

Nama : Muhammad Hikmal Al-Ghifary

Kelas : TI – 1B

Matkul : Praktikum Algoritma dan Struktur Data

---



# OBJECT

## Percobaan 1: Deklarasi Class, Atribut dan Method

1. Buat folder baru bernama Jobsheet2 di dalam repository Praktikum ASD

 Jobsheet 2 Create ASD Pertemuan 2 17 minutes ago

2. Buka text editor. Buat file baru, beri nama Mahasiswa.java

 Mahasiswa15.java 


3. Lengkapi class Mahasiswa dengan atribut yang telah digambarkan di dalam class diagram tersebut

```
public class Mahasiswa15 {  
    String nama;  
    String nim;  
    String kelas;  
    double ipk;  
}
```

4. Lengkapi class Mahasiswa dengan method yang telah digambarkan di dalam class diagram tersebut

```
7 void tampilkanInformasi () {  
8     System.out.println("Nama: " + nama);  
9     System.out.println("NIM: " + nim);  
10    System.out.println("IPK: " + ipk);  
11    System.out.println("Kelas: " + kelas);  
12 }  
13  
14 void ubahKelas (String kelasBaru) {  
15     kelas = kelasBaru;  
16 }  
17  
18 void updateIpk (double ipkBaru) {  
19     ipk = ipkBaru;  
20 }  
21  
22 String nilaiKerja () {  
23     if (ipk >= 3.5) {  
24         return "Kinerja sangat baik";  
25     } else if (ipk >= 3.0) {  
26         return "Kinerja baik";  
27     } else if (ipk >= 2.0) {  
28         return "Kinerja cukup";  
29     } else {  
30         return "Kinerja kurang";  
31     }  
32 }
```

5. Compile dan run program.

 Error: Main method not found in the file, please define  
the main method as: public static void main(String[] args)  
  
Source: Debugger for Java

### Pertanyaan

1. Karakteristik object: *Mempunyai sesuatu* dan *Bisa melakukan sesuatu*.
2. Atribut pada class mahasiswa adalah nama, nim, ipk, dan kelas
3. Class tersebut memiliki method sebanyak 4 buah
  - a. Tampilkan informasi

- b. Ubah kelas
- c. Update IPK
- d. Nilai kerja

4. Modifikasi kode

```
18     void updateIpk (double ipkBaru) {  
19         if (ipkBaru >= 0.0 && ipkBaru <= 4.0) {  
20             ipk = ipkBaru;  
21         } else {  
22             ipk = ipkBaru;  
23             System.out.println(x:"IPK tidak valid. Harus antara 0.0 dan 4.0");  
24         }  
25     }
```

5. Cara nilai kerja memberikan input adalah dengan melihat nilai masukan ipk yang didapat dari luar method nilaiKerja().Setelah nilai didapat, program akan melakukan pengecekan kondisi melalui if. Apabila nilai yang dimasukkan lebih besar dari sma dengan 3.5, maka akan menampilkan “Kinerja sangat baik”. Begitupun seterusnya hingga kondisi terakhir.

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

1. Buat file baru, beri nama MahasiswaMain<absen>.java

J MahasiswaMain15.java X

2. Tuliskan struktur dasar bahasa pemrograman Java yang terdiri dari fungsi main()
3. Di dalam fungsi main(), lakukan instansiasi, kemudian lanjutkan dengan mengakses atribut dan method dari objek yang telah terbentuk.

```
1 public class MahasiswaMain15 {
    Run | Debug
2     public static void main(String[] args) {
3         Mahasiswa15 mhs1 = new Mahasiswa15 ();
4         mhs1.nama = "Muhammad Ali Farhan";
5         mhs1.nim = "2241720171";
6         mhs1.kelas = "SI 2J";
7         mhs1.ipk = 3.55;
8
9         mhs1.tampilkanInformasi();
10        mhs1.ubahKelas(kelasBaru:"SI 2K");
11        mhs1.updateIpk(ipkBaru:3.60);
12        mhs1.tampilkanInformasi();
13    }
14 }
```

4. Compile dan run program.

```
Nama: Muhammad Ali Farhan
NIM: 2241720171
IPK: 3.55
Kelas: SI 2J
Nama: Muhammad Ali Farhan
NIM: 2241720171
IPK: 3.6
Kelas: SI 2K
```

### Pertanyaan

1. Baris kode program yang digunakan untuk proses instansiasi

```
3         Mahasiswa15 mhs1 = new Mahasiswa15 ();
```

Objek yang dihasilkan adalah mhs1

2. Cara mengakses atribut dan method dari suatu objek

Atribut:

Mengetikkan *namaObjek.atribut* Contoh:

Untuk mengakses atribut nama mahasiswa ke 1, ketik:

```
4         mhs1.nama = "Muhammad Ali Farhan";
```

Method:

Mengetikkan *namaObjek.method()* Contoh:

Untuk mengakses method tampilkanInformasi(), ketik:

```
9         mhs1.tampilkanInformasi();
```

3. Perbedaan tampilan tampilkanInformasi() pertama dan kedua dikarenakan telah ada perubahan nilai kelas dan ipk. Perubahan tersebut adalah:

```
10        mhs1.ubahKelas(kelasBaru:"SI 2K");
11        mhs1.updateIpk(ipkBaru:3.60);
```

## Percobaan 3: Membuat Konstruktor

1. Buka kembali class Mahasiswa. Tambahkan dua buah konstruktor di dalam class Mahasiswa tersebut, yang terdiri dari satu konstruktor default dan satu konstruktor berparameter. Konstruktor merupakan method istimewa, penempatan kode program untuk konstruktor dapat diperlakukan sama seperti method yang lain (setelah atribut).

```
38     public Mahasiswa15 () {
39     }
40
41     public Mahasiswa15 (String nm, String nim, double ipk, String kls) {
42         nama = nm;
43         this.nim = nim; // karena variabel dari parameter sama dengan nama atribut, jadi ditambahkan (.this)
44         this.ipk = ipk; // karena variabel dari parameter sama dengan nama atribut, jadi ditambahkan (.this)
45         kelas = kls;
46     }
```

2. Buka kembali class MahasiswaMain. Buat sebuah object lagi bernama mhs2 dengan menggunakan konstruktor berparameter.

```
14     Mahasiswa15 mhs2 = new Mahasiswa15(nm:"Annisa Nabila",nim:"2141720160", ipk:3.25, kls:"TI 2L");
15     mhs2.updateIpk(ipkBaru:3.30);
16     mhs2.tampilkanInformasi();
```

3. Compile dan run program.

```
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
```

### Pertanyaan

1. Baris kode program yang digunakan untuk mendeklarasikan konstruktor berparameter

```
41     public Mahasiswa15 (String nm, String nim, double ipk, String kls) {
42         nama = nm;
43         this.nim = nim; // karena variabel dari parameter sama dengan nama atribut, jadi ditambahkan (.this)
44         this.ipk = ipk; // karena variabel dari parameter sama dengan nama atribut, jadi ditambahkan (.this)
45         kelas = kls;
46     }
```

2. Pada baris

```
16     Mahasiswa15 mhs2 = new Mahasiswa15(nm:"Annisa Nabila",nim:"2141720160", ipk:3.25, kls:"TI 2L");
```

Dilakukan instansiasi objek baru bernama mhs2. Instansiasi berparameter sama saja seperti mengisi variabel / deklarasi secara satu per satu. Jika dilakukan secara per variabel, maka menggunakan perintah *mhs2.namaAtribut*

3. Jika metode public Mahasiswa15() dihilangkan dari class Mahasiswa15, maka konstruktor default yang tidak menerima parameter tidak akan tersedia lagi. Alhasil, instansiasi mhs1 harus dilakukan dengan menggunakan parameter dan tidak bisa dilakukan secara satu per satu.
4. Method di dalam class Mahasiswa tidak harus dipanggil secara berurutan. Kita bisa memanggilnya secara acak, asalkan sudah diinstansiasi dengan benar.
5. Modifikasi kode program

```
21     Mahasiswa15 mhsHikmal = new Mahasiswa15(nm:"Muhammad Hikmal",nim:"244107020074",ipk:3.78,cls:"TI 1B");
22     mhsHikmal.tampilkanInformasi();
23     mhsHikmal.updateIpk(ipkBaru:5.0);
24     mhsHikmal.tampilkanInformasi();
```