

**LAPORAN PRAKTIKUM ALGORITMA DAN STRUKTUR DATA**  
**JOBSHEET 2**



**Adi Luhung**  
**244107020088**

**Program Studi Teknik Informatika**  
**Jurusan Teknologi Informasi**  
**Politeknik Negeri Malang**  
**2024**

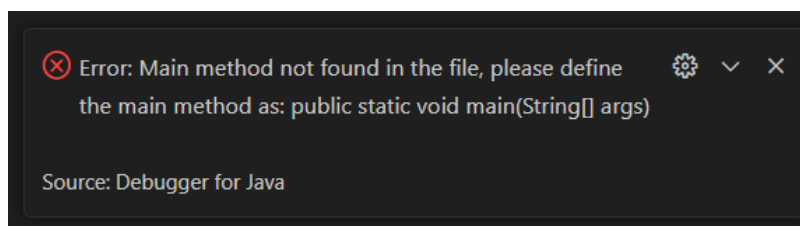
## 1. Percobaan 1

### 1.1 Langkah-langkah

#### 1. Kode program untuk menampilkan data mahasiswa

```
public class Mahasiswa {  
    String nama;  
    String nim;  
    String kelas;  
    double ipk;  
    void tampilakInformasi() {  
        System.out.println("Nama: " + nama);  
        System.out.println("Nim: " + nim);  
        System.out.println("IPK: " + ipk);  
        System.out.println("Kelas: " + kelas);  
    }  
    void ubahKelas(String kelasBaru) {  
        kelas = kelasBaru;  
    }  
    void updateIPK(double ipkBaru) {  
        ipk = ipkBaru;  
    }  
    String nilaiKinerja() {  
        if (ipk >= 3.5) {  
            return "Kinerja sangat baik";  
        } else if (ipk >= 3.0) {  
            return "Kinerja baik";  
        } else if (ipk >= 2.0) {  
            return "Kinerja cukup";  
        } else {  
            return "Kinerja kurang";  
        }  
    }  
}
```

#### 2. Hasil run kode program



### 1.2 Pertanyaan

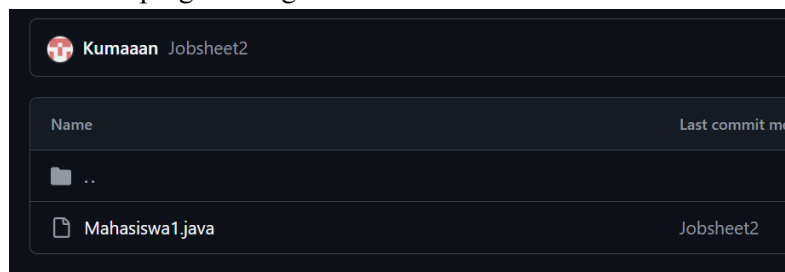
1. Karakteristik class atau objek:
  - Atribut adalah variabel yang digunakan untuk menyimpan suatu nilai
  - Method adalah fungsi yang mendefinisikan perilaku dari objek, biasanya dituliskan menggunakan kata kerja
2. Didalam class mahasiswa terdapat 4 atribut atau variabel, yang diantaranya adalah nama, nim, ipk, kelas
3. Ada 4 method didalam class mahasiswa yaitu, tampilkanInformasi(), ubahKelas(), updateIpk(), nilaiKerja().

4. Memodifikasi kode program agar dapat melakukan validasi pada fungsi updateIpk()

```
void updateIPK(double ipkBaru) {  
    if (ipkBaru > 0.0 && ipkBaru < 4.0) {  
        ipk = ipkBaru;  
    } else {  
        System.out.println("IPK tidak valid. Harus antara 0.0 sampai 4.0");  
    }  
}
```

5. Cara kerja method nilaiKerja() adalah dengan menggolongkan nilai ipk setiap mahasiswa dengan kriteria ipk  $\geq 3.5$ , ipk  $\geq 3.0$ , ipk  $\geq 2.0$ . Kemudian return dari method nilaiKinerja() adalah “Kinerja sangat baik”, “Kinerja baik”, Kinerja cukup”, dan “Kinerja kurang”.

6. Push kode program ke github



2. Percobaan 2

2.1 Langkah-langkah

1. Kode program untuk menjalankan fungsi yang sudah dibuat di percobaan 1

```
public class MahasiswaMain1 {  
    public static void main(String[] args) {  
        Mahasiswa1 mhs1 = new Mahasiswa1();  
        mhs1.nama = "Muhammad Ali Farhan";  
        mhs1.nim = "2241720171";  
        mhs1.kelas = "SI 2J";  
        mhs1.ipk = 3.55;  
  
        mhs1.tampilkanInformasi();  
        mhs1.ubahKelas("SI 2K");  
        mhs1.updateIpk(3.60);  
        mhs1.tampilkanInformasi();  
    }  
}
```

2. Hasil run kode program diatas dan push ke github

```
Nama: Muhammad Ali Farhan
Nim: 2241720171
IPK: 3.55
Kelas: SI 2J
Nama: Muhammad Ali Farhan
Nim: 2241720171
IPK: 3.6
Kelas: SI 2K
```

Kumaaan Jobsheet2	
Name	Last commit message
..	
Mahasiswa1.java	Jobsheet2
MahasiswaMain1.java	Jobsheet2

## 2.2 Pertanyaan

1. Kode program yang menunjukkan instansiasi pada class MahasiswaMain adalah kata new. Kemudian object yang dihasilkan adalah mhs1

```
Mahasiswa1 mhs1 = new Mahasiswa1();
```

2. Untuk mengakses atribut dan method pada suatu object adalah dengan menggunakan tanda titik (.) setelah nama objek. Contohnya:

- Mengakses atribut

```
mhs1.nama = "Muhammad Ali Farhan";
```

- Mengakses method

```
mhs1.tampilkanInformasi();
```

### 3. Percobaan 3

#### 3.1 Langkah-langkah

##### 1. Memodifikasi class Mahasiswa dengan menambahkan dua konstruktor

```
public class Mahasiswa1 {
    String nama;
    String nim;
    String kelas;
    double ipk;
    void tampilkanInformasi() {
        System.out.println("Nama: " + nama);
        System.out.println("Nim: " + nim);
        System.out.println("IPK: " + ipk);
        System.out.println("Kelas: " + kelas);
    }
    void ubahKelas(String kelasBaru) {
        kelas = kelasBaru;
    }
    void updateIpk(double ipkBaru) {
        if (ipkBaru >= 0.0 && ipkBaru <= 4.0) {
            ipk = ipkBaru;
        } else {
            System.out.println("IPK tidak valid.Harus antara 0.0 sampai 4.0");
        }
    }
    String nilaiKinerja() {
        if (ipk >= 3.5) {
            return "Kinerja sangat baik";
        } else if (ipk >= 3.0) {
            return "Kinerja baik";
        } else if (ipk >= 2.0) {
            return "Kinerja cukup";
        } else {
            return "Kinerja kurang";
        }
    }
    public Mahasiswa1() {
    }
    public Mahasiswa1(String nm, String nim, double ipk, String kls) {
        nama = nm;
        this.nim = nim;
        this.ipk = ipk;
        kelas = kls;
    }
}
```

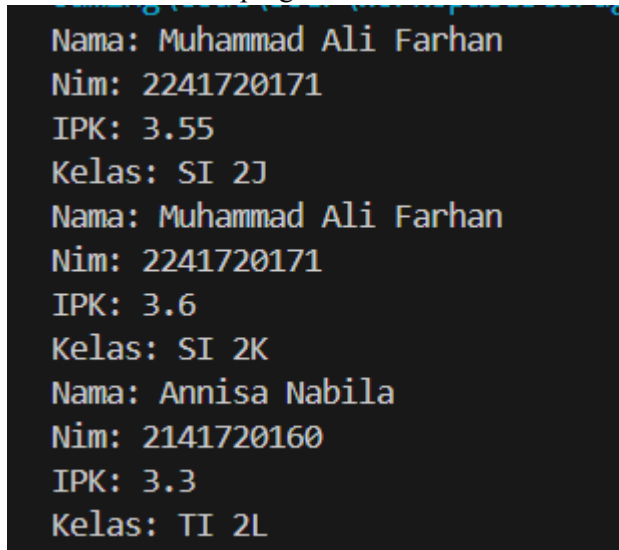
2. Membuat object baru di class MahasiswaMain dengan nama mhs2

```
public class MahasiswaMain1 {
    public static void main(String[] args) {
        Mahasiswa1 mhs1 = new Mahasiswa1();
        mhs1.nama = "Muhammad Ali Farhan";
        mhs1.nim = "2241720171";
        mhs1.kelas = "SI 2J";
        mhs1.ipk = 3.55;

        mhs1.tampilkanInformasi();
        mhs1.ubahKelas("SI 2K");
        mhs1.updateIpk(3.60);
        mhs1.tampilkanInformasi();

        Mahasiswa1 mhs2 = new Mahasiswa1 ("Annisa Nabila",
"2141720160", 3.25, "TI 2L");
        mhs2.updateIpk(3.30);
        mhs2.tampilkanInformasi();
    }
}
```

3. Hasil run dari kode program



```
Nama: Muhammad Ali Farhan
Nim: 2241720171
IPK: 3.55
Kelas: SI 2J
Nama: Muhammad Ali Farhan
Nim: 2241720171
IPK: 3.6
Kelas: SI 2K
Nama: Annisa Nabila
Nim: 2141720160
IPK: 3.3
Kelas: TI 2L
```

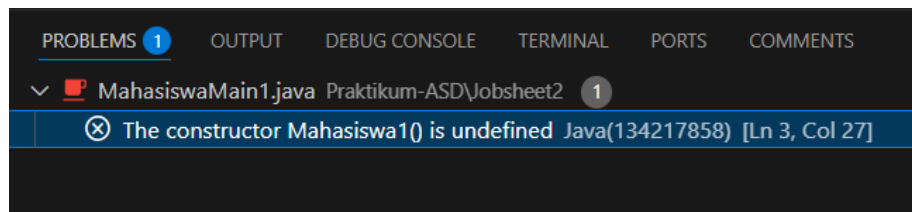
### 3.2 Pertanyaan

1. Deklarasi konstruktor berparameter

```
public Mahasiswa1(String nm, String nim, double ipk, String kls) {
    nama = nm;
    this.nim = nim;
    this.ipk = ipk;
    kelas = kls;
}
```

2. Yang dilakukan pada baris tersebut adalah instansiasi object mhs2 dan menginisialisasi atribut yang ada dalam konstruktor berparameter di class Mahasiswa

3. Menghapus konstruktor default



Jika konstruktor default dihapus dari class Mahasiswa1, maka kode yang mengandalkan konstruktor default akan menghasilkan error.

4. Method akan dipanggil secara berurutan, Karena terdapat method untuk update data. Sehingga jika method tampilkanInformasi() dipanggil sebelum method updateIpk(), maka data yang ditampilkan adalah data yang lama, bukan data sesudah di update.

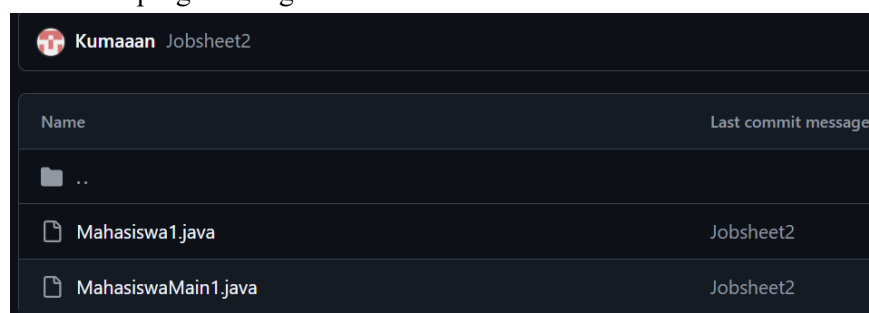
5. Menambahkan object baru dengan nama mhsAdiLuhung

```
Mahasiswa1 mhsAdiLuhung = new Mahasiswa1 ("Adi Luhung",  
"244107020088", 3.80, "TI 1E");  
mhsAdiLuhung.tampilkanInformasi();
```

6. Hasil run kode program

```
Nama: Muhammad Ali Farhan  
Nim: 2241720171  
IPK: 3.55  
Kelas: SI 2J  
Nama: Muhammad Ali Farhan  
Nim: 2241720171  
IPK: 3.6  
Kelas: SI 2K  
Nama: Annisa Nabila  
Nim: 2141720160  
IPK: 3.3  
Kelas: TI 2L  
Nama: Adi Luhung  
Nim: 244107020088  
IPK: 3.8  
Kelas: TI 1E
```

7. Push kode program ke github



## 4. Latihan Praktikum

### 4.1 Latihan 1

#### 1. Kode program untuk mengimplementasikan class MataKuliah

- Class MataKuliah1

```
public class MataKuliah1 {
    String kodeMK;
    String nama;
    int sks;
    int jumlahJam;

    void tampilInformasi() {
        System.out.println("Kode Mata Kuliah: " + kodeMK);
        System.out.println("Nama Mata Kuliah: " + nama);
        System.out.println("SKS: " + sks);
        System.out.println("Jumlah Jam: " + jumlahJam);
    }
    void ubahSks(int sksBaru) {
        sks = sksBaru;
    }
    void tambahJam(int jam) {
        jumlahJam += jam;
        System.out.println("Jumlah jam berhasil ditambahkan. selama " + jam +
" jam.");
    }
    void kurangiJam(int jam) {
        if (jumlahJam <= jam) {
            System.out.println("Pengurangan jam tidak dapat dilakukan!
Pengurang terlalu besar.");
        } else if (jam < 0){
            System.out.println("Jumlah jam tidak boleh negatif.");
        } else {
            jumlahJam -= jam;
            System.out.println("Jumlah jam berhasil dikurangi. selama " + jam
+ " jam.");
        }
    }
    public MataKuliah1() {

    }
    public MataKuliah1(String kodeMK, String nama, int sks, int jumlahJam) {
        this.kodeMK = kodeMK;
        this.nama = nama;
        this.sks = sks;
        this.jumlahJam = jumlahJam;
    }
}
```



- Class MataKuliahMain1

```
public class MataKuliahMain1 {
    public static void main(String[] args) {
        MataKuliah1 mk1 = new MataKuliah1();
        mk1.nama = "Pancasila";
        mk1.kodeMK = "001";
        mk1.sks = 3;
        mk1.jumlahJam = 6;

        mk1.tampilInformasi();
        mk1.ubahSks(2);
        mk1.kurangiJam(2);
        mk1.tampilInformasi();





        MataKuliah1 mk2 = new MataKuliah1("Matematika Dasar", "002", 2, 4);
        mk2.tampilInformasi();
        mk2.ubahSks(3);
        mk2.tambahJam(2);
        mk2.tampilInformasi();
    }
}
```

## 2. Hasil run kode program

```
Kode Mata Kuliah: 001
Nama Mata Kuliah: Pancasila
SKS: 3
Jumlah Jam: 6
Jumlah jam berhasil dikurangi. selama 2 jam.
Kode Mata Kuliah: 001
Nama Mata Kuliah: Pancasila
SKS: 2
Jumlah Jam: 4

Kode Mata Kuliah: Matematika Dasar
Nama Mata Kuliah: 002
SKS: 2
Jumlah Jam: 4
Jumlah jam berhasil ditambahkan. selama 2 jam.
Kode Mata Kuliah: Matematika Dasar
Nama Mata Kuliah: 002
SKS: 3
Jumlah Jam: 6
```

## 3. Push kode program ke github

 Mahasiswa1.java	Jobsheet2
 MahasiswaMain1.java	Jobsheet2
 MataKuliah1.java	Jobsheet2
 MataKuliahMain1.java	Jobsheet2

## 4.2 Latihan 2

### 1. Kode program untuk mengelola data dosen

- Class Dosen1

```
public class Dosen1 {
    String idDosen;
    String nama;
    boolean statusAktif;
    int tahunBergabung;
    String bidangKeahlian;

    void tampilInformasi(int thnSkrng) {
        System.out.println("ID Dosen: "+ idDosen);
        System.out.println("Nama: "+ nama);
        System.out.println("Status Aktif: "+ statusAktif);
        System.out.println("Masa Kerja: "+ hitungMasaKerja(thnSkrng));
        System.out.println("Bidang Keahlian: "+ bidangKeahlian);
    }
    void setStatusAktif(boolean status) {
        if (status == true) {
            statusAktif = true;
        } else {
            statusAktif = false;
        }
    }
    int hitungMasaKerja(int thnSkrng) {
        int masaKerja = thnSkrng - tahunBergabung;
        return masaKerja;
    }
    void ubahKeahlian(String bidang) {
        bidangKeahlian = bidang;
    }
    public Dosen1 () {

    }
    public Dosen1 (String idDosen, String nama, boolean statusAktif, int
tahunBergabung, String bidangKeahlian) {
        this.idDosen = idDosen;
        this.nama = nama;
        this.statusAktif = statusAktif;
        this.tahunBergabung = tahunBergabung;
        this.bidangKeahlian = bidangKeahlian;
    }
}
```

- Class DosenMain1

```
public class DosenMain1 {
    public static void main(String[] args) {
        Dosen1 ds1 = new Dosen1();
        ds1.idDosen = "021";
        ds1.nama = "Rudi";
        ds1.statusAktif = true;
        ds1.tahunBergabung = 2010;
        ds1.bidangKeahlian = "Matematika";

        ds1.tampilInformasi(2025);
        ds1.setStatusAktif(false);
        ds1.ubahKeahlian("Fisika");
        ds1.tampilInformasi(2027);
        System.out.println("");

        Dosen1 ds2 = new Dosen1("010", "Andi", true, 2000, "Bahasa Inggris");
        ds2.tampilInformasi(2030);
        ds2.setStatusAktif(false);
        ds2.ubahKeahlian("Bahasa Jepang");
        ds2.tampilInformasi(2032);
    }
}
```

## 2. Hasil running kode program

```
ID Dosen: 021
Nama: Rudi
Status Aktif: true
Masa Kerja: 15
Bidang Keahlian: Matematika
ID Dosen: 021
Nama: Rudi
Status Aktif: false
Masa Kerja: 17
Bidang Keahlian: Fisika
```

```
ID Dosen: 010
Nama: Andi
Status Aktif: true
Masa Kerja: 30
Bidang Keahlian: Bahasa Inggris
ID Dosen: 010
Nama: Andi
Status Aktif: false
Masa Kerja: 32
Bidang Keahlian: Bahasa Jepang
PS C:\Users\luhun\Kuliah1 Semester2>
```

## 3. Push kode program ke github

Kumaaan Jobsheet2	
Name	Last commit
..	
Dosen1.java	Jobsheet2
DosenMain1.java	Jobsheet2