

**LAPORAN PRAKTIKUM**  
**JOBSHEET 2**



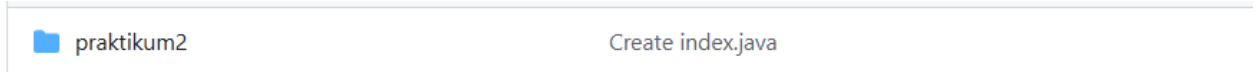
**VANESA MARDIANA PUTRI**  
**244107020129 / 23**  
**KELAS TI 1B**

**PRODI D-IV TEKNIK INFORMATIKA**  
**JURUSAN TEKNOLOGI INFORMASI**  
**POLITEKNIK NEGERI MALANG**

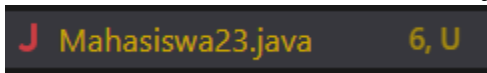
## 2.1 Latihan 1

Langkah – Langkah percobaan :

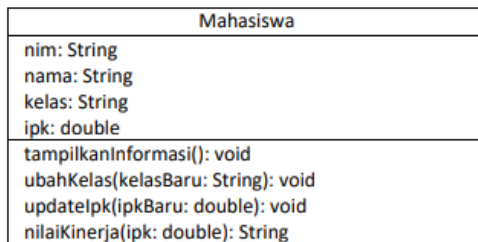
1. Buat folder baru bernama Jobsheet2 di dalam repository Praktikum ASD.



2. Buat file baru, beri nama Mahasiswa.java.

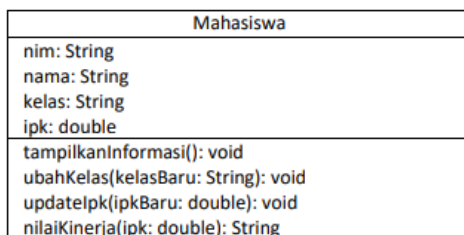


3. Lengkapi class Mahasiswa dengan atribut yang telah digambarkan di dalam class diagram.



```
String nim;  
String nama;  
String kelas;  
double ipk;
```

4. Lengkapi class Mahasiswa dengan method yang telah digambarkan di dalam class diagram tersebut.



```
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) {  
    ipk = ipkBaru;  
}  
  
String nilaiKinerja() {  
    if (ipk >= 3.5) {  
        return "Kinerja Sangat Baik";  
    } else if (ipk >= 3.0) {  
        return "Kinerja Baik";  
    } else if (ipk >= 2.5) {  
        return "Kinerja Cukup";  
    } else {  
        return "Kinerja Kurang";  
    }  
}
```

5. Compile dan run program.

```
[Running] cd "d:\kuliah\smt 2\alsd\praktikum-alsd-1\praktikum2\" && javac Mahasiswa23.java && j  
Error: Could not find or load main class Mahasiswa23
```

Jawaban pertanyaan :

1. Adanya method dan atribut.
2. Memiliki 4 atribut, yaitu nama (String), nim (String), kelas (String), dan ipk (double).
3. Memiliki 4 method, yaitu tampilkanInformasi(): void ubahKelas(kelasBaru: String): void updateIpk(ipkBaru: double): void nilaiKinerja(ipk: double): String.
4. Modifikasi method updateIpk() untuk memvalidasi nilai.

```
void updateIpk(double ipkBaru) {  
    if (ipkBaru < 0.0 || ipkBaru > 4.0){  
        System.out.println(x:"IPK tidak valid. Harus antara 0.0 dan 4.0");  
    }  
    else{  
        ipk = ipkBaru;  
    }  
}
```

5. Method nilaiKerja(); mengevaluasi kerja mahasiswa berdasarkan nilai IPK, dengan membandingkan IPK terhadap beberapa kriteria yang telah ditentukan menggunakan struktur if-else  
Method nilaiKerja(); mengembalikan String yang berisi evaluasi kerja mahasiswa sesuai dengan IPK yang dimilikinya

## 2.2 Latihan 2

Langkah – Langkah percobaan :

1. Buat file baru, beri nama MahasiswaMain.java

```
J MahasiswaMain23.java 2, U
```

2. Tuliskan struktur dasar bahasa pemrograman Java yang terdiri dari fungsi main().

```
public static void main(String[] args) {
```

3. Di dalam fungsi main(), lakukan instansiasi, kemudian lanjutkan dengan mengakses atribut dan method dari objek yang telah terbentuk.

```
public class MahasiswaMain23 {  
    Run | Debug  
    public static void main(String[] args) {  
        Mahasiswa23 mhs1 = new Mahasiswa23();  
        mhs1.nama = "Muhammad Ali Farhan";  
        mhs1.nim = "2241720171";  
        mhs1.kelas = "SI 2J";  
        mhs1.ipk = 3.55;  
  
        mhs1.tampilkanInformasi();  
        mhs1.ubahKelas(kelasBaru:"SI 2K");  
        mhs1.updateIpk(ipkBaru:3.60);  
        mhs1.tampilkanInformasi();  
    }  
}
```

4. Compile dan run 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  
IPK tidak valid. Harus antara 0.0 dan 4.0
```

Jawaban pertanyaan :

1. Proses instansiasi ditandai dengan adanya kata “new”, contoh :

```
Mahasiswa23 mhs1 = new Mahasiswa23();
```

2. Cara mengakses atribut atau method dengan cara menambah “.” Contoh :

```
mhs1.ipk = 3.55;  
  
mhs1.tampilkanInformasi();
```

3. Berbeda, karena sudah terjadi perubahan pada nilai IPK dan kelas.

## 2.3 Latihan 3

Langkah – Langkah percobaan :

1. Buka kembali class Mahasiswa. Tambahkan dua buah konstruktor di dalam class Mahasiswa tersebut, yang terdiri dari satu konstruktor default dan satu konstruktor berparameter. Konstruktor merupakan method istimewa, penempatan kode program untuk konstruktor dapat diperlakukan sama seperti method yang lain (setelah atribut).

```
public Mahasiswa23(){  
}  
  
public Mahasiswa23(String nm, String nim, double ipk, String kls) {  
    nama = nm;  
    this.nim = nim;  
    this.ipk = ipk;  
    kelas = kls;  
}
```

2. Buka kembali class MahasiswaMain. Buat sebuah object lagi bernama mhs2 dengan menggunakan konstruktor berparameter.

```
Mahasiswa23 mhs2 = new Mahasiswa23(nm:"Annisa Nabila",nim:"2141720160", ipk:3.25, kls:"TI 2L");  
mhs2.updateIpk(ipkBaru:5);  
mhs2.tampilkanInformasi();
```

3. Run 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  
IPK tidak valid. Harus antara 0.0 dan 4.0
```

Jawaban pertanyaan :

1. Konstruktor berparameter :

```
public Mahasiswa23(String nm, String nim, double ipk, String kls) {  
    nama = nm;  
    this.nim = nim;  
    this.ipk = ipk;  
    kelas = kls;  
}
```

2. Yang sebenarnya terjadi adalah membuat object baru menggunakan konstruktor berparameter bernama mhs2 dari class Mahasiswa23.
3. Jika dihapus hasilnya akan error, karena java atau kode kehilangan nilai defaultnya.
4. Tidak, karena method tidak saling berhubungan atau bersifat independent.
5. Object baru :

```
Mahasiswa23 mhsVanessa = new Mahasiswa23(nm:"Vanessa Mardiana Putri", nim:"242107020129", ipk:3.90, kls:"TI 1B");  
mhsVanessa.tampilkanInformasi();
```

## 2.4 TUGAS PRAKTIKUM

- Tugas 1

Kode program main :

```
public class MataKuliahMain23{
    Run | Debug
    public static void main(String[] args) {
        MataKuliah23 mk1 = new MataKuliah23();
        mk1.kodeMK = "ALSD";
        mk1.nama = "Algoritma dan Struktur Data";
        mk1.sks = 3;
        mk1.jumlahJam = 4;
        mk1.tampilInformasi();

        mk1.ubahSKS(sksBaru:4);
        mk1.tambahJam(jam:2);
        mk1.kurangiJam(jam:3);
        mk1.tampilInformasi();

        MataKuliah23 mk2 = new MataKuliah23(kodeMK:"BD", nama:"Basis Data", sks:3, jumlahJam:4);
        mk2.tampilInformasi();

        mk2.tambahJam(jam:1);
        mk2.kurangiJam(jam:6);
        mk2.tampilInformasi();
    }
}
```

Kode program :

```
package praktikum2;    Rename this package name to match the regular expression '^([a-z_]+)(\.[a-z_]+)?$'

public class MataKuliah23 {
    String kodeMK;
    String nama;
    int sks;
    int jumlahJam;

    public MataKuliah23() {
    }

    public MataKuliah23(String kodeMK, String nama, int sks, int jumlahJam) {
        this.kodeMK = kodeMK;
        this.nama = nama;
        this.sks = sks;
        this.jumlahJam = jumlahJam;
    }

    void tampilInformasi() {
        System.out.println("Kode MK: " + kodeMK);    Replace this use of System.out by a logger.
        System.out.println("Nama: " + nama);    Replace this use of System.out by a logger.
        System.out.println("SKS: " + sks);    Replace this use of System.out by a logger.
        System.out.println("Jumlah Jam: " + jumlahJam);    Replace this use of System.out by a logger.
        System.out.println();    Replace this use of System.out by a logger.
    }

    void ubahSKS(int sksBaru) {
        sks = sksBaru;
        System.out.println("SKS telah diubah menjadi: " + sks);    Replace this use of System.out by a logger.
    }

    void tambahJam(int jam) {
        jumlahJam += jam;
        System.out.println("Jumlah jam bertambah. Sekarang: " + jumlahJam);    Replace this use of System.out by a logger.
    }
}
```

```

void kurangiJam(int jam) {
    if (jam > jumlahJam) {
        System.out.println("Pengurangan jam tidak dapat dilakukan. Jumlah jam tidak cukup.");
    } else {
        jumlahJam -= jam;
        System.out.println("Jumlah jam telah dikurangi. Sekarang: " + jumlahJam + " jam.");
    }
}
}

```

Hasil run kode program :

```

Kode MK: ALSD
Nama: Algoritma dan Struktur Data
SKS: 3
Jumlah Jam: 4

SKS telah diubah menjadi: 4
Jumlah jam bertambah. Sekarang: 6
Jumlah jam telah dikurangi. Sekarang: 3 jam.
Kode MK: ALSD
Nama: Algoritma dan Struktur Data
SKS: 4
Jumlah Jam: 3

Kode MK: BD
Nama: Basis Data
SKS: 3
Jumlah Jam: 4

Jumlah jam bertambah. Sekarang: 5
Pengurangan jam tidak dapat dilakukan. Jumlah jam tidak cukup.
Kode MK: BD
Nama: Basis Data
SKS: 3
Jumlah Jam: 5

```

- Tugas 2

Kode program main :

```
package praktikum2;    Rename this package name to match the regular expression '^([a-z_]+(\.[a-z_][a-z0-9_]*)*)$'.

public class DosenMain23 {
    Run | Debug
    public static void main(String[] args) {
        Dosen23 dsn1 = new Dosen23();
        dsn1.idDosen = "D001";
        dsn1.nama = "Mungki Astiningrum, S.T., M.Kom.";
        dsn1.statusAktif = true;
        dsn1.tahunBergabung = 2005;
        dsn1.bidangKeahlian = "Algoritma dan Struktur Data";
        dsn1.tampilInformasi();

        dsn1.setStatusAktif(status:false);
        dsn1.ubahKeahlian(bidang:"Data Science");
        dsn1.tampilInformasi();

        Dosen23 dsn2 = new Dosen23(idDosen:"D002", nama:"Imam Fahrur Rozi, S.T., M.T.", statusAktif:true, tahunBergabung:2015, bidang:"Sistem Inf
        dsn2.tampilInformasi();

        dsn2.setStatusAktif(status:true);
        dsn2.ubahKeahlian(bidang:"Manajemen Proyek TI");
        dsn2.tampilInformasi();
    }
}
```

Kode Program :

```
package praktikum2;    Rename this package name to match the regular expression '^([a-z_]+(\.[a-z_][a-z0-9_]*)*)$'.

public class Dosen23 {
    String idDosen;
    String nama;
    boolean statusAktif;
    int tahunBergabung;
    String bidangKeahlian;

    public Dosen23() {
    }

    public Dosen23(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;
    }

    void tampilInformasi() {
        System.out.println("ID Dosen: " + idDosen);    Replace this use of System.out by a logger.
        System.out.println("Nama: " + nama);    Replace this use of System.out by a logger.
        System.out.println("Status Aktif: " + statusAktif);    Replace this use of System.out by a logger.
        System.out.println("Tahun Bergabung: " + tahunBergabung);    Replace this use of System.out by a logger.
        System.out.println("Masa Kerja: " + hitungMasaKerja(thnSkrg:2025) + " tahun");    Replace this use of System.out by a
        System.out.println("Bidang Keahlian: " + bidangKeahlian);    Replace this use of System.out by a logger.
        System.out.println();    Replace this use of System.out by a logger.
    }

    void setStatusAktif(boolean status) {
        statusAktif = status;
        System.out.println("Status dosen telah diubah menjadi: " + (statusAktif ? "Aktif" : "Tidak Aktif"));    Replace this u
    }
}
```



```

int hitungMasaKerja(int thnSkrj) {
    return thnSkrj - tahunBergabung;
}

void ubahKeahlian(String bidang) {
    bidangKeahlian = bidang;
    System.out.println("Bidang keahlian telah diubah menjadi: " + bidangKeahlian);
}
}

```

Hasil kode program :

```

PS D:\kuliah\smt 2\alsd\praktikum-alsd-1> & 'C:\Program Files\Java\jdk-22\bin\
ng\Code\User\workspaceStorage\5d67e40688239c9a123e563de53f8bf6\redhat.java\jdt_
ID Dosen: D001
Nama: Mungki Astiningrum, S.T., M.Kom.
Status Aktif: true
Tahun Bergabung: 2005
Masa Kerja: 20 tahun
Bidang Keahlian: Algoritma dan Struktur Data

Status dosen telah diubah menjadi: Tidak Aktif
Bidang keahlian telah diubah menjadi: Data Science
ID Dosen: D001
Nama: Mungki Astiningrum, S.T., M.Kom.
Status Aktif: false
Tahun Bergabung: 2005
Masa Kerja: 20 tahun
Bidang Keahlian: Data Science

ID Dosen: D002
Nama: Imam Fahrur Rozi, S.T., M.T.
Status Aktif: true
Tahun Bergabung: 2015
Masa Kerja: 10 tahun
Bidang Keahlian: Sistem Informasi

Status dosen telah diubah menjadi: Aktif
Bidang keahlian telah diubah menjadi: Manajemen Proyek TI
ID Dosen: D002
Nama: Imam Fahrur Rozi, S.T., M.T.
Status Aktif: true
Tahun Bergabung: 2015
Masa Kerja: 10 tahun
Bidang Keahlian: Manajemen Proyek TI

```