

LAPORAN PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK
MODUL 3



Oleh :

Nico Salomo Putra Lumban Gaol

2311103127

S1SI-07-C

PROGRAM STUDI S1 SISTEM INFORMASI
FAKULTAS REKAYASA INDUSTRI
UNIVERSITAS TELKOM PURWOKER

I. 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:
 - a. **Person**: Kelas abstrak yang menjadi *superclass* bagi **Mahasiswa** dan **Dosen**.
 - b. **Mahasiswa**: Kelas turunan dari **Person** yang memiliki informasi KRS.
 - c. **Dosen**: Kelas turunan dari **Person** yang mengelola daftar matakuliah yang diajarkan.
 - d. **Matakuliah**: Kelas yang berisi informasi matakuliah, dengan metode **aturJadwal** yang di-overload.
 - e. **KRS**: Interface yang mengatur pengambilan dan penghapusan matakuliah oleh mahasiswa.

Jawab:

a) **Person**:

```
package latihanp7nico;
/**
 *
 * @author LENOVO
 */
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: dalam class person merupakan bagian dari package LatihanP7nico. dan disini terdapat class abstract yang dimana artinya kelas abstract tidak bisa dibuat objeknya secara langsung. dan dikelas abstract terdapat 2 atribut yaitu nama dan id dengan tipe data string. dan disini terdapat constructor yaitu pada bagian public person yang dimana hanya dapat menampung 2 tipe data yaitu string nama dan string id, dan menggunakan this untuk menunjuk ke atribut tersebut. dan terakhir pada public abstract void showInfo bertujuan untuk menampilkan informasi yang sesuai dengan implementasi di class turunannya.

b) **MataKuliah**

```
c) package latihanp7nico;
d)
e) /**
f)  *
g)  * @author LENOVO
```

```

*/
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 class matakuliah ini menggunakan private yang dimana maksudnya adalah atribut tersebut hanya bisa diakses oleh class itu sendiri .dan pada class mata kuliah ini terdapat 5 atribut yaitu String kode yang dimana fungsi nya untuk menyimpan kode matakuliah,String matakuliah berfungsi untuk menyimpan nama matakuliah,String sks untuk menyimpan jumlah sks,String jadwalhari untuk menyimpan hari jadwalkuliah,String jadwaljam untuk menyimpan jadwal jam kuliah.dan pada public matakuliah hanya menerima 3 atribut yaitu kode,nama matakuliah dan sks.

c) **KRS**

```

package latihanp7nico;
interface KRS {

    void tambahMatakuliah(MataKuliah mk);

    void hapusMatakuliah(MataKuliah mk);

}

```

Penjelasan: Dalam class krs tersebut terdapat interface yang dimana fungsi nya untuk mengumpulkan method tanpa implementasi.dan pada class krs ini terdapat void tambah matakuliah yang bertujuan menambahkan mata kuliah.dan ada juga void hapus mata kuliah yang bertujuan menghapus mata kuliah.

d) **Dosen**

```

package latihanp7nico;

```

```

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: Pada kelas dosen terdapat import yang bertujuan untuk menyimpan daftar mata kuliah dalam bentuk list. Constructor class Dosen menerima tiga atribut (nama, id, dan nip). dan fungsi nip untuk menyimpan nomor induk pegawai dosen. dan class dosen ini merupakan turunan dari kelas person. dan pada list<matakuliah> berfungsi untuk menyimpan mata kuliah yang diajar oleh dosen tersebut.

e) Mahasiswa

```

package latihanp7nico;
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: Kelas Mahasiswa adalah subclass dari Person dan menggunakan interface KRS. Kelas ini juga memiliki atribut nim, prodi, dan daftar KRS untuk menyimpan daftar mata kuliah yang telah diambil. Untuk menampilkan informasi khusus siswa, kelas ini mengoverride method showInfo() dan menyediakan implementasi untuk method tambahMatakuliah() dan hapusMatakuliah() sesuai dengan persyaratan KRS.

f) Main

```

public class main {

    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)
    }
}

```

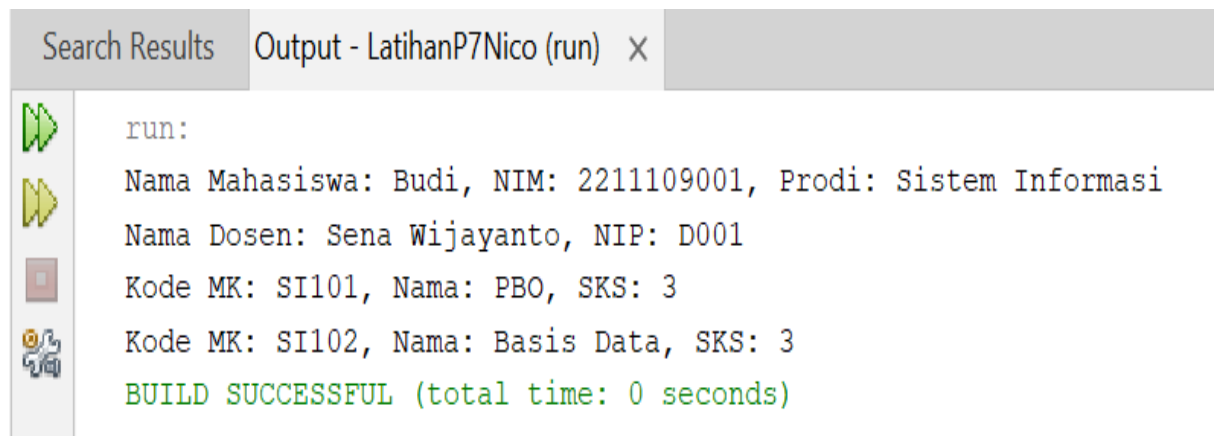
```

        dosen.showInfo(); // Menampilkan informasi Dosen
        (Polimorfisme)
        mk1.showInfo(); // Informasi Matakuliah 1
        mk2.showInfo(); // Informasi Matakuliah 2
    }
}

```

Penjelasan: Kode program ini dapat membuat objek mahasiswa, dosen, dan matakuliah, menambahkan mata kuliah ke KRS mahasiswa, dan menggunakan metode showInfo() untuk menampilkan informasi masing-masing.

Hasil Program:



```

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)

```

Penjelasan : Dari gambar diatas akan muncul nama mahasiswa,nama dosen,kode mata kuliah yang sesuai dengan kode program pada main.

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.

Jawab:

```

public void showKrs() {
    System.out.println("Daftar mata kuliah yang diambil oleh " + Nama()
        + ":");

    for(MataKuliah mk : daftarKrs) {
        mk.showInfo();
    }
}

```

Penjelasan : Pada program diatas menambahkan showKrs untuk menampilkan daftar mata kuliah dan pada program ini menambah metode looping agar menampilkan daftar mata kuliah.

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.

Jawab:

```

public int hitungTotalSKS() {
    int totalSKS = 0;
}

```

```

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

    public int getSks() {
        return sks;
    }
}

```

Penjelasan : Pada program diatas menambah metode hitung totalsks dan menggunakan looping untuk menampilkan total sks

```

        mhs.showKRS();
        int totalSKS = mhs.hitungTotalSKS();
        System.out.println("Total SKS yang Diambil: " + totalSKS);
    }
}

```

Penjelasan: Pada program diatas terletak pada class main.java. yang dimana fungsinya untuk menampilkan seluruh daftar matakuliah dan menghitung total sks.