

**LAPORAN PRAKTIKUM ALGORITMA DAN STRUKTUR DATA**  
**JOBSHEET II : OBJECT**



Aylafada Syakira

TI-1C

254107020116

**PROGRAM STUDI D-IV TEKNIK INFORMATIKA**  
**JURUSAN TEKNOLOGI INFORMASI**  
**POLITEKNIK NEGERI MALANG 2026**

## 2.1 Percobaan 1: Deklarasi Class, Atribut dan Method

### 2.1.1 Langkah-langkah Percobaan

Berikut hasil program percobaan 1:

```
package minggu2;

//class: template untuk membuat object
public class Mahasiswa06 {
    //atribut: data yang memiliki object
    String nama;
    String nim;
    String kelas;
    double ipk;

    //method/fungsi: digunakan object untuk berinteraksi dengan object yang lain
    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 >= 2.0) {
            return "Kinerja cukup";
        } else {
            return "Kinerja kurang";
        }
    }
}
```

### 2.1.3 Pertanyaan

1. Karakteristik Object: Mempunyai sesuatu (seperti data, properti, variabel, state, atribut) dan melakukan sesuatu
2. Atribut yang dimiliki oleh class Mahasiswa adalah nama, nim, kelas, dan ipk
3. Method yang dimiliki oleh class tersebut adalah tampilkanInformasi, ubahKelas, updateIpk, nilaiKinerja

4. Berikut hasil modifikasi updateIpk():

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

5. Kriteria yang digunakan untuk menentukan nilai kinerja adalah
- IPK > 3.5 "Kinerja sangat baik"  
IPK >= 3.0 "Kinerja baik"  
IPK >= 2.0 "Kinerja cukup"  
Selain itu "Kinerja kurang"
  - Yang dikembalikan oleh method nilaiKinerja() adalah  
"Kinerja sangat baik"  
"Kinerja baik"  
"Kinerja cukup"  
"Kinerja kurang".

## 2.2 Percobaan 2: Instansiasi Object, serta Mengakses Atribut dan Method

### 2.2.1 Langkah-langkah Percobaan

Berikut hasil program percobaan 2:

```
//object  
Mahasiswa06 mhs1 = new Mahasiswa06();  
mhs1.nama = "Muhammad Ali Farhan";  
mhs1.nim = "2241720171";  
mhs1.kelas = "SI 2J";  
mhs1.ipk = 3.55;  
  
//data sebelum diubah  
mhs1.tampilkanInformasi();  
//memberikan nilai atribut dari object mhs1  
mhs1.ubahKelas(kelasBaru: "SI 2K");  
mhs1.updateIpk(ipkBaru: 3.60);  
//data sesudah diubah (menjalankan method dari object mhs1)  
mhs1.tampilkanInformasi();
```

### 2.2.3 Pertanyaan

1. Berikut baris program yang digunakan untuk proses instansiasi:

```
Mahasiswa06 mhs2 = new Mahasiswa06 (nm: "Annisa Nabila", nim: "2141720160", ipk: 3.25, kls: "TI 2L");  
mhs2.updateIpk (ipkBaru: 3.30);  
mhs2.tampilkanInformasi();
```

Nama object yang dihasilkan adalah mhs2

2. Cara mengakses atribut dan method dari suatu objek adalah dengan menggunakan operator titik. Berikut baris program yang digunakan untuk mengakses atribut dan method dari suatu objek:

```
mhs1.nama = "Muhammad Ali Farhan";  
mhs1.nim = "2241720171";  
mhs1.kelas = "SI 2J";  
mhs1.ipk = 3.55;
```

3. Hasil output pemanggilan method tampilkanInformasi() pertama dan kedua berbeda karena sebelum pemanggilan data kedua ada perubahan data object, karena jika state berubah maka output method yang membaca state itu juga ikut berubah

## 2.3 Percobaan 3: Membuat Konstruktor

### 2.3.1 Langkah-langkah Percobaan

1. Berikut 2 buah konstruktor yang telah ditambahkan ke percobaan 1:

```
//konstruktor default  
public Mahasiswa06(){  
}  
  
//konstruktor berparameter  
public Mahasiswa06 (String nm, String nim, double ipk, String kls) {  
    //digunakan untuk mengakses atribut  
    nama = nm;  
    this.nim = nim;  
    this.ipk = ipk;  
    kelas = kls;  
}
```

2. Berikut object yang telah ditambahkan dengan menggunakan konstruktor berparameter:

```
//instansiasi: pembuatan object dari class (new)  
//konstruktor (istimewa): nama method sama dengan nama class, tdk ada tipe data  
Mahasiswa06 mhs2 = new Mahasiswa06 (nm: "Annisa Nabila", nim: "2141720160", ipk: 3.25, kls: "TI 2L");  
mhs2.updateIpk (ipkBaru: 3.30);  
mhs2.tampilkanInformasi();
```

### 2.3.3 Pertanyaan

1. Berikut baris program yang digunakan untuk mendeklarasikan konstruktor berparameter:

```
public Mahasiswa06 (String nm, String nim, double ipk, String kls) {  
    nama = nm;  
    this.nim = nim;  
    this.ipk = ipk;  
    kelas = kls;  
}
```

2. Membuat object baru dari class mahasiswa dan menginisialisasi data
3. Maka akan terjadi error, object harus dibuat dengan parameter
4. Tidak, karena method dapat dipanggil kapan saja dan urutan pemanggilan hanya berpengaruh jika method tersebut mengubah data
5. Berikut object baru menggunakan konstruktor berparameter dari class Mahasiswa:

```
Mahasiswa06 mhsAyla = new Mahasiswa06 (nm: "Aylafada Syakira", nim: "2541070116", ipk: 3.95, kls: "TI 1C");  
mhsAyla.updateIpk (ipkBaru: 4.00);  
mhsAyla.tampilkanInformasi();
```

### 2.4 Latihan Praktikum

1. Tugas 1

Berikut adalah hasil program MataKuliah06:

```
package minggu2;  
  
public class MataKuliah06 {  
    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);  
        System.out.println();  
    }  
  
    void ubahSKS (int sksBaru) {  
        sks = sksBaru;  
        System.out.println("SKS telah diubah menjadi: " + sks);  
    }  
  
    void tambahJam (int jam) {  
        jumlahJam += jam;  
        System.out.println("Jumlah jam setelah ditambah: " + jumlahJam);  
    }  
  
    void kurangiJam (int jam) {  
        if(jumlahJam >= jam){  
            jumlahJam -= jam;  
            System.out.println("Jumlah jam setelah dikurangi: " + jumlahJam);  
        } else if (jumlahJam <= jam) {  
            System.out.println(x: "Pengurangan tidak dapat dilakukan. ");  
        }  
    }  
  
    public MataKuliah06(){  
    }  
  
    public MataKuliah06(String kodeMK, String nm, int sks, int jumlahJam){  
        this.kodeMK = kodeMK;  
        nama = nm;  
        this.sks = sks;  
        this.jumlahJam = jumlahJam;  
    }  
}
```

Berikut hasil program MataKuliahMain06:

```
public class MataKuliahMain06 {  
    Run | Debug | Run main | Debug main  
    public static void main(String[] args) {  
        MataKuliah06 mk1 = new MataKuliah06();  
        mk1.kodeMK = "ASD1C";  
        mk1.nama = "Praktikum Algoritma dan Struktur Data";  
        mk1.sks = 4;  
        mk1.jumlahJam = 8;  
  
        mk1.tampilInformasi();  
        mk1.ubahSKS(sksBaru: 5);  
        mk1.tambahJam(jam: 2);  
        mk1.kurangiJam(jam: 4);  
        mk1.tampilInformasi();  
  
        MataKuliah06 mk2 = new MataKuliah06 (kodeMK: "BD1C", nm: "Basis Data", sks: 3, jumlahJam: 6);  
  
        mk2.tampilInformasi();  
        mk2.ubahSKS(sksBaru: 2);  
        mk2.tambahJam(jam: 5);  
        mk2.kurangiJam(jam: 4);  
        mk2.tampilInformasi();  
    }  
}
```

Berikut Output Program:

```
Kode Mata Kuliah: ASD1C  
Nama Mata Kuliah: Praktikum Algoritma dan Struktur Data  
SKS: 4  
Jumlah Jam: 8  
  
SKS telah diubah menjadi: 5  
Jumlah jam setelah ditambah: 10  
Jumlah jam setelah dikurangi: 6  
Kode Mata Kuliah: ASD1C  
Nama Mata Kuliah: Praktikum Algoritma dan Struktur Data  
SKS: 5  
Jumlah Jam: 6  
  
Kode Mata Kuliah: BD1C  
Nama Mata Kuliah: Basis Data  
SKS: 3  
Jumlah Jam: 6  
  
SKS telah diubah menjadi: 2  
Jumlah jam setelah ditambah: 11  
Jumlah jam setelah dikurangi: 7  
Kode Mata Kuliah: BD1C  
Nama Mata Kuliah: Basis Data  
SKS: 2  
Jumlah Jam: 7
```

a. Atribut

Berikut atribut pada class MataKuliah06:

```
String kodeMK;  
String nama;  
int sks;  
int jumlahJam;
```

b. Method

Berikut adalah method pada class MataKuliah06:

```
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);  
    System.out.println();  
}  
  
void ubahSKS (int sksBaru) {  
    sks = sksBaru;  
    System.out.println("SKS telah diubah menjadi: " +sks);  
}  
  
void tambahJam (int jam) {  
    jumlahJam += jam;  
    System.out.println("Jumlah jam setelah ditambah: " +jumlahJam);  
}  
  
void kurangiJam (int jam) {  
    if(jumlahJam>=jam){  
        jumlahJam -= jam;  
        System.out.println("Jumlah jam setelah dikurangi: " +jumlahJam);  
    } else if (jumlahJam<=jam) {  
        System.out.println(x: "Pengurangan tidak dapat dilakukan. ");  
    }  
}
```

2. Tugas 2

Berikut hasil program Dosen06:

```

public class Dosen06 {
    String idDosen;
    String nama;
    boolean statusAktif;
    int tahunBergabung;
    String bidangKeahlian;

    void tampilInformasi() {
        System.out.println("ID Dosen: " +idDosen);
        System.out.println("Nama Lengkap: " +nama);
        System.out.println("Status Aktif: " +statusAktif);
        System.out.println("Tahun Bergabung: " +tahunBergabung);
        System.out.println("Bidang Keahlian: " +bidangKeahlian);
        System.out.println();
    }

    void setStatusAktif (boolean status) {
        statusAktif = status;
    }

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

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

    public Dosen06() {
    }

    public Dosen06(String id, String nm, boolean statusAktif, int tahunBergabung, String bidangKeahlian) {
        idDosen = id;
        nama = nm;
        this.statusAktif = statusAktif;
        this.tahunBergabung = tahunBergabung;
        this.bidangKeahlian = bidangKeahlian;
    }
}

```

Berikut hasil program DosenMain06:

```

public class DosenMain06 {
    Run | Debug | Run main | Debug main
    public static void main(String[] args) {
        Dosen06 dsn1 = new Dosen06();
        dsn1.idDosen = "12345";
        dsn1.nama = "Aviv Ardhillah";
        dsn1.statusAktif = true;
        dsn1.tahunBergabung = 2017;
        dsn1.bidangKeahlian = "Kalkulus";

        dsn1.tampilInformasi();
        dsn1.ubahKeahlian(bidang: "Aljabar Linear");
        System.out.println("Masa Kerja: " +dsn1.hitungMasaKerja(thnSkrng: 2026) + " tahun");
        dsn1.tampilInformasi();

        Dosen06 dsn2 = new Dosen06(id: "12334", nm: "Rina", statusAktif: false, tahunBergabung: 2010, bidangKeah... "English Literature");
        dsn2.tampilInformasi();
        dsn2.ubahKeahlian(bidang: "English for Communication");
        System.out.println("Masa Kerja: " +dsn2.hitungMasaKerja(thnSkrng: 2026) + " tahun");
        dsn2.tampilInformasi();
    }
}

```



Berikut adalah Output Program:

```
ID Dosen: 12345
Nama Lengkap: Aviv Ardhillah
Status Aktif: true
Tahun Bergabung: 2017
Bidang Keahlian: Kalkulus

Bidang keahlian telah diubah menjadi Aljabar Linear
Masa Kerja: 9 tahun
ID Dosen: 12345
Nama Lengkap: Aviv Ardhillah
Status Aktif: true
Tahun Bergabung: 2017
Bidang Keahlian: Aljabar Linear

ID Dosen: 12334
Nama Lengkap: Rina
Status Aktif: false
Tahun Bergabung: 2010
Bidang Keahlian: English Literature

Bidang keahlian telah diubah menjadi English for Communication
Masa Kerja: 16 tahun
ID Dosen: 12334
Status Aktif: false
Tahun Bergabung: 2010
Bidang Keahlian: English for Communication
```