

Nama : Rofiq Aristiyawan  
NIM : 254107020060  
Kelas/Absen : TI-1G/27

### Percobaan 1

1. Atribut dan Method
2. Terdapat 4, yaitu :
  - a. Nama : String
  - b. nim : String
  - c. kelas : String
  - d. ipk : double
3. Terdapat 4, yaitu :
  - a. tampilInformasi() ; void
  - b. ubahKelas(String kelasBaru) ; void
  - c. updateIpk(ipkBaru) ; void
  - d. nilaiKinerja() ; String
4. Modifikasi:

```
void updateIpk(double ipkBaru) {  
    if (ipkBaru <= 4 && ipkBaru >= 0) {  
        ipk = ipkBaru;  
    } else {  
        System.out.println("IPK tidak calid. Harus antara 0.0 dan 4.0");  
    }  
}
```

5. Pada method nilaiKinerja, hasil dari return ditentukan oleh nilai dari variable ipk yang menggunakan ketentuan percabangan (if). Penjelasan, jika ipk lebih besar atau sama dengan 3,5 maka akan mengembalikan nilai "Kinerja sangat baik", jika ipk lebih besar atau sama dengan 3,0 tetapi dibawah 3,5 maka akan mengembalikan nilai "Kinerja baik", jika ipk lebih besar atau sama dengan 2,0 tetapi dibawah 3,0 maka akan mengembalikan nilai "Kinerja cukup", lalu jika ipk dibawah 2,0 maka akan mengembalikan nilai "kinerja kurang".

### Percobaan 2

1. Baris kode yg digunakan yaitu:

```
Mahasiswa27 mhs1 = new Mahasiswa27();
```

Untuk nama object dari instansiasi adalah mhs1.

2. Cara nya dengan menggunakan operator titik (.) setelah nama objek contohnya:

```
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();
```

nama object nya mhs1 lalu operator titik dan setelahnya bisa atribut atau method sesuai yg dibutuhkan

3. Dikarenakan pada kode tersebut sudah menjalankan method ubahKelas dan updatelpk yang membuat nilai dari atribut berubah ketika dipanggil pada method tampilkanInformasi yg kedua.

### Percobaan 3

1. Kode program yg digunakan untuk mendeklarasikan konstruktor berparameter yaitu:

```
public Mahasiswa27(String nm, String nim, double ipk, String kls) {  
    nama = nm;  
    this.nim = nim;  
    this.ipk = ipk;  
    kelas = kls;  
}
```

2. Pada kode tersebut melakukan instansiasi dan langsung memberikan nilai pada atribut menggunakan konstruktor
3. Terjadi error, dikarenakan di java kalau tidak ada konstruktor sama sekali, java otomatis bikin konstruktor default kosong, tapi kalau sudah membuat minimal 1 konstruktor sendiri, java tidak lagi membuat konstruktor default otomatis.
4. Tidak harus berurutan, karena method itu berdiri sendiri dan eksekusi nya bergantung pemanggilan.
5. Hasil

```
Mahasiswa27 mhsRofiq = new Mahasiswa27("Rofiq Aristiyawan",  
"254107020060", 3.5, "TI 1G");  
mhsRofiq.updateIpk(3.6);  
mhsRofiq.tampilkanInformasi();
```