

**LAPORAN PRAKTIKUM**  
**PEMROGRAMAN BERORIENTASI OBJEK**  
**MODUL II**  
**CLASS dan OBJECT**



**Disusun Oleh:**

**Mohamad Dandung Sadat 105222033**

**PROGRAM STUDI ILMU KOMPUTER**  
**FAKULTAS SAINS DAN KOMPUTER**  
**UNIVERSITAS PERTAMINA**  
**2025**

## I. Pendahuluan

Sistem informasi akademik merupakan bagian integral dari proses administrasi dan evaluasi dalam lingkungan perguruan tinggi. Salah satu komponen penting dari sistem ini adalah pengelolaan dan evaluasi nilai mahasiswa yang mencerminkan tingkat pencapaian belajar selama menempuh pendidikan. Penilaian akademik tidak hanya diperlukan untuk menentukan kelulusan suatu mata kuliah, tetapi juga menjadi dasar dalam pengambilan keputusan yang lebih luas, seperti beasiswa, pemantauan prestasi, hingga strategi perbaikan belajar. Dalam praktiknya, proses ini sering kali dilakukan secara manual atau melalui sistem yang kompleks, sehingga dibutuhkan solusi digital sederhana yang dapat membantu mahasiswa maupun dosen dalam menilai performa akademik secara efisien. Program **Sistem Nilai Mahasiswa** yang dikembangkan menggunakan bahasa pemrograman Java ini bertujuan untuk menjawab kebutuhan tersebut dengan pendekatan yang praktis dan interaktif. Melalui antarmuka berbasis teks, program memungkinkan pengguna (dalam hal ini mahasiswa atau staf akademik) untuk memasukkan informasi dasar seperti nama, NIM, usia, jumlah mata kuliah yang diambil, serta nilai dari masing-masing mata kuliah. Sistem kemudian menghitung rata-rata nilai dan mengonversinya ke dalam skala IPK 4.00, yang merupakan standar umum dalam sistem pendidikan tinggi di Indonesia. Tidak hanya itu, program ini juga mengklasifikasikan status akademik mahasiswa ke dalam beberapa kategori, seperti "Sangat Memuaskan", "Memuaskan", "Cukup Memuaskan", dan "Perlu Perbaikan" berdasarkan nilai IPK yang diperoleh.

## II. Variabel

No	Nama Variabel	Tipe data	Fungsi
1	<i>Input</i>	scanner	Objek untuk menerima input dari pengguna.
2	Nama	String	Menyimpan nama mahasiswa.
3	Nim	String	Menyimpan Nomor Induk Mahasiswa.
4	Usia	Int	Menyimpan usia mahasiswa dalam tahun.
5	jumlahMataKuliah	Int	Menyimpan jumlah mata kuliah yang diambil mahasiswa.
6	totalNilai	Double	Akumulator untuk menjumlahkan semua nilai mata kuliah.
7	nilaiMataKuliah	Double	Menyimpan nilai untuk setiap mata kuliah yang dimasukkan.
8	rataRataNilai	Double	Menyimpan rata-rata dari semua nilai mata kuliah.
9	Ipk	Double	Menyimpan Indeks Prestasi Kumulatif (IPK) mahasiswa pada skala 4.0.
10	usiaLebihDari22	Boolean	Hasil evaluasi apakah usia mahasiswa lebih dari 22 tahun.
11	kondisiIpkBaikDanMataKuliahCukup	Boolean	Hasil evaluasi logika AND untuk IPK dan jumlah mata kuliah.
12	kondisiIpkKurangAtauMataKuliahSedikit	Boolean	Hasil evaluasi logika OR untuk IPK dan jumlah mata kuliah.
13	nomorAntrianKonsultasi	Int	Nomor antrian acak untuk konsultasi akademik.
14	statusAkademik	string	Status kualitatif performa akademik berdasarkan IPK. Ekspor ke Spreadsheet

### III. Constructor dan Method

No	Nama Metode	Jenis Metode	Fungsi
1	SistemNilaiMahasiswa()	<i>Procedural</i>	(Implicit) Constructor default untuk kelas SistemNilaiMahasiswa . Tidak didefinisikan secara eksplisit.
2	main(String[] args)	Functional (dari Scanner)	Metode utama (entry point) program Java. Mengandung semua logika alur program, mulai dari input hingga output.
3	nextLine()	Functional (dari Scanner)	Membaca seluruh baris input dari konsol sebagai String.
4	nextInt()	Functional (dari Scanner)	Membaca token berikutnya dari konsol sebagai int.
5	nextDouble()	Functional (dari Math)	Membaca token berikutnya dari konsol sebagai double.
6	random()	Functional (dari Math)	Menghasilkan nilai double positif yang lebih besar dari atau sama dengan 0.0 dan kurang dari 1.0.
7	print() / println() / printf()	Procedural (dari System.out)	Menampilkan output ke konsol. printf digunakan untuk format output.
8	close()	Procedural (dari Scanner)	Menutup objek Scanner dan Ekspor ke Spreadsheet

## Dokumentasi dan Pembahasan Code

### 1. Struktur Program dan Input Pengguna

Program utama berada dalam kelas SistemNilaiMahasiswa yang berisi metode main sebagai titik awal eksekusi.

Kelas java.util.Scanner digunakan untuk menerima input dari pengguna.

```
1 package THT;
2
3 import java.util.Scanner;
4 import java.lang.Math;
5
6 public class SistemNilaiMahasiswa {
7     public static void main(String[] args) {
8         Scanner input = new Scanner(System.in);
9     }
```

Program utama (main method) dienkapsulasi dalam kelas SistemNilaiMahasiswa. Import statement untuk java.util.Scanner dan java.lang.Math (meskipun Math tidak perlu diimpor secara eksplisit karena berada di java.lang yang diimpor secara otomatis) diperlukan untuk fungsionalitas input dan matematika. Objek Scanner diinisialisasi untuk membaca input dari konsol (System.in).

### 2. Penerimaan Input Data Mahasiswa:

Program memulai interaksi dengan pengguna (dosen atau mahasiswa) untuk mengumpulkan data dasar mahasiswa. Metode nextLine() digunakan untuk membaca input String (nama dan NIM),

```
public static void main(String[] args) {
    Scanner input = new Scanner(System.in);

    System.out.println(x:"=== SISTEM PENGELOLAAN NILAI MAHASISWA ===");
    System.out.print(s:"Masukkan Nama Mahasiswa: ");
    String nama = input.nextLine();

    System.out.print(s:"Masukkan NIM Mahasiswa: ");
    String nim = input.nextLine();

    System.out.print(s:"Masukkan Usia Mahasiswa: ");
    int usia = input.nextInt();

    System.out.print(s:"Masukkan Jumlah Mata Kuliah yang Diambil: ");
    int jumlahMataKuliah = input.nextInt();
}
```

sedangkan nextInt() digunakan untuk membaca input integer (usia dan jumlah mata kuliah). Penting untuk dicatat penggunaan input.nextLine() setelah nextInt() atau nextDouble() untuk mengonsumsi karakter newline (\n) yang tersisa di buffer input, sehingga nextLine() berikutnya dapat membaca input dengan benar.

### 3. Perhitungan Rata-rata Nilai dan IPK:

Setelah semua nilai mata kuliah terkumpul, program menghitung rata-rata nilai. Sebuah pengecekan if (`jumlahMataKuliah > 0`) dilakukan untuk mencegah `ArithmeticException` (pembagian

```
double totalNilai = 0;
for (int i = 1; i <= jumlahMataKuliah; i++) {
    System.out.print("Masukkan nilai mata kuliah ke-" + i + ": ");
    double nilaiMataKuliah = input.nextDouble();
    totalNilai += nilaiMataKuliah;
}

input.nextLine();

double rataRataNilai = 0;
if (jumlahMataKuliah > 0) {
    rataRataNilai = totalNilai / jumlahMataKuliah;
}

double ipk = (rataRataNilai / 100.0) * 4.0;
```

dengan nol) jika tidak ada mata kuliah yang dimasukkan. Selanjutnya, `rataRataNilai` dikonversi menjadi IPK pada skala 4.0 dengan asumsi nilai maksimal adalah 100. 100.0 digunakan untuk memastikan operasi pembagian menghasilkan double.

### 4. Evaluasi Performa Akademik (Operator Perbandingan & Logika):

Program melakukan serangkaian evaluasi akademik yang hasilnya disimpan dalam variabel

```
System.out.println(x:"\n--- Evaluasi Akademik ---");

boolean usiaLebihDari22 = usia > 22;
System.out.println("Apakah usia mahasiswa lebih dari 22 tahun? " + usiaLebihDari22);

boolean kondisiIpkBaikDanMataKuliahCukup = (ipk >= 3.5) && (jumlahMataKuliah > 4);
System.out.println("Apakah IPK >= 3.5 DAN jumlah mata kuliah > 4? " + kondisiIpkBaikDanMataKuliahCukup);

boolean kondisiIpkKurangAtauMataKuliahSedikit = (ipk < 2.5) || (jumlahMataKuliah < 4);
System.out.println("Apakah IPK < 2.5 ATAU jumlah mata kuliah < 4? " + kondisiIpkKurangAtauMataKuliahSedikit);
```

boolean. Ini mendemonstrasikan penggunaan operator perbandingan (`>`, `>=`, `<`) dan operator logika (`&&` untuk AND, `||` untuk OR).

## 5. Penentuan Nomor Antrian Konsultasi Akademik

Math.random() menghasilkan angka double pseudorandom antara 0.0 (inklusif) dan 1.0

```
int nomorAntrianKonsultasi = (int) (Math.random() * 100) + 1;
```

(eksklusif). Untuk mendapatkan nomor antrian integer dalam rentang spesifik (misalnya 1 hingga 100), hasilnya diskalikan dan digeser.

## 6. Menampilkan Laporan Akademik Mahasiswa:

Semua informasi yang telah dikumpulkan, dihitung, dan dievaluasi ditampilkan dalam format laporan yang terstruktur dan rapi. System.out.printf() digunakan untuk memformat output numerik

```
System.out.println(x:"\n=== LAPORAN AKADEMIK MAHASISWA ===");
System.out.println("Nama: " + nama);
System.out.println("NIM: " + nim);
System.out.println("Usia: " + usia + " tahun");
System.out.println("Jumlah Mata Kuliah Diambil: " + jumlahMataKuliah);
System.out.printf(format:"Rata-rata Nilai: %.2f\n", rataRataNilai);
System.out.printf(format:"IPK: %.2f\n", ipk);
System.out.println("Nomor Antrian Konsultasi: " + nomorAntrianKonsultasi);

String statusAkademik;
if (ipk >= 3.5) {
    statusAkademik = "Sangat Memuaskan";
} else if (ipk >= 3.0) {
    statusAkademik = "Memuaskan";
} else if (ipk >= 2.5) {
    statusAkademik = "Cukup Memuaskan";
} else {
    statusAkademik = "Perlu Perbaikan";
}
System.out.println("Status Akademik: " + statusAkademik);

input.close();
}
```

(rata-rata nilai dan IPK) agar ditampilkan dengan dua angka desimal (%.2f) dan memastikan baris baru (%n). Selain itu, sebuah blok if-else if-else digunakan untuk menentukan statusAkademik kualitatif berdasarkan rentang IPK.

## Kesimpulan

Program **Sistem Nilai Mahasiswa** ini bertujuan untuk mengelola dan mengevaluasi performa akademik mahasiswa berdasarkan input dari pengguna, seperti nama, NIM, usia, jumlah mata kuliah yang diambil, dan nilai-nilai tiap mata kuliah. Program ini menghitung rata-rata nilai dan mengkonversinya ke skala IPK (Indeks Prestasi Kumulatif) 4.00. Berdasarkan IPK dan jumlah mata kuliah, program juga mengevaluasi kondisi akademik mahasiswa melalui operasi logika seperti pengecekan apakah usia lebih dari 22 tahun, apakah IPK tinggi dengan beban mata kuliah yang cukup, atau sebaliknya. Selain itu, program menggunakan fungsi `Math.random()` untuk menghasilkan nomor antrian konsultasi secara acak. Di akhir program, ditampilkan laporan akademik lengkap, termasuk IPK, status akademik (dari “Sangat Memuaskan” hingga “Perlu Perbaikan”), serta hasil evaluasi logika, yang memberikan gambaran menyeluruh tentang performa akademik mahasiswa yang bersangkutan. Program ini sederhana namun mencakup konsep dasar Java seperti perulangan, pengambilan input, konversi nilai, percabangan, dan penggunaan library bawaan.