

**Nama : Rayana Jaka Surya**  
**Kelas : TI/1C**  
**Absen : 25**  
**NIM : 254107020026**

### 2.1.3 Pertanyaan

**Jawaban :**

1. karakteristik class atau object adalah memiliki atribut(menyimpan data) dan method(perilaku yang dapat dilakukan object).
2. ada 4 atribut pada class mahasiswa yaitu nim, nama, kelas, ipk
3. Terdapat 4 method operasional yang dimiliki oleh class tersebut yaitu tampilkanInformasi(), ubahKelas(kelasBaru: String), updateIpk(ipkBaru: double), nilaiKinerja(ipk: double)
- 4.

```
void updateIpk(double ipkBaru){  
    if (ipk >= 0 && ipk <= 4.0) {  
        ipk = ipkBaru;  
    } else {  
        System.out.println("IPK tidak valid. Harus antara 0.0 dan 4.0");  
    }  
}
```

```
● PS E:\GitHub\PraktikumASD\PraktikumASD_1>  
-XX:+ShowCodeDetailsInExceptionMessages  
t.java\jdt_ws\PraktikumASD_1>  
Nama Muhammad Ali Farhan  
NIM 2241720271  
Kelas S1 2J  
IPK 3.55  
Nama Muhammad Ali Farhan  
NIM 2241720271  
Kelas SI 2K  
IPK 3.6
```

5. Method nilaiKinerja() mengevaluasi kinerja mahasiswa dengan cara melakukan pengecekan bertahap pada nilai atribut ipk menggunakan percabangan if-else. Method ini mengembalikan data jenis String.

### 2.2.3 Pertanyaan

**Jawaban :**

1. Proses instansiasi pada class MahasiswaMain dilakukan pada baris kode Mahasiswa mhs1 = new Mahasiswa(), dari baris kode tersebut, object yang dihasilkan diberi nama mhs1.
2. dengan menggunakan operator titik(.). contoh mengakses atribut: mhs1.nama = "Muhammad Ali Farhan";. contoh mengakses method: mhs1.tampilkanInformasi();
3. Hasil output pada pemanggilan pertama dan kedua berbeda karena adanya perubahan nilai (state) pada atribut objek mhs1. Setelah pemanggilan method tampilkanInformasi() yang pertama , program mengeksekusi dua method, yaitu mhs1.ubahKelas("SI 2K"); dan mhs1.updateIpk(3.60);. Pemanggilan ini menimpa data kelas dan IPK sebelumnya. Oleh karena itu, saat tampilkanInformasi() dipanggil untuk kedua kalinya, program mencetak data state objek mhs1 yang paling baru.

### **2.3.3 Pertanyaan**

**Jawaban :**

1.

```
public Mahasiswa25(String nm, String nim, double ipk, String kls) {
    nama = nm;
    this.nim = nim;
    this.ipk = ipk;
    kelas = kls;
}
```

2. memanggil konstruktor berparameter untuk melakukan inisialisasi (pemberian nilai awal) secara langsung ke dalam atribut-atribut milik objek tersebut.
3. saat konstruktor default dihapus hasilnya error karena aturan java kalau sebuah class sudah memiliki konstruktor berparameter java tidak akan otomatis membuatkan konstruktor kosong. Karena konstruktor kosongnya saya hapus, program jadi error karena tidak menemukan konstruktor untuk membuat mhs1.
4. Tidak harus diakses secara berurutan. Setelah objek diinstansiasi, pemanggilan method bisa dilakukan secara bebas kapan saja dan urutannya disesuaikan dengan kebutuhan alur logika program yang sedang dibuat.

5.

The screenshot shows a terminal window within a development environment. The code in the editor is:

```
22
23     Mahasiswa25 mhsRayana = new Mahasiswa25(nm: "Rayana Jaka Surya", nim:
24         mhsRayana.tampilkanInformasi();
25     }
26
27 }
```

The terminal tab is selected at the top, showing the command and its output:

```
PS E:\GitHub\PraktikumASD\PraktikumASD> & 'C:\Program Files\Java\jdk-24
\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users
\putri\AppData\Roaming\Code\User\workspaceStorage\ef6f5af3a5234de4991393
4daf14dd98\redhat.java\jdt_ws\PraktikumASD_f441bebd\bin' 'Jobsheet2.Maha
siswaMain25'
Kelas TI 2L
IPK 3.3
Nama Rayana Jaka Surya
NIM 254107020026
Kelas TI 1C
IPK 3.73
PS E:\GitHub\PraktikumASD\PraktikumASD>
```

On the right side of the terminal, there are two small icons: one labeled "pwsh" and another labeled "Run: Maha".