

CONTOH KASUS: AGREGASI

Disusun Oleh:
Reza Budiawan

Untuk:
Tim Dosen Algoritma & Pemrograman Lanjut

Hanya dipergunakan untuk kepentingan pengajaran di lingkungan Fakultas Ilmu Terapan, Universitas Telkom

Kasus 1

- Buatlah sebuah program sederhana untuk mendata mahasiswa dengan mata kuliah yang diambilnya.
- Note:
 - Class mata kuliah memiliki atribut kode, nama, sks
 - Class MataKuliah merupakan atribut bagi class Mahasiswa
 - Class Mahasiswa memiliki atribut nama dan nim selain atribut mata kuliah
 - Satu objek mahasiswa memiliki banyak objek mata kuliah.
 - Objek mata kuliah yang diambil mahasiswa bersifat tetap (tidak dapat di-drop)

Kasus 1

- Pada class Main, buatlah objek dengan data berikut:

NIM	Nama	Kode	Mata Kuliah	SKS
6701160001	Rayizan	MI1274	Alpro Lanjut	4
6701160001	Rayizan	MI1264	Dasar Alpro	4
6701160001	Rayizan	MI2294	Pemr. Visual	3
6701160002	Alfizar	MI1274	Alpro Lanjut	4
6701160002	Alfizar	MI2294	Pemr. Visual	3

Penyelesaian Kasus 1

- Langkah 1: Buat class MataKuliah
- Langkah 2: Buat class Mahasiswa
- Langkah 3: Buat Class Main
 - 3-1: Bentuk objek dari tiap class
 - 3-2: Hubungkan objek class yang satu dengan lainnya
 - 3-3: Tampilkan

PENYELESAIAN (KODE)

```
public class MataKuliah {
    private String kode, nama;
    private int sks;

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

    public String getKode() {
        return kode;
    }

    public String getNama() {
        return nama;
    }

    public void setNama(String nama) {
        this.nama = nama;
    }

    public int getSks() {
        return sks;
    }

    public void setSks(int sks) {
        this.sks = sks;
    }
}
```

Langkah 1

```
import java.util.ArrayList;

public class Mahasiswa {
    private String nama, nim;
    private final ArrayList<MataKuliah> matkul = new ArrayList<>();

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

    public String getNama() {
        return nama;
    }

    public String getNim() {
        return nim;
    }

    public void setName(String nama) {
        this.nama = nama;
    }

    public void addMataKuliah(MataKuliah m){
        matkul.add(m);
    }

    public ArrayList<MataKuliah> getMatkul() {
        return matkul;
    }
}
```

Langkah 2

```

import java.util.ArrayList;

public class Main {
    public static void main(String[] args) {
        //bentuk 3 objek matkul
        MataKuliah mk1 = new MataKuliah("MI1274", "Alpro Lanjut", 4);
        MataKuliah mk2 = new MataKuliah("MI1264", "Dasar Alpro", 4);
        MataKuliah mk3 = new MataKuliah("MI2294", "Pemr. Visual", 3);

        //bentuk objek mahasiswa
        Mahasiswa mhs1 = new Mahasiswa("6701160001", "Rayizan");
        Mahasiswa mhs2 = new Mahasiswa("6701160002", "Alfizar");

        //hubungkan keduanya
        mhs1.addMataKuliah(mk1);
        mhs1.addMataKuliah(mk2);
        mhs1.addMataKuliah(mk3);

        mhs2.addMataKuliah(mk1);
        mhs2.addMataKuliah(mk3);

        //tampilkan data
        System.out.println("Mahasiswa 1");
        System.out.println("Data: "+mhs1.getNim()+"--"+mhs1.getNama());
        System.out.println("Mengambil mata kuliah: ");
        ArrayList<MataKuliah> al = mhs1.getMatkul();
        for(MataKuliah x:al){
            System.out.println(" -"+x.getKode()+" # "+x.getNama()+" # "+x.getSks());
        }

        System.out.println("\nMahasiswa 2");
        System.out.println("Data: "+mhs2.getNim()+"--"+mhs2.getNama());
        System.out.println("Mengambil mata kuliah: ");
        al = mhs2.getMatkul();
        for(MataKuliah x:al){
            System.out.println(" -"+x.getKode()+" # "+x.getNama()+" # "+x.getSks());
        }
    }
}

```

Langkah 3


```
run:
Mahasiswa 1
Data: Rayizan--6701160001
Mengambil mata kuliah:
-MI1274 # Alpro Lanjut # 4
-MI1264 # Dasar Alpro # 4
-MI2294 # Pemr. Visual # 3

Mahasiswa 2
Data: Alfizar--6701160002
Mengambil mata kuliah:
-MI1274 # Alpro Lanjut # 4
-MI2294 # Pemr. Visual # 3
BUILD SUCCESSFUL (total time: 0 seconds)
```

Hasil



Modifikasi

- Ubahlah program di atas sehingga data dapat dimasukkan oleh pengguna.
- Pertahankan bentuk agregasi dari hubungan antar class di atas.