

NAMA : FAFIQ LUTFI AZANA
KELAS : TI_1G
NIM : 254107020058

JOB SHEET 2

PERCOBAAN 1

```
package PraktikumAlgoritmaDanStrukturData.jobsheet2;

public class Mahasiswa09 {
    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) {
        ipk = ipkBaru;
    }

    String nilaiKerja() {
        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 kurangss";
        }
    }
}
```

VERIFIKASI HASIL PERCOBAAN 1

```
fafiq@fafiq-IdeaPad-Slim-5-14AKP10:~/Documents/kuliah/semester2/ALGORITMA DAN STRUKTUR DATA/p
aktikum-algoritma-dan-struktur-data$ java Mahasiswa09
Error: Could not find or load main class Mahasiswa09
Caused by: java.lang.ClassNotFoundException: Mahasiswa09
```

PERTANYAAN

1. Dua karakteristik dalam class/object yang pertama adalah attribute/property berupa variabel yang dimiliki class. pada percobaan 1 contoh atributnya adalah nama, nim, kelas, dan ipk. kemudian yang kedua adalah method/behavior yang menentukan aksi/perilaku yang dapat dilakukan object. pada percobaan 1 contoh method/behavior adalah tampilkanInformasi untuk menampilkan informasi dari attribute yang dimiliki class.
2. Class Mahasiswa memiliki 4 attribute yaitu nama, nim, kelas, dan ipk.
3. Class Mahasiswa memiliki 4 method, yaitu tampilkanInfotmasi, ubahKelas, updatelpk, dan nilaiKerja.

4.

```
void updateIpk(double ipkBaru) {
    if (ipk >= 0 && ipk <= 4) {
```

```
        ipk = ipkBaru;
    } else {
        System.out.println("IPK tidak valid. Harus antara
0.0 dan 4.0");
    }
}
```

5. Method nilaiKinerja menilai kinerja mahasiswa berdasarkan nilai atribut ipk yang dimiliki mahasiswa. Jika mahasiswa memiliki ipk minimal 3.5 atau lebih, maka kinerja mahasiswa dikatakan sangat baik. Jika mahasiswa memiliki ipk minimal 3.0 dan kurang dari 3.5 maka kinerja mahasiswa dikatakan baik. Jika mahasiswa memiliki ipk minimal 2.0 dan kurang dari 3.0 maka kinerja mahasiswa dikatakan cukup. Jika mahasiswa memiliki ipk dibawah 2.0 maka kinerja mahasiswa dikatakan kurang.

azana123 Percobaan 1	
Name	Last commit message
..	
Mahasiswa09.java	Percobaan 1

6.

PERCOBAAN 2

```
package PraktikumAlgoritmaDanStrukturData.jobsheet2;

public class MahasiswaMain09 {
    public static void main(String[] args) {
        Mahasiswa09 mhs1 = new Mahasiswa09();
        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();
}
}

```

PraktikumAlgoritmaDanStrukturData / jobsheet2 / Add file ...

azana123 Percobaan 2 dfbd2b2 · now History

Name	Last commit message	Last commit da...
..		
Mahasiswa09.java	Percobaan 1	35 minutes ago
MahasiswaMain09.java	Percobaan 2	now

VERIFIKASI HASIL PERCOBAAN 2

```

* fafiq@fafiq-IdeaPad-Slim-5-14AKP10:~/Documents/kuliah/semester2/ALGORITMA DAN STRUKTUR DATA/praktikum-algoritma-dan-struktur-datas /usr/bin/env /usr/lib/jvm/jdk-25.0.2-oracle-x64/bin/java --e
nable-preview -XX:+ShowCodeDetailsInExceptionMessages -cp /home/fafiq/.config/Code/User/workspaceStorage/46a45ab369a54c8fa6da2cb508f893d0/redhat.java/jdt_ws/praktikum-algoritma-dan-struktur-da
ta_4e9c6975/bin PraktikumAlgoritmaDanStrukturData.jobsheet2.MahasiswaMain09
Nama: Muhammad Ali Farhan
NIM: 2241720171
IPK: 3.55
Kelas: SI 2J
Nama: Muhammad Ali Farhan
NIM: 2241720171
IPK: 3.6
Kelas: SI 2K
* fafiq@fafiq-IdeaPad-Slim-5-14AKP10:~/Documents/kuliah/semester2/ALGORITMA DAN STRUKTUR DATA/praktikum-algoritma-dan-struktur-datas

```

PERTANYAAN

1. Pada class MahasiswaMain baris kode yang digunakan untuk instansiasi adalah baris `Mahasiswa09 mhs1 = new Mahasiswa09();` dimana Mahasiswa09 adalah nama class, new Mahasiswa09() adalah proses instansiasi, dan mhs1 adalah object yang dihasilkan
2. Cara mengakses atribut dan method dari suatu objek adalah melakukan instansiasi dan melakukan pemanggilan menggunakan operator titik ".". Pada percobaan 2 ini dilakukan instansiasi pada baris kode `Mahasiswa09 mhs1 = new Mahasiswa09();` kemudian dilakukan pemanggilan method dan atribut dari class yang telah diinstansiasi pada menggunakan baris kode `mhs1.tampilkanInformasi();` dan `mhs1.nama = "Muhammad Ali Farhan";`
3. Karena pada pemanggilan method tampilkanInformasi() yang keuda dilakukan setelah pemanggilan method updateIpk() dan ubahKelas() yang mengganti nilai dari atribut yang lama menjadi baru sehingga hasil pemanggilannya berbeda.

PERCOBAAN 3

A. MahasiswaMain09

```

package PraktikumAlgoritmaDanStrukturData.jobsheet2;

public class MahasiswaMain09 {
    public static void main(String[] args) {

```

```

        Mahasiswa09 mhs1 = new Mahasiswa09();
        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();

        Mahasiswa09 mhs2 = new Mahasiswa09("Annisa Nabila",
"2141720160", 3.25, "TI 2L");
        mhs2.updateIpk(3.30);
        mhs2.tampilkanInformasi();
    }
}

```

B. Mahasiswa09

```

package PraktikumAlgoritmaDanStrukturData.jobsheet2;

public class Mahasiswa09 {
    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 (ipk >= 0 && ipk <= 4) {
            ipk = ipkBaru;
        }
    }
}

```

```

        } else {
            System.out.println("IPK tidak valid. Harus antara 0.0 dan
4.0");
        }
    }

    String nilaiKerja() {
        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 kurangss";
        }
    }

    public Mahasiswa09() {
    }

    public Mahasiswa09(String nm, String nim, double ipk, String kls)
{
        nama = nm;
        this.nim = nim;
        this. ipk = ipk;
        kelas = kls;
    }
}

```

VERIFIKASI HASIL PERCOBAAN

```

• fafiq@fafiq-IdeaPad-Slim-5-14AKP10:~/Documents/kuliah/ser
exceptionMessages -cp /home/fafiq/.config/Code/User/worksp
9
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
○ fafiq@fafiq-IdeaPad-Slim-5-14AKP10:~/Documents/kuliah/ser

```

PERTANYAAN

1. Baris yang kode yang menunjukkan deklarasi konstruktor berparameter adalah

```
public Mahasiswa09(String nm, String nim, double ipk, String kls) {
```

 karena memiliki nama yang sama dengan nama class, dan memiliki parameter dan tidak memiliki nilai return.
2. pada class MahasiswaMain, baris tersebut berfungsi untuk membuat objek baru pada class MahasiswaMain dengan nama mhs2 dan mengisi nilai atribut objek tersebut dengan data yang ada pada parameter sehingga mhs2 mereferensikan data di dalam parameter tersebut.
3. ketika konstruktor default dihapus, ketika mencoba mengompile dan menjalankan program akan terdapat pesan "the constructor Mahasiswa is undefined. program tidak berjalan ketika konstruktor default dihapus karena java tidak lagi memiliki konstruktor kosong untuk dipanggil ketika objek dibuat dengan new Mahasiswa09() sehingga terjadi error karena tidak ada konstruktor yang sesuai dengan pemanggilan tersebut
4. method dalam class tidak harus diakses secara berurutan setelah instansiasi objek karena method bersifat independen dan dapat diakses kapan saja dengan urutan apa saja asalkan memenuhi syarat pemanggilan seperti parameter yang sesuai.
- 5.

```

Mahasiswa09 mhs3 = new Mahasiswa09("Fafiq Lutfi Azana",
"254107020058", 3.68, "TI 1G");
mhs3.tampilkanInformasi();

```

..		
Mahasiswa09.java	Soal Nomor 6	NOV
MahasiswaMain09.java	Soal Nomor 6	NOV

6.