

Nama : Elsa Marthalinda

NIM : 254107020204

Prodi : D-IV Teknik Informatika

Kelas : 1G

## Percobaan 1

The screenshot shows a Java code editor in VS Code. The code reads an integer from the user and prints it followed by a series of asterisks. The code is as follows:

```
1 package jobsheet8;
2 import java.util.Scanner;
3 public class Star7 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         System.out.print("Masukkan nilai n = ");
8         int n = sc.nextInt();
9
10        for(int i=1; i<n; i++) {
11            System.out.print("*");
12        }
13    }
14 }
```

Below the code, the terminal window shows the execution of the program:

```
PS C:\Users\elif\Pictures\Pemrograman\TUGAS> & 'C:\Program Files\Java\jdk-24\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\elif\AppData\Roaming\Code\User\workspaces\storage\bda0eca37a02f6f4ad7c77def577fc2\redhat.java\jdt_ws\TUGAS_8bf3a14\bin' 'jobsheet8.star7'
Masukkan nilai n = 5
*****
PS C:\Users\elif\Pictures\Pemrograman\TUGAS>
```



1. Jika pada perulangan for, inisialisasi `i=1` diubah menjadi `i=0`, apa akibatnya? Mengapa bisa demikian?
    - Akibatnya jumlah total bintang (\*) yang dicetak akan bertambah satu, menjadi  $n+1$  bintang.Mengapa? Karena perulangan dimulai dari  $i = 0$ . Dengan kondisi akhir `i <= n`, perulangan akan beriterasi untuk nilai  $i = 0, 1, 2, \dots, n$ . Ini adalah  $n + 1$  kali iterasi, satu lebih banyak daripada kode asli yang dimulai dari  $i = 1$

2. Jika pada perulangan for, kondisi `i<=n` diubah menjadi `i<n`, bagaimana bentuk outputnya jika input n = 5? Mengapa hasilnya berbeda?

- Bentuk output jika input n = 5: \*\*\*\* (4 bintang).

Hasilnya berbeda karena  $5 < 5$  adalah false. Akibatnya, perulangan berhenti satu iterasi lebih awal, mencetak satu bintang lagi.

3. Jika pada perulangan for, kondisi `i <= n` diubah menjadi `i > n`, apa akibatnya?

Mengapa bisa demikian?

- Akibatnya tidak ada bintang yang akan dicetak, dan perulangan tidak akan berjalan sama sekali.

Mengapa? Karena kondisi perulangan diuji segera setelah inisialisasi. Dengan inisialisasi  $I = 1$  dan kondisi baru  $I > n$ , jika n (misalnya 5 diinput, maka  $1 > 5$  adalah *false*). Karena kondisi perulangan langsung *false* di awal, perulangan (System.out.print("\*")); tidak pernah dieksekusi.

4. Jika pada perulangan for, kondisi step `i++` diubah menjadi `i--` apa akibatnya? Mengapa bisa demikian?

- Akibatnya program akan mengalami Infinite Loop (perulangan tak terbatas).

Mengapa? Karena Inisialisasi adalah  $I = 1$  dan kondisi tetap  $i \leq n$  (misalnya  $n=5$ ).

Ketika *step* diubah menjadi `i--`, nilai i akan terus berkurang ( $1, 0, -1, -2, \dots$ ).

Karena nilai n akan selalu lebih besar atau sama dengan nilai i yang terus menurun, kondisi  $i \leq n$  akan selalu *true*. Program akan terus mencetak bintang tanpa batas.

5. Jika pada perulangan for, step `i++` diubah menjadi `i += 2`, bagaimana pola outputnya jika input n = 6? Apa yang menyebabkan perubahan tersebut?

- Pola Output input n = 6: \*\*\* (3 bintang).

Penyebab perubahan:  $i+ = 2$  menyebabkan counter i bertambah 2 pada setiap iterasi. Jadi mengurangi jumlah total iterasi karena i "melompati" nilai-nilai ganjil.

## Percobaan 2

The screenshot shows a Java IDE interface with multiple tabs open. The active tab contains the following Java code:

```

1 package jobsheet8;
2 import java.util.Scanner;
3 public class Square7 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         System.out.print("Masukkan nilai N = ");
8         int N = sc.nextInt();
9
10        for (int iOuter=1; iOuter<=N; iOuter++) {
11            for (int i=1; i<=N; i++) {
12                System.out.print("*");
13            }
14            System.out.println();
15        }
16    }
17 }

```

The code prompts the user to enter a value for N and then prints a square pattern of asterisks of size N by N. The output window shows the execution of the program:

```

PS C:\Users\elif\Pictures\Pemrograman\TUGAS> & 'C:\Program Files\Java\jdk-24\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\elif\AppData\Roaming\Code\User\workspaces\storage\ddadeca37a02f6f4d7c77de6f577fc2\redhat.java\jdt_ws\TUGAS_8bf3a14\bin' 'jobsheet8.Square7'
Masukkan nilai N = 5
*****
*****
*****
*****
*****
PS C:\Users\elif\Pictures\Pemrograman\TUGAS>

```

The terminal also displays the commit history from GitHub:

Percobaan 2  
Elsaamarthalinda committed 2 minutes ago

- Perhatikan perulangan luar. Jika pada sintaks **for**, inisialisasi `iOuter=1` diubah menjadi `iOuter=0`, apa yang akibatnya? Mengapa bisa demikian?
  - Akibatnya jumlah total bintang (\*) yang dicetak akan bertambah satu menjadi  $n + 1$  bintang. Mengapa? Karena perulangan akan mulai dari  $i = 0$ . Dengan kondisi  $i \leq n$ , perulangan akan beriterasi untuk nilai  $i = 0, 1, 2, \dots, n$ . Ini menghasilkan total  $n + 1$  iterasi, sedangkan kode asli (dimulai dari  $i = 1$ ) hanya menghasilkan  $n$  iterasi.
- Kembalikan program semula dimana inisialisasi `iOuter=1`. Kemudian perhatikan perulangan dalam. Jika pada sintaks **for**, inisialisasi `i=1` diubah menjadi `i=0`, apa yang akibatnya? Mengapa bisa demikian?
  - Tidak ada akibat apa-apa pada eksekusi program. Mengapa? Karena tidak memiliki perulangan dalam (nested loop); ia hanya memiliki satu perulangan tunggal. Variabel `j` tidak dideklarasikan maupun digunakan dalam kode tersebut.

3. Apakah perbedaan kegunaan antara perulangan luar dengan perulangan yang berada di dalamnya?
  - **Outer loop** digunakan untuk menghitung atau melakukan perulangan sebanyak jumlah baris
  - **Inner loop** digunakan untuk mengatur kolom dalam satu baris. Ia menentukan jumlah elemen yang dicetak di setiap baris yang ditentukan oleh perulangan luar.
4. Mengapa perlu ditambahkan sintaks `System.out.println();` di bawah perulangan dalam? Apa akibatnya jika sintaks tersebut dihilangkan?
  - Perintah `System.out.println();` berfungsi untuk membuat baris baru (newline). Sintaks ini diletakkan di bawah perulangan untuk memastikan bahwa setelah perulangan dalam selesai mencetak semua elemen untuk satu baris, kursor akan pindah ke baris berikutnya sebelum perulangan luar memulai iterasi baris selanjutnya.  
Jika dihilangkan, maka semua elemen yang dicetak oleh perulangan dalam setiap iterasi perulangan luar akan dicetak berturut-turut dalam satu baris horizontal yang sangat panjang.
5. **Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 2”**

The screenshot shows a Java IDE interface with multiple tabs open. The active tab contains the following Java code:

```
jobsheet8 > J Square7.java 1 M > Square7 > main(String[])
1 package jobsheet8;
2 import java.util.Scanner;
3 public class Square7 {
4     Run|Debug
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7
8         System.out.print("Masukkan nilai N = ");
9         int N = sc.nextInt();
10
11        for (int iOuter=1; iOuter<=N; iOuter++) {
12            for (int i=1; i<N; i++) {
13                System.out.print("*");
14            }
15        }
16    }
}
```

The terminal window below shows the execution of the program:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\elif\Pictures\Pemrograman\TUGAS> & 'C:\Program Files\Java\jdk-24\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\elif\AppData\Roaming\Code\User\workspaceStorage\0deca37a02f6f40d7c7de6f577fc2\redhat.java\jdt_ws\TUGAS_8bf3a14\bin' 'jobsheet8.Square7'
Masukkan nilai N = 5
*****
PS C:\Users\elif\Pictures\Pemrograman\TUGAS> [
```

The status bar at the bottom indicates "Java: Ready".

## Modifikasi Percobaan 2

Elsaamarthalinda committed 1 minute ago

## Percobaan 3

The screenshot shows a Java IDE interface with multiple tabs open. The active tab contains the following Java code:

```
jobsheet8 > J Triangle7.java 1 U > Triangle7 > main(String[])
1 package jobsheet8;
2 import java.util.Scanner;
3 public class Triangle7 {
4     Run|Debug
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7
8         System.out.print("Masukkan nilai n = ");
9         int n = sc.nextInt();
10        int i = 0;
11        while (i < n) {
12            int j = 0;
13            while (j < i) {
14                System.out.print("*");
15                j++;
16            }
17            i++;
18        }
19    }
}
```

The terminal window below shows the execution of the program:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\elif\Pictures\Pemrograman\TUGAS> & 'C:\Program Files\Java\jdk-24\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\elif\AppData\Roaming\Code\User\workspaceStorage\0deca37a02f6f40d7c7de6f577fc2\redhat.java\jdt_ws\TUGAS_8bf3a14\bin' 'jobsheet8.Triangle7'
Masukkan nilai n = 5
*****
PS C:\Users\elif\Pictures\Pemrograman\TUGAS> [
```

The status bar at the bottom indicates "Java: Ready".

### Percobaan 3

Elsaamarthalinda committed now

6251575

- Perhatikan, apakah output yang dihasilkan dengan nilai  $n = 5$  sesuai dengan tampilan berikut?

```
*  
**  
***  
****  
*****
```

- Tidak
- Jika tidak sesuai, bagian mana saja yang harus diperbaiki/ditambahkan? Jelaskan setiap bagian yang perlu diperbaiki/ditambahkan.

- Perlu ditambahkan `System.out.println();` di bawah perulangan dalam, agar menghasilkan output yang sesuai.  
Sintaks tersebut berfungsi untuk memindahkan kursor baris baru (newline) setiap kali perulangan luar selesai menjalankan seluruh perulangan dalam.

- Jelaskan peran masing-masing variabel  $i$  dan  $j$  dalam program ini. Mengapa  $j$  di-set ulang ke 0 di awal setiap iterasi outer loop? Apa yang akan terjadi jika  $j$  tidak di-reset?

- **Variabel i:** Berperan sebagai penghitung baris (dimensi vertikal). Nilai  $i$  menentukan sudah berapa baris yang telah dicetak dan seringkali juga menentukan batas atas perulangan dalam (jumlah kolom) untuk pola segitiga.
- **Variabel j:** Berperan sebagai penghitung kolom (dimensi horizontal). Nilai  $j$  menentukan jumlah elemen (\*) yang dicetak dalam satu baris tertentu.

Variabel  $j$  harus di-reset ke nilai awalnya (0) di dalam perulangan luar agar perulangan dalam (`while ( $j < i$ )`) dapat mulai menghitung kolom dari awal (0) untuk setiap baris baru.

Akibat jika  $j$  tidak di-set, maka setelah baris pertama selesai dicetak, nilai  $j$  akan tetap pada nilai akhir iterasi sebelumnya. Pada iterasi luar berikutnya, kondisi (`while ( $j < i$ )`) kemungkinan akan langsung false atau menghasilkan pola yang salah karena  $j$  tidak dimulai dari 0, sehingga baris-baris berikutnya hanya mencetak 0 bintang.

#### 4. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 3”

The screenshot shows the VS Code interface with multiple tabs open at the top: Tugas2.java, Start.java 1, Square.java 1, Triangle7.java (active), cm\_elsa.java 1, SiakadWhile7.java, and Merging: Tugas3SistemPerpustakaan.java 1. The main editor area contains the following Java code for Triangle7.java:

```
1 package jobsheet;
2 import java.util.Scanner;
3 public class Triangle7 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         System.out.print("Masukkan nilai n = ");
8         int n = sc.nextInt();
9         int i = 0;
10        while (i <= n) {
11            int j = 0;
12            while (j < i) {
13                System.out.print("*");
14                j++;
15            }
16            System.out.println();
17            i++;
18        }
19    }
20 }
```

The terminal below shows the output of running the program:

```
ElifAppData\Roaming\Code\User\workspaceStorage\0da0eca37a02ff40d7c77de6f57fc2\redhat.java\jdt_ws\TUGAS_8bf3a14\bin\jobsheet8.Triangle7
Masukkan nilai n = 5
*
**
***
****
*****
PS C:\Users\Elif\Pictures\Programmer\TUGAS> [ ]
```

At the bottom right, it says "Elsaamarthalinda committed now" and "18bdccf".

#### Percobaan 4

The screenshot shows the VS Code interface with multiple tabs open at the top: PROBLEMS (17), OUTPUT, DEBUG CONSOLE, TERMINAL (active), and PORTS. The main editor area contains the following Java code for NilaiKelompok.java:

```
1 package jobsheet;
2 public class NilaiKelompok {
3     Kelompok 1
4     Nilai dari Kelompok Penilaian 1: 88
5     Nilai dari Kelompok Penilaian 2: 79
6     Nilai dari Kelompok Penilaian 3: 90
7     Nilai dari Kelompok Penilaian 4: 82
8     Nilai dari Kelompok Penilaian 5: 85
9     Kelompok 1: nilai rata-rata = 84.8
10    Kelompok 2
11    Nilai dari Kelompok Penilaian 1: 89
12    Nilai dari Kelompok Penilaian 2: 85
13    Nilai dari Kelompok Penilaian 3: 90
14    Nilai dari Kelompok Penilaian 4: 85
15    Nilai dari Kelompok Penilaian 5: 82
16    Kelompok 2: nilai rata-rata = 86.2
17    Kelompok 3
18    Nilai dari Kelompok Penilaian 1: 90
19    Nilai dari Kelompok Penilaian 2: 91
20    Nilai dari Kelompok Penilaian 3: 85
21    Nilai dari Kelompok Penilaian 4: 84
22    Nilai dari Kelompok Penilaian 5: 89
23    Kelompok 3: nilai rata-rata = 88.2
24    Kelompok 4
25    Nilai dari Kelompok Penilaian 1: 77
26    Nilai dari Kelompok Penilaian 2: 75
27    Nilai dari Kelompok Penilaian 3: 78
28    Nilai dari Kelompok Penilaian 4: 79
29    Nilai dari Kelompok Penilaian 5: 76
30    Kelompok 4: nilai rata-rata = 77.4
31    Kelompok 5
32    Nilai dari Kelompok Penilaian 1: 89
33    Nilai dari Kelompok Penilaian 2: 82
34    Nilai dari Kelompok Penilaian 3: 81
35    Nilai dari Kelompok Penilaian 4: 77
36    Nilai dari Kelompok Penilaian 5: 83
37    Kelompok 5: nilai rata-rata = 80.6
38    Kelompok 6
39    Nilai dari Kelompok Penilaian 1: 91
40    Nilai dari Kelompok Penilaian 2: 90
41    Nilai dari Kelompok Penilaian 3: 85
42    Nilai dari Kelompok Penilaian 4: 88
43    Nilai dari Kelompok Penilaian 5: 90
44    Kelompok 6: nilai rata-rata = 88.8
45    Kelompok 7
46    Nilai dari Kelompok Penilaian 1: 80
47    Nilai dari Kelompok Penilaian 2: 85
48    Nilai dari Kelompok Penilaian 3: 85
49    Nilai dari Kelompok Penilaian 4: 85
50    Nilai dari Kelompok Penilaian 5: 85
51    Kelompok 7: nilai rata-rata = 80.6
52    Kelompok 8
53    Nilai dari Kelompok Penilaian 1: 91
54    Nilai dari Kelompok Penilaian 2: 90
55    Nilai dari Kelompok Penilaian 3: 85
56    Nilai dari Kelompok Penilaian 4: 88
57    Nilai dari Kelompok Penilaian 5: 90
58    Kelompok 8: nilai rata-rata = 88.8
```

The terminal below shows the output of running the program:

```
ElifAppData\Roaming\Code\User\workspaceStorage\0da0eca37a02ff40d7c77de6f57fc2\redhat.java\jdt_ws\TUGAS_8bf3a14\bin\NilaiKelompok
Ln 23, Col 10 Spaces: 4 UTF-8 CRLF Java Signed out
8:06 AM 11/4/2025
```

At the bottom right, it says "Elsaamarthalinda committed 1 minute ago" and "daacf09".

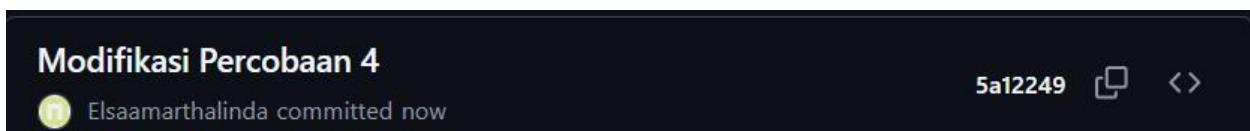
1. Jelaskan apa yang terjadi pada variabel **totalNilai** di setiap iterasi outer loop dan mengapa inisialisasinya (`total = 0`) berada di dalam outer loop, bukan di luar.
  - Di setiap iterasi perulangan luar, variabel **totalNilai** digunakan untuk mengakumulasi (menjumlahkan) semua nilai dari 5 penilaian untuk satu kelompok tertentu. Setelah 5 penilaian dijumlahkan, nilai rata-rata kelompok dihitung dan ditampilkan.
  - Inisialisasi (`total = 0`) berada di dalam outer loop karena **totalNilai** berfungsi sebagai akumulator khusus untuk setiap kelompok. Setiap kali program beralih ke kelompok baru (yaitu, setiap kali perulangan luar berlanjut ke iterasi berikutnya), jumlah total untuk kelompok sebelumnya harus diabaikan. Jika inisialisasi diletakkan di luar perulangan luar, **totalNilai** tidak akan pernah kembali ke nol. Program akan terus menambahkan nilai kelompok saat ini ke total nilai kelompok sebelumnya.
2. Modifikasi program di atas, sehingga dapat mencari kelompok dengan rata-rata nilai tertinggi dan tampilkan nomor kelompok tersebut.

```
rataNilai = totalNilai / 5;
System.out.println("Kelompok " + i + ": nilai rata-rata = " + rataNilai);

if (rataNilai > rataTertinggi) {
    rataTertinggi = rataNilai;
    kelompokTertinggi = i;
}
```

3. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 4”

```
File Edit Selection View ... ← → ⌂ TUGAS PROBLEMS 20 OUTPUT DEBUG CONSOLE TERMINAL PORTS
Nilai dari Kelompok Penilai 5: 85
Kelompok 1: nilai rata-rata = 84.8
Kelompok 2
Nilai dari Kelompok Penilai 1: 89
Nilai dari Kelompok Penilai 2: 85
Nilai dari Kelompok Penilai 3: 90
Nilai dari Kelompok Penilai 4: 85
Nilai dari Kelompok Penilai 5: 82
Kelompok 2: nilai rata-rata = 86.2
Kelompok 3
Nilai dari Kelompok Penilai 1: 90
Nilai dari Kelompok Penilai 2: 91
Nilai dari Kelompok Penilai 3: 86
Nilai dari Kelompok Penilai 4: 84
Nilai dari Kelompok Penilai 5: 90
Kelompok 3: nilai rata-rata = 88.2
Kelompok 4
Nilai dari Kelompok Penilai 1: 77
Nilai dari Kelompok Penilai 2: 75
Nilai dari Kelompok Penilai 3: 80
Nilai dari Kelompok Penilai 4: 79
Nilai dari Kelompok Penilai 5: 76
Kelompok 4: nilai rata-rata = 77.4
Kelompok 5
Nilai dari Kelompok Penilai 1: 80
Nilai dari Kelompok Penilai 2: 82
Nilai dari Kelompok Penilai 3: 81
Nilai dari Kelompok Penilai 4: 77
Nilai dari Kelompok Penilai 5: 83
Kelompok 5: nilai rata-rata = 80.6
Kelompok 6
Nilai dari Kelompok Penilai 1: 91
Nilai dari Kelompok Penilai 2: 90
Nilai dari Kelompok Penilai 3: 85
Nilai dari Kelompok Penilai 4: 88
Nilai dari Kelompok Penilai 5: 90
Kelompok 6: nilai rata-rata = 88.8
Kelompok dengan rata-rata tertinggi:
Kelompok 6 dengan nilai rata-rata 88.8
PS C:\Users\elif\Pictures\Penugrahan\TUGAS>
```



## Tugas 1

A screenshot of a Java IDE interface. The code editor shows a file named `Tugas1.java` with the following content:

```
1 package jobsheet8;
2 import java.util.Scanner;
3 public class Tugas1 {
4     Run(Debug)
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7         int n, i, j, jumlah;
8         for (n = 1; n <= 5; n++) {
9             jumlah = 0;
10            System.out.print("n = " + n + " jumlah kuadrat = ");
11            for (i = 1; i <= n; i++) {
12                jumlah += i * i;
13                System.out.print(i * i);
14                if (i < n) {
15                    System.out.print(" + ");
16                }
17            }
18            System.out.println(" = " + jumlah);
19        }
20    }
21 }
22 }
```

The terminal window shows the output of running the program:

```
PS C:\Users\elif\Pictures\Penograman\TUGAS> & 'C:\Program Files\Java\jdk-24\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\elif\AppData\Roaming\Code\User\workspaceStorage\0da0eca3a7a02f64ad9c77de5f57f2\redhat\java\jdt_ws\TUGAS_8bf3a14\bin' 'jobsheet8.Tugas1'
n = 1 jumlah kuadrat = 1
n = 2 jumlah kuadrat = 1 + 4 = 5
n = 3 jumlah kuadrat = 1 + 4 + 9 = 14
n = 4 jumlah kuadrat = 1 + 4 + 9 + 16 = 30
n = 5 jumlah kuadrat = 1 + 4 + 9 + 16 + 25 = 55
PS C:\Users\elif\Pictures\Penograman\TUGAS>
```

The status bar at the bottom indicates Java Ready.

## Tugas 1

Elsaamarthalinda committed now

60fc0ef

## Tugas 2

A screenshot of a Java IDE interface. The code editor shows a file named `Tugas2.java` with the following content:

```
1 package jobsheet2;
2 import java.util.Scanner;
3 public class Tugas2 {
4     Run(Debug)
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7         System.out.print("Masukkan nilai n (minimal 3): ");
8         int n = sc.nextInt();
9         if (n < 3) {
10             System.out.println("Nilai n minimal 3!");
11             return;
12         }
13         for (int i = 1; i <= n; i++) {
14             for (int j = 1; j <= n; j++) {
15                 if (i == 1 || i == n || j == 1 || j == n) {
16                     System.out.print(n + " ");
17                 } else {
18                     System.out.print("  ");
19                 }
20             }
21             System.out.println();
22         }
23     }
24 }
```

The terminal window shows the output of running the program:

```
PS C:\Users\elif\Pictures\Penograman\TUGAS> & 'C:\Program Files\Java\jdk-24\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\elif\AppData\Roaming\Code\User\workspaceStorage\0da0eca3a7a02f64ad9c77de5f57f2\redhat\java\jdt_ws\TUGAS_8bf3a14\bin' 'jobsheet2.Tugas2'
Masukkan nilai n (minimal 3): 5
5 5 5 5
5 5
5 5
5 5 5
PS C:\Users\elif\Pictures\Penograman\TUGAS>
```

The status bar at the bottom indicates Java Ready.

## Tugas 2

 Elsaamarthalinda committed now

49e9db5  < >

## Tugas 3

```
PS C:\Users\elif\Pictures\Pemrograman\TUGAS> & 'C:\Program Files\Java\jdk-24\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\elif\AppData\Roaming\Code\User\workspaceStorage\bda0eca37a02f6f40d7c77de6f577fc2\redhat.java\jdt_ws\TUGAS_8bf3a14\bin' 'jobsheet8.Tugas3'
Jumlah cabang kafe: 2
*** Input Penjualan Per Cabang ===
--- Cabang 1 ---
Jumlah pelanggan: 3
- Pelanggan 1 memesan berapa item?: 2
- Pelanggan 2 memesan berapa item?: 4
- Pelanggan 3 memesan berapa item?: 1
Cabang 1 :
- Pelanggan: 3 orang
- Item terjual: 7
--- Cabang 2 ---
Jumlah pelanggan: 4
- Pelanggan 1 memesan berapa item?: 3
- Pelanggan 2 memesan berapa item?: 5
- Pelanggan 3 memesan berapa item?: 1
- Pelanggan 4 memesan berapa item?: 2
Cabang 2 :
- Pelanggan: 4 orang
- Item terjual: 11
----- TOTAL SELURUH CABANG -----
Pelanggan : 7 orang
Item terjual : 18 item
PS C:\Users\elif\Pictures\Pemrograman\TUGAS>
```

## Tugas 3

 Elsaamarthalinda committed 2 minutes ago