

ALGORITMA DAN STRUKTUR DATA

“Laporan hasil Praktikum Algoritma dan Struktur Data Tentang Object Pada Jobsheet 2”

Oleh:

Hafiz Rizqi Hernanda_TI_1B

NIM (244107020154)



Jurusan Teknologi informasi

Teknik Informatika

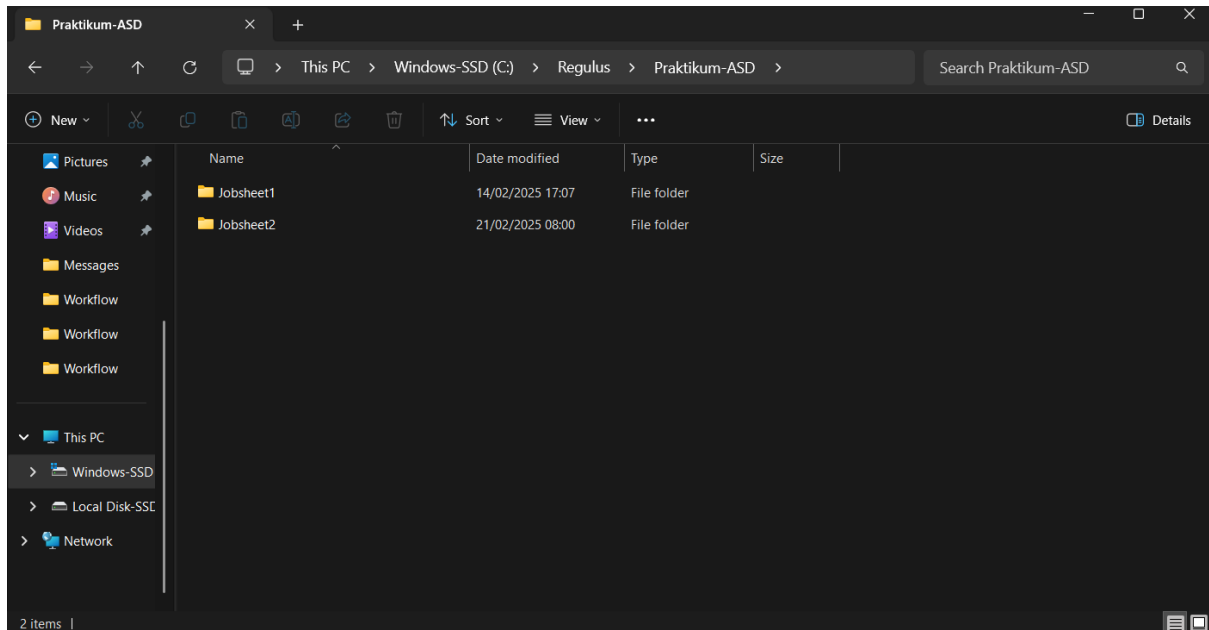
Politeknik Negeri Malang

2. Praktikum

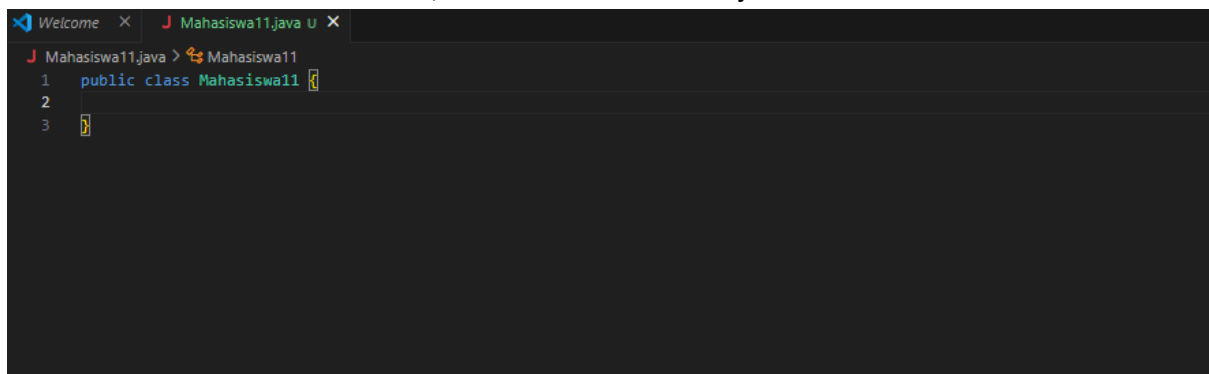
2.1 Percobaan 1: Deklarasi Class, Atribut dan Method

2.1.1 Langkah-langkah Percobaan

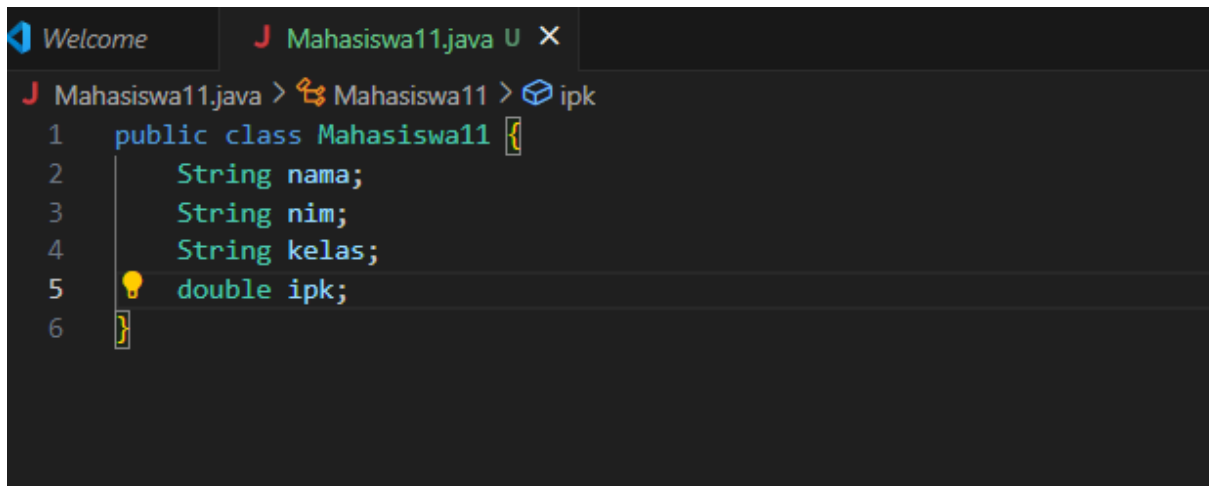
1. Buat folder baru bernama Jobsheet2 di dalam repository Praktikum ASD



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




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



The screenshot shows an IDE with a tab for 'Mahasiswa11.java'. The code defines a public class 'Mahasiswa11' with four attributes: 'nama' (String), 'nim' (String), 'kelas' (String), and 'ipk' (double). The 'ipk' attribute has a yellow lightbulb icon next to it, indicating a warning or suggestion.

```
1 public class Mahasiswa11 {  
2     String nama;  
3     String nim;  
4     String kelas;  
5     double ipk;  
6 }
```

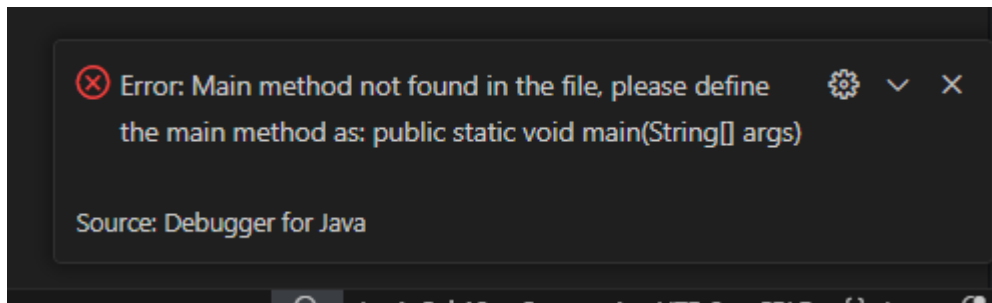
4. Lengkapi class Mahasiswa dengan method yang telah digambarkan di dalam class diagram tersebut.



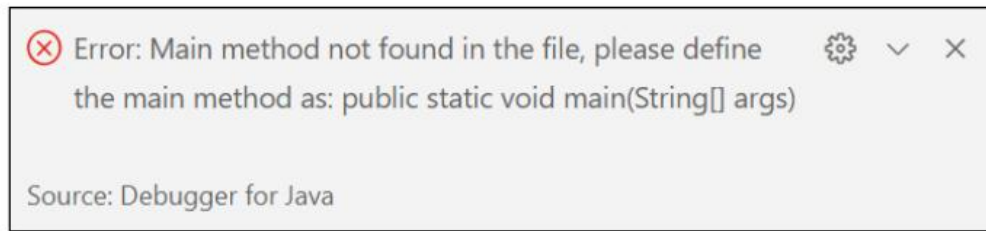
The screenshot shows the continuation of the 'Mahasiswa11' class. It includes four methods: 'tampilkanInformasi()' which prints the student's details; 'ubahKelas()' which updates the 'kelas' attribute; 'updateIPK()' which updates the 'ipk' attribute; and 'nilaiKinerja()' which evaluates the 'ipk' value and returns a performance rating. The 'nilaiKinerja()' method has a yellow lightbulb icon next to the final 'return' statement.

```
6  
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    void updateIPK(double ipkBaru) {  
18        ipk = ipkBaru;  
19    }  
20  
21    String nilaiKinerja() {  
22        if (ipk >= 3.5 ) {  
23            return "Kinerja sangat Baik";  
24        }  
25        else if (ipk >= 3.0 ) {  
26            return "Kinerja Baik";  
27        }  
28        else if (ipk >= 2.0 ) {  
29            return "Kinerja cukup";  
30        }  
31        else {  
32            return "Kinerja Kurang";  
33        }  
34    }  
35 }
```

5. Compile dan run program



2.1.2 Verifikasi Hasil Percobaan



2.1.3 Pertanyaan

1. Sebutkan dua karakteristik class atau object!

Atribut dan Method

2. Perhatikan class Mahasiswa pada Praktikum 1 tersebut, ada berapa atribut yang dimiliki oleh class Mahasiswa? Sebutkan apa saja atributnya!

String nama;

String nim;

String kelas;

double ipk;

3. Ada berapa method yang dimiliki oleh class tersebut? Sebutkan apa saja methodnya!

tampilkanInformasi()

ubahKelas()

updateIPK()

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) {  
    ipk = ipkBaru;  
    if (ipk < 0.0 && ipk > 4.0) {  
        System.out.println(x:"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 nilaiKinerja() berguna dalam mengevaluasi kinerja mahasiswa dengan membandingkan ipk dalam batasan yang di tentukan dalam kode program dituliskan:

Jika IPK ≥ 3.5 akan mengembalikan (return) "Kinerja sangat baik"

Jika IPK ≥ 3.0 akan mengembalikan (return) "Kinerja baik"

Jika IPK ≥ 2.0 akan mengembalikan (return) "Kinerja cukup"

Jika IPK < 2.0 akan mengembalikan (return) "Kinerja kurang"

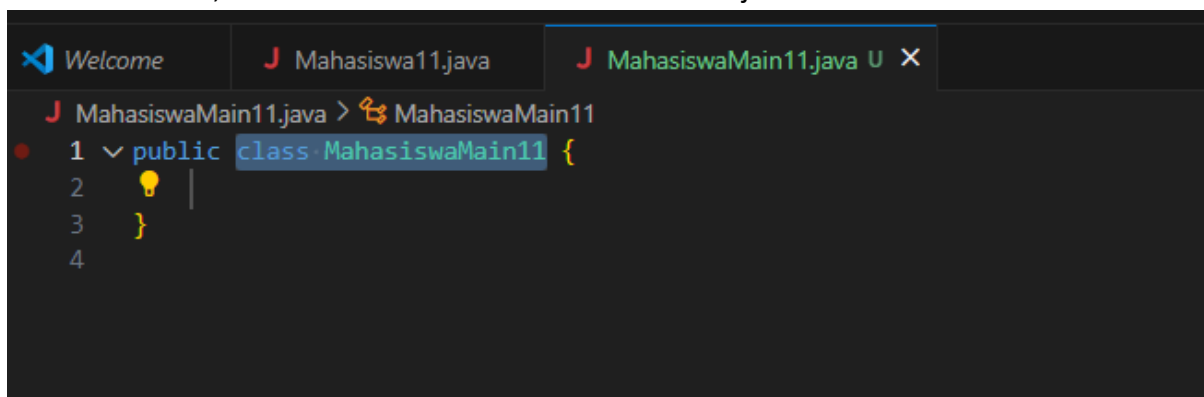
6. Commit dan push kode program ke Github

Name	Last commit message	Last commit date
..		
Mahasiswa11.java	Mahasiswa11 (Percobaan 1)	34 minutes ago

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

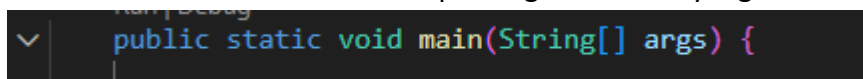
2.2.1 Langkah-langkah Percobaan

1. Buat file baru, beri nama MahasiswaMain<no.Absen>.java



```
1 public class MahasiswaMain11 {  
2  
3 }  
4
```

2. Tuliskan struktur dasar bahasa pemrograman Java yang terdiri dari fungsi main()



```
public static void main(String[] args) {
```

3. Di dalam fungsi main(), lakukan instansiasi, kemudian lanjutkan dengan mengakses atribut dan method dari objek yang telah terbentuk.

```

2     public static void main(String[] args) {
3
4         Mahasiswa11 mhs1 = new Mahasiswa11();
5         mhs1.nama = "Muhammad Ali farhan";
6         mhs1.kelas = "SI 2J";
7         mhs1.kelas = "SI 2J";
8         mhs1.ipk = 3.55;
9
10        mhs1.tampilkanInformasi();
11        mhs1.ubahKelas(kelasBaru:"SI 2K");
12        mhs1.updateIPK(ipkBaru:3.60);
13        mhs1.tampilkanInformasi();
14    }
15 }

```


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
PS C:\Regulus\Praktikum-ASD\Jobsheet2>

```

5. Commit dan push kode program ke Github

<div>  HafizRizqi MahasiswaMain11 (percobaan 2) e8cc79a · 11 minutes ago History </div>		
Name	Last commit message	Last commit date
..		
<div> Mahasiswa11.java </div>	Mahasiswa11 (Percobaan 1)	34 minutes ago
<div> MahasiswaMain11.java </div>	MahasiswaMain11 (percobaan 2)	11 minutes ago

2.2.2 Verifikasi Hasil Percobaan

```

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

```

2.2.3 Pertanyaan

1. Pada class MahasiswaMain, tunjukkan baris kode program yang digunakan untuk proses instansiasi! Apa nama object yang dihasilkan?

Proses instansiasi nya mahasiswa11 mhs1 = new mahasiswa11()

Object yang di hasilkan mhs1

2. Bagaimana cara mengakses atribut dan method dari suatu objek?

Cara mengakses atribut dan method dari suatu object dengan menggunakan tanda titik (.) seperti halnya di kode program :

mhs1. nama

mhs1.nim

mhs1.kelas

mhs1.ipk

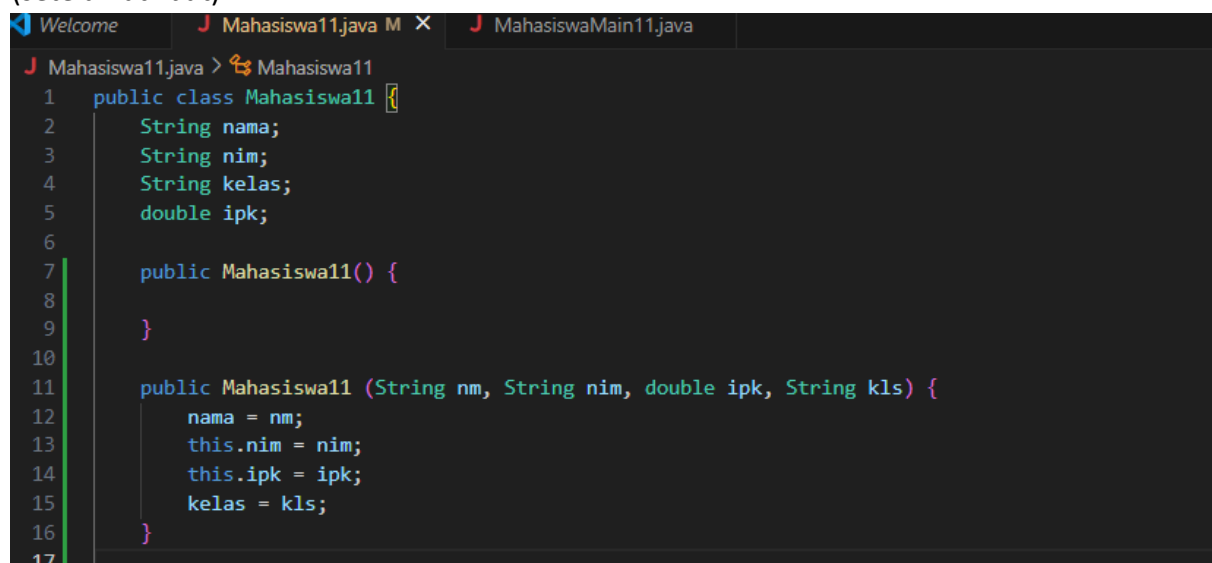
3. Mengapa hasil output pemanggilan method tampilkanInformasi() pertama dan kedua berbeda?

Pada pemanggilan method tampilkanInformasi() pertama akan memanggil data awal yang sudah ada dan pada pemanggilan method tampilkanInformasi() kedua memanggil perubahan data yang ada pada method ubahKelas() dan updateIPK()

2.3 Percobaan 3: Membuat Konstruktor

2.3.1 Langkah-langkah Percobaan

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).



```
1 public class Mahasiswa11 {
2     String nama;
3     String nim;
4     String kelas;
5     double ipk;
6
7     public Mahasiswa11() {
8
9     }
10
11     public Mahasiswa11 (String nm, String nim, double ipk, String kls) {
12         nama = nm;
13         this.nim = nim;
14         this.ipk = ipk;
15         kelas = kls;
16     }
17 }
```

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

```
Mahasiswa11 mhs2 = new Mahasiswa11(nm:"Annisa Nabila", nim:"2141720160", ipk:3.25, kls:"TI 2L");  
mhs2.updateIPK(ipkBaru:3.30);  
mhs2.tampilkanInformasi();
```

3. Compile dan run program.

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  
  
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  
PS C:\Regulus\Praktikum-ASD\Jobsheet2>
```

4. Commit dan push kode program ke Github

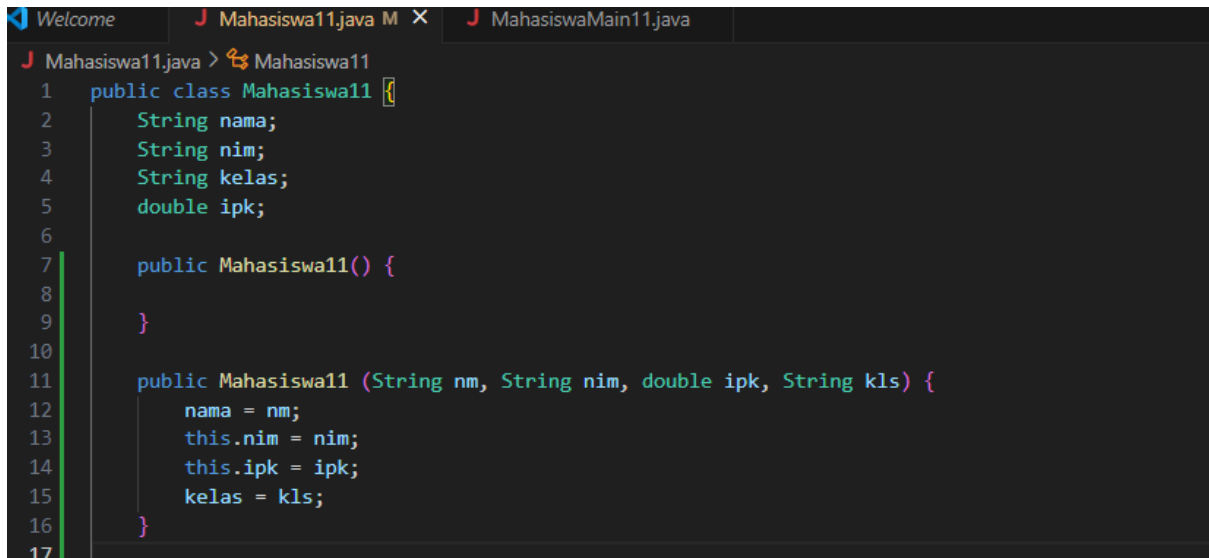
Name	Last commit message	Last commit date
..		
Mahasiswa11.java	Percobaan 3	5 minutes ago
MahasiswaMain11.java	Percobaan 3	5 minutes ago

2.3.2 Verifikasi Hasil Percobaan

```
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
```

2.3.3 Pertanyaan

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



```
1 public class Mahasiswa11 {
2     String nama;
3     String nim;
4     String kelas;
5     double ipk;
6
7     public Mahasiswa11() {
8
9     }
10
11    public Mahasiswa11 (String nm, String nim, double ipk, String kls) {
12        nama = nm;
13        this.nim = nim;
14        this.ipk = ipk;
15        kelas = kls;
16    }
17 }
```

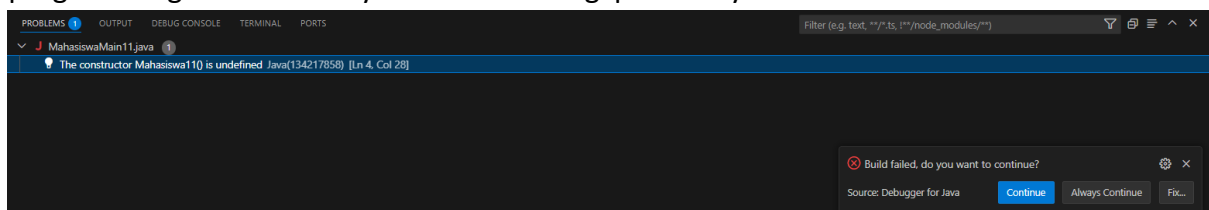
Pada baris 11 sampai baris 16

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

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

? Baris kode ini membuat sebuah object mhs2 dengan konstruksi berparameter.

3. Hapus konstruktor default pada class Mahasiswa, kemudian compile dan run program. Bagaimana hasilnya? Jelaskan mengapa hasilnya demikian!

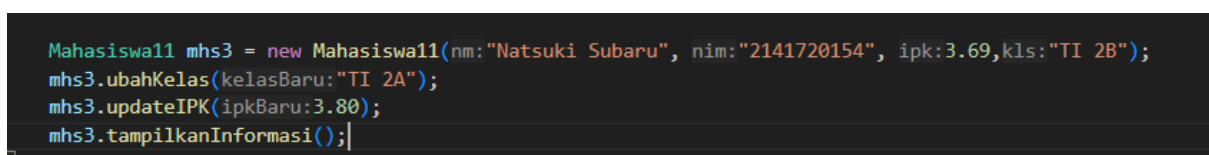


Mengapa mengalami error? Karena di saat kita menghapus konstruktor default yang tersisa hanyalah konstruktor berparameter.. di dalam class MahasiswaMain11 object mhs1 tanpa menggunakan parameter.

4. Setelah melakukan instansiasi object, apakah method di dalam class Mahasiswa harus diakses secara berurutan? Jelaskan alasannya!

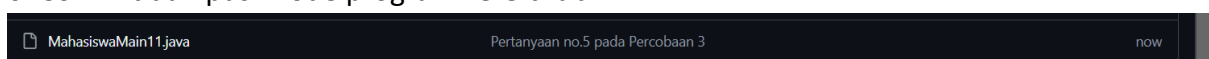
Tidak, Karena Method itu bisa di panggil kapan saja setelah object dibuat dan

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



```
Mahasiswa11 mhs3 = new Mahasiswa11(nm:"Natsuki Subaru", nim:"2141720154", ipk:3.69, kls:"TI 2B");
mhs3.ubahKelas(kelasBaru:"TI 2A");
mhs3.updateIPK(ipkBaru:3.80);
mhs3.tampilkanInformasi();
```

6. Commit dan push kode program ke Github



Latihan Praktikum

1. Diberikan class diagram dari class MataKuliah sebagai berikut:

MataKuliah
kodeMK: String nama: String sks: int jumlahJam: int
tampilInformasi(): void ubahSKS(sksBaru: int): void tambahJam(jam: int): void kurangiJam(jam: int): void

Buat program untuk mengimplementasikan class MataKuliah berdasarkan class diagram di atas, yang terdiri dari:

- Class MataKuliah (MataKuliah.java)
- Class MataKuliahMain (MataKuliahMain.java) Pada class MataKuliahMain buatlah minimal 2 objek.

Gunakan konstruktor default dan konstruktor berparameter saat mengintansiasi objek.

Lalu panggil semua method yang sudah dibuat pada class MataKuliah.

Penjelasan dari atribut dan method pada class MataKuliah tersebut adalah sebagai berikut:

a. Atribut

- kodeMK (String): kode unik untuk mata kuliah.
- nama (String): nama lengkap dari mata kuliah
- sks (int): SKS (Satuan Kredit Semester)
- jumlahJam (int): jumlah total jam pertemuan per minggu untuk mata kuliah

b. Method

- tampilInformasi():

method ini digunakan untuk menampilkan semua informasi yang berkaitan dengan mata kuliah.

- ubahSKS(int sksBaru): method ini memungkinkan pengubahan nilai SKS untuk mata kuliah. Setelah mengubah nilai, method ini memberi tahu pengguna bahwa SKS telah diubah.
- tambahJam(int jam): method ini menambahkan jumlah jam tambahan ke jumlah jam yang sudah ada untuk mata kuliah.
- kurangiJam(int jam): method ini berfungsi untuk mengurangi jumlah jam dari mata kuliah. Sebelum mengurangi, method ini melakukan pengecekan untuk memastikan bahwa jumlah jam yang tersisa cukup untuk dikurangi. Jika jumlah jam tidak mencukupi (jumlah jam awal lebih kecil dari jam pengurang), method ini akan memberi tahu pengguna bahwa pengurangan tidak dapat dilakukan. Jika pengurangan berhasil, method ini mengupdate jumlah jam dan mencetak nilai jumlah jam yang baru.

```

J MataKuliah11.java > MataKuliah11 > kurangiJam(int)
1  public class MataKuliah11 {
2      String kodeMK;
3      String nama;
4      int sks;
5      int jumlahJam;
6
7      public MataKuliah11() {
8
9      }
10
11     public MataKuliah11(String kdMK, String nm, int sks, int jmlhJam){
12         kodeMK = kdMK;
13         nama = nm;
14         this.sks = sks;
15         jumlahJam = jmlhJam;
16     }
17     void tampilkanInformasi() {
18         System.out.println("Kode MK: " + kodeMK);
19         System.out.println("Nama MataKuliah: " + nama );
20         System.out.println("SKS: " + sks);
21         System.out.println("total jam pertemuan per minggu: " + jumlahJam);
22     }
23
24     void ubahSKS(int sksBaru) {
25         sks = sksBaru;
26     }
27     void tambahJam (int jam) {
28         jumlahJam = jumlahJam + jam;
29     }
30     void kurangiJam (int jam) {
31         if (jumlahJam < jam ) {
32             System.out.println(x:"Pengurangan tidak dapat di lakukan. ");
33         }
34         else {
35             jumlahJam = jumlahJam - jam;
36         }
37     }
38 }
39
40
41

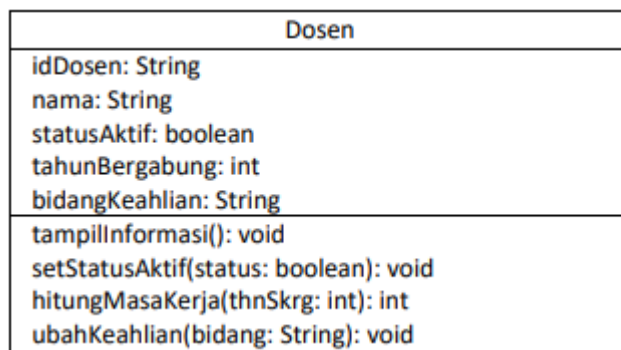
```

```

J MataKuliahMain11.java > ...
1  public class MataKuliahMain11 {
2      Run | Debug
3      public static void main(String[] args) {
4          MataKuliah11 MK1 = new MataKuliah11();
5          MK1.kodeMK = "1234";
6          MK1.nama = "Data Struktur";
7          MK1.sks = 2;
8          MK1.jumlahJam = 4;
9
10         MK1.tampilkanInformasi();
11         MK1.ubahSKS(sksBaru:6);
12         MK1.tambahJam(jam:8);
13         MK1.tampilkanInformasi();
14
15         MataKuliah11 MK2 = new MataKuliah11(kdMK:"9876", nm:"kalkulus", sks:4, jmlhJam:8);
16         MK2.tampilkanInformasi();
17         MK2.ubahSKS(sksBaru:3);
18         MK2.kurangiJam(jam:2);
19         MK2.tampilkanInformasi();
20     }
21 }
22

```

2. Diberikan class diagram dari class Dosen sebagai berikut:



Buat program untuk mengimplementasikan class Dosen berdasarkan class diagram di atas, yang terdiri dari:

- Class Dosen (Dosen.java)
- Class DosenMain (DosenMain.java) Pada class DosenMain buatlah minimal 2 objek.

Gunakan konstruktor default dan konstruktor berparameter saat mengintansiasi objek.

Lalu panggil semua method yang sudah dibuat pada class Dosen Penjelasan dari atribut dan method pada class Dosen tersebut adalah sebagai berikut:

a. Atribut

- idDosen (String): id unik untuk setiap dosen.
- nama (String): nama lengkap dari dosen.
- statusAktif (boolean): menunjukkan apakah dosen tersebut aktif (true) atau tidak aktif (false) dalam menjalankan tugasnya.
- tahunBergabung (int): tahun ketika dosen mulai bergabung dengan perguruan tinggi
- bidangKeahlian (String): bidang keahlian dosen, yang menjelaskan spesialisasi atau fokus akademik dosen

b. Method

- tampilInformasi(): method ini digunakan untuk menampilkan informasi lengkap tentang dosen
- setStatusAktif(status: boolean): method ini digunakan untuk mengatur status aktif dosen. Jika parameter status diatur ke true, berarti dosen tersebut menjadi aktif. Sebaliknya, jika false, dosen tersebut dinyatakan tidak aktif.
- hitungMasaKerja(thnSkrng: int): method ini menghitung dan mengembalikan (me-returnkan) masa kerja dosen dalam tahun, berdasarkan tahun bergabung dan tahun saat ini (thnSkrng) yang menjadi parameter input method ini. Hasil perhitungan memberikan informasi tentang lamanya dosen bekerja di perguruan tinggi.
- ubahKeahlian(bidang: String): method ini digunakan untuk mengubah bidang keahlian dosen.

J Dosen11.java > Dosen11 > setStatusAktif(boolean)

```
1 public class Dosen11 {
2     String idDosen;
3     String nama;
4     boolean statusAktif;
5     int tahunBergabung;
6     String bidangKeahlian;
7
8     public Dosen11(){
9
10    }
11    public Dosen11 (String id, String nm, boolean status, int thnBergabung, String bdngKeahlian) {
12        idDosen = id;
13        nama = nm;
14        statusAktif = status;
15        tahunBergabung = thnBergabung;
16        bidangKeahlian = bdngKeahlian;
17    }
18
19    void tampilInfomasi() {
20        System.out.println("ID Dosen: " + idDosen);
21        System.out.println("Nama Lengkap: " + nama);
22        System.out.println("Status Keaktifan: " + statusAktif);
23        System.out.println("Kapan tahun Bergabung: " + tahunBergabung);
24        System.out.println("Keahliannya: " + bidangKeahlian);
25    }
26    void setStatusAktif(boolean status) {
27        statusAktif = status;
28    }
29    int hitungMasaKerja(int thnSkrg) {
30        return thnSkrg - tahunBergabung;
31    }
32    void ubahKeahlian(String bidang){
33        bidangKeahlian = bidang;
34    }
35 }
36
37
```

J DosenMain11.java > DosenMain11 > main(String[])

```
1 public class DosenMain11 {
2     Run | Debug
3     public static void main(String[] args) {
4         Dosen11 dsn1 = new Dosen11();
5         dsn1.idDosen = "1234";
6         dsn1.nama = "Arif Syamsudin";
7         dsn1.statusAktif = true;
8         dsn1.tahunBergabung = 2015;
9         dsn1.bidangKeahlian = "Chemical Science";
10
11         dsn1.tampilInfomasi();
12         dsn1.setStatusAktif(status:false);
13         System.out.println("Masa Kerja: " + dsn1.hitungMasaKerja(thnSkrg:2025) + " Tahun");
14         dsn1.ubahKeahlian(bidang:"Kuantum Science");
15         dsn1.tampilInfomasi();
16         System.out.println("Masa Kerja: " + dsn1.hitungMasaKerja(thnSkrg:2025) + " Tahun");
17
18         Dosen11 dsn2 = new Dosen11(id:"5678", nm:"Rizky Rachmat", status:true, thnBergabung:2010, bdngKeahlian:"English Speaking British");
19         dsn2.tampilInfomasi();
20         System.out.println("Masa Kerja: " + dsn2.hitungMasaKerja(thnSkrg:2025) + " Tahun");
21         dsn2.ubahKeahlian(bidang:"Interklutural bisnis communication");
22         dsn2.tampilInfomasi();
23         System.out.println("Masa Kerja: " + dsn1.hitungMasaKerja(thnSkrg:2025) + " Tahun");
24
25    }
26 }
```