotalSKS();  
System.out.println("Total SKS yang Diambil: " + totalSKS);
```

Menambahkan code baru pada class Main.java, untuk menampilkan daftar matakuliah yang diambil, serta menghitung dan menampilkan total sks.

Output

```
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
Daftar Mata Kuliah yang Diambil oleh Budi:
Kode MK: SI101, Nama: PBO, SKS: 3
Kode MK: SI102, Nama: Basis Data, SKS: 3
Total SKS yang Diambil: 6
```

```
-----
BUILD SUCCESS
-----
```

```
Total time: 4.967 s
Finished at: 2024-11-03T17:51:27+07:00
-----
```

Pada code yang telah dimodifikasi, terdapat informasi tambahan yaitu : informasi mata kuliah yang diambil oleh mahasiswa dan total sks.