

POLITEKNIK NEGERI MALANG
TEKNOLOGI INFORMASI
TEKNIK INFORMATIKA



Nama: Abdul Rahman Hanif Darmawan

NIM: 244107020232

Kelas: TI-1A

Prodi: D4-TEKNIK INFORMATIKA

2.1.3 Pertanyaan

1. Sebutkan dua karakteristik class atau object!
 - Encapsulation adalah Data dalam class hanya bisa diakses melalui method yang disediakan. dan Inheritance adalah Class dapat mewarisi atribut dan method dari class lain.
2. Perhatikan class Mahasiswa pada Praktikum 1 tersebut, ada berapa atribut yang dimiliki oleh class Mahasiswa? Sebutkan apa saja atributnya!
 - nama(String), nim (String), kelas (String), ipk (double).
3. Ada berapa method yang dimiliki oleh class tersebut? Sebutkan apa saja methodnya!
 - tampilkanInformasi(), ubahKelas(String kelasBaru), updateIpk(double ipkBaru), nilaiKinerja()
4. Perhatikan method updateIpk() yang terdapat di dalam class Mahasiswa. Modifikasi isi method tersebut sehingga IPK yang dimasukkan valid yaitu terlebih dahulu dilakukan pengecekan apakah IPK yang dimasukkan di dalam rentang 0.0 sampai dengan 4.0 ($0.0 \leq \text{IPK} \leq 4.0$). Jika IPK tidak pada rentang tersebut maka dikeluarkan pesan: "IPK tidak valid. Harus antara 0.0 dan 4.0".

-

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

5. Jelaskan bagaimana cara kerja method nilaiKinerja() dalam mengevaluasi kinerja mahasiswa, kriteria apa saja yang digunakan untuk menentukan nilai kinerja tersebut, dan apa yang dikembalikan (di-return-kan) oleh method nilaiKinerja() tersebut?
 - Method ini mengevaluasi kinerja mahasiswa berdasarkan ipk dan kriteria yang digunakan adalah
 - $\text{IPK} \geq 3.5 \rightarrow \text{"Kinerja sangat baik"}$
 - $3.0 \leq \text{IPK} < 3.5 \rightarrow \text{"Kinerja baik"}$
 - $2.0 \leq \text{IPK} < 3.0 \rightarrow \text{"Kinerja cukup"}$
 - $\text{IPK} < 2.0 \rightarrow \text{"Kinerja kurang"}$
 - Return value method yang mengembalikan string sesuai dengan kategori kinerja mahasiswa

2.2.3 Pertanyaan

1. Pada class MahasiswaMain, tunjukkan baris kode program yang digunakan untuk proses instansiasi! Apa nama object yang dihasilkan?
 - ```
Mahasiswa02 mhs1 = new Mahasiswa02();
```
2. Bagaimana cara mengakses atribut dan method dari suatu objek?
  - Mengakses atribut dan method dari suatu objek menggunakan tanda “.” Dan memanggil method juga menggunakan tanda “.”
3. Mengapa hasil output pemanggilan method tampilkanInformasi() pertama dan kedua berbeda?
  - output tampilkanInformasi() pertama berbeda dengan kedua, karena yang pertama sebelum dipanggil atribut “kelas” dan “ipk” sudah diubah, dan setelah itu tidak ada perubahan.

### 2.3.3 Pertanyaan

1. Pada class Mahasiswa di Percobaan 3, tunjukkan baris kode program yang digunakan untuk mendeklarasikan konstruktor berparameter!

```
public Mahasiswa02(String nm, String nim, double ipk, String kls) {
 nama = nm;
 this.nim = nim;
 this.ipk = ipk;
 kelas = kls;
}
```

2. Perhatikan class MahasiswaMain. Apa sebenarnya yang dilakukan pada baris program berikut?

```
Mahasiswa mhs2 = new Mahasiswa("Annisa Nabila", "2141720160", 3.25, "TI 2L");
```

- Baris ini membuat objek mhs2 dari class Mahasiswa02 menggunakan konstruktor parameter. Dan data mahasiswa Annissa langsung diinisialisasi ke atribut nama, nim, ipk dan kelas.
3. Hapus konstruktor default pada class Mahasiswa, kemudian compile dan run program. Bagaimana hasilnya? Jelaskan mengapa hasilnya demikian!
    - Terdapat error pada “Mahasiswa02 mhs1 = new Mahasiswa02();”, dan hal ini terjadi karena konstruktor utama atau default dihapus dan konstruktor yang tersisa adalah konstruktor parameter.
  4. Setelah melakukan instansiasi object, apakah method di dalam class Mahasiswa harus diakses secara berurutan? Jelaskan alasannya!
    - Tidak harus berurutan method dapat dipanggil dalam urutan yang sesuai dengan kebutuhan programnya.

5. Buat object baru dengan nama mhs menggunakan konstruktor berparameter dari class Mahasiswa!

-

```
Mahasiswa02 mhsHanif = new Mahasiswa02("Abdul Rahman Hanif Darmawan",
"244107020232", 3.55, "TI 1A");
 mhsHanif.updateIpk(3.89);
 mhsHanif.tampilkanInformasi();
```

## 2.4 Latihan Praktikum

### 1. MataKuliah02.java

```
package Tugas_1;
public class MataKuliah02 {
 String kodeMK;
 String nama;
 int sks;
 int jumlahJam;

 // konstruktor default
 public MataKuliah02() {
 this.kodeMK = "";
 this.nama = "";
 this.sks = 0;
 this.jumlahJam = 0;
 }

 // konstruktor berparameter
 public MataKuliah02(String kodeMK, String nama, int sks, int jumlahJam)
 {
 this.kodeMK = kodeMK;
 this.nama = nama;
 this.sks = sks;
 this.jumlahJam = jumlahJam;
 }

 // method yang digunakan untuk menampilkan informasi mata kuliah
 public 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("-----");
 }

 // method yang digunakan untuk mengubah SKS
 public void ubahSKS(int sksBaru) {
 this.sks = sksBaru;
 System.out.println("SKS berhasil diubah menjadi: " + sks);
 }

 // mehtod yang digunakan untuk menambah jam pada mata kuliah
 public void tambahJam(int jam) {
 this.jumlahJam += jam;
 System.out.println("Jumlah jam setelah ditambah: " + jumlahJam);
 }

 // method yang digunakan untuk mengurangi jam mata kuliah
 public void kurangiJam(int jam) {
 if (jam > jumlahJam) {
 System.out.println("Pengurangan tidak dapat dilakukan. Jumlah
jam tidak mencukupi.");
 } else {
 this.jumlahJam -= jam;
 System.out.println("Jumlah jam setelah dikurangi: " +
jumlahJam);
 }
 }
}
```

## MataKuliah02Main.java

```
package Tugas_1;

public class MataKuliah02Main {
 public static void main(String[] args) {
 MataKuliah02 mk1 = new MataKuliah02();
 mk1.kodeMK = "TI101";
 mk1.nama = "Pemrograman Java";
 mk1.sks = 3;
 mk1.jumlahJam = 4;

 mk1.tampilInformasi();
 // mengubah SKS dan mengatur jumlah jam
 mk1.ubahSKS(4);
 mk1.tambahJam(2);
 mk1.kurangiJam(3);
 mk1.tampilInformasi();

 MataKuliah02 mk2 = new MataKuliah02("TI102", "Struktur Data", 3,
6);
 mk2.tampilInformasi();

 // mngubah SKS dan mengatur jumlah jam
 mk2.ubahSKS(4);
 mk2.tambahJam(1);
 mk2.kurangiJam(5);
 mk2.tampilInformasi();
 }
}
```

```
Kode Mata Kuliah: TI101
Nama Mata Kuliah: Pemrograman Java
SKS: 3
Jumlah Jam: 4

SKS berhasil diubah menjadi: 4
Jumlah jam setelah ditambah: 6
Jumlah jam setelah dikurangi: 3
Kode Mata Kuliah: TI101
Nama Mata Kuliah: Pemrograman Java
SKS: 4
Jumlah Jam: 3

Kode Mata Kuliah: TI102
Nama Mata Kuliah: Struktur Data
SKS: 3
Jumlah Jam: 6

SKS berhasil diubah menjadi: 4
Jumlah jam setelah ditambah: 7
Jumlah jam setelah dikurangi: 2
Kode Mata Kuliah: TI102
Nama Mata Kuliah: Struktur Data
SKS: 4
Jumlah Jam: 2

```

## 2. Dosen02.java

```
package Tugas_2;
public class Dosen02 {
 String idDosen;
 String nama;
 boolean statusAktif;
 int tahunBergabung;
 String bidangKeahlian;

 // konstruktor default
 public Dosen02() {
 this.idDosen = "";
 this.nama = "";
 this.statusAktif = false;
 this.tahunBergabung = 0;
 this.bidangKeahlian = "";
 }

 // konstruktor berparameter
 public Dosen02(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;
 }

 // method digunakan untuk menampilkan informasi dosen
 public void tampilInformasi() {
 System.out.println("ID Dosen: " + idDosen);
 System.out.println("Nama: " + nama);
 System.out.println("Status Aktif: " + (statusAktif ? "Aktif" :
"Tidak Aktif"));
 System.out.println("Tahun Bergabung: " + tahunBergabung);
 System.out.println("Bidang Keahlian: " + bidangKeahlian);
 System.out.println("-----");
 }

 // method yang mengatur status aktif tidak nya dosen
 public void setStatusAktif(boolean status) {
 this.statusAktif = status;
 System.out.println("Status aktif berhasil diubah menjadi: " +
(statusAktif ? "Aktif" : "Tidak Aktif"));
 }

 // method yang digunakan untuk menghitung masa kerja
 public int hitungMasaKerja(int thnSkrng) {
 return thnSkrng - tahunBergabung;
 }

 // method yang digunakan untuk mengubah bidang keahlian dosen
 public void ubahKeahlian(String bidang) {
 this.bidangKeahlian = bidang;
 System.out.println("Bidang keahlian berhasil diubah menjadi: " +
bidangKeahlian);
 }
}
```

## DosenMain02.java

```
package Tugas_2;

public class DosenMain02 {
 public static void main(String[] args) {
 Dosen02 dosen1 = new Dosen02();
 dosen1.idDosen = "D001";
 dosen1.nama = "Dr. Ahmad Fauzi";
 dosen1.statusAktif = true;
 dosen1.tahunBergabung = 2010;
 dosen1.bidangKeahlian = "Kecerdasan Buatan";
 dosen1.tampilInformasi();

 // mengubah status aktif, bidang keahlian, dan menghitung masa
 kerja
 dosen1.setStatusAktif(false);
 dosen1.ubahKeahlian("Machine Learning");
 System.out.println("Masa kerja dosen1: " +
 dosen1.hitungMasaKerja(2025) + " tahun");
 dosen1.tampilInformasi();

 Dosen02 dosen2 = new Dosen02("D002", "Prof. Siti Aminah", true,
 2015, "Sistem Informasi");
 dosen2.tampilInformasi();

 // mengubah status aktif dan bidang keahlian
 dosen2.setStatusAktif(true);
 dosen2.ubahKeahlian("Data Science");
 System.out.println("Masa kerja dosen2: " +
 dosen2.hitungMasaKerja(2025) + " tahun");
 dosen2.tampilInformasi();
 }
}
```

```
ID Dosen: D001
Nama: Dr. Ahmad Fauzi
Status Aktif: Aktif
Tahun Bergabung: 2010
Bidang Keahlian: Kecerdasan Buatan

Status aktif berhasil diubah menjadi: Tidak Aktif
Bidang keahlian berhasil diubah menjadi: Machine Learning
Masa kerja dosen1: 15 tahun
ID Dosen: D001
Nama: Dr. Ahmad Fauzi
Status Aktif: Tidak Aktif
Tahun Bergabung: 2010
Bidang Keahlian: Machine Learning

ID Dosen: D002
Nama: Prof. Siti Aminah
Status Aktif: Aktif
Tahun Bergabung: 2015
Bidang Keahlian: Sistem Informasi

Status aktif berhasil diubah menjadi: Aktif
Bidang keahlian berhasil diubah menjadi: Data Science
Masa kerja dosen2: 10 tahun
ID Dosen: D002
Nama: Prof. Siti Aminah
Status Aktif: Aktif
Tahun Bergabung: 2015
Bidang Keahlian: Data Science

```



Link Github Praktikum dan Tugas:

<https://github.com/baynobu/ALSD/tree/c979c1014ac1b3108938631aded82ce1c451e885/Pertemuan%202>