

JOBSHEET 8

Nama : Dandin Sesilia

NIM : 254107020142

Mata Kuliah : Praktikum Dasar Pemrograman

Kelas : TI – 1H

Percobaan 1: Review Perulangan 1

```
package jobsheet8;

import java.util.Scanner;

public class star02 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in); Resource leak:
        System.out.print("Masukkan nilai n : ");
        int n = sc.nextInt();

        for (int i = 1; i <= n; i++) {
            System.out.print("*");
        }
    }
}
```

PS D:\DASPRO\DA
Masukkan nilai n : 5

PS D:\DASPRO\DA

1. Jika pada perulangan for, inisialisasi $i=1$ diubah menjadi $i=0$, apa akibatnya? Mengapa bisa demikian?

```
C:\Users\ANDIN\appdata\Roaming\Code\User
Masukkan nilai n : 5
*****
PS D:\DASPRO\DA
```

 - Jika inisialisasi $i = 1$ diubah menjadi $i = 0$, maka perulangan akan berjalan sebanyak 6 kali karena kondisinya adalah $<= n$. Output bintang yang dihasilkan juga sebanyak 6 karena perulangan dilakukan sebanyak 6 kali dimulai dari $i = 0$.
2. Jika pada perulangan for, kondisi $i<=n$ diubah menjadi $i<n$, bagaimana bentuk outputnya jika input $n = 5$? Mengapa hasilnya berbeda?

```
C:\Users\ANDIN\appdata\Roaming\Code\User
Masukkan nilai n = 5
*****
PS D:\DASPRO\DA
```

 - Jika kondisi diubah menjadi $i < n$, maka output yang dihasilkan adalah bintang sebanyak 4 karena inisialisasi $i = 1$ dan akan terus berulang sampai nilai $i < 5$. Perulangan berhenti setelah melakukan 4 kali perulangan karena nilai i sudah tidak lagi memenuhi kondisi (false). Apabila ingin memunculkan output bintang sebanyak 5, maka inisialisasi $i = 0$.
3. Jika pada perulangan for, kondisi $i<=n$ diubah menjadi $i > n$, apa akibatnya? Mengapa bisa demikian?

```
C:\Users\ANDIN\appdata\Roaming\Code\User
Masukkan nilai n = 5
PS D:\DASPRO\DA
```

- Apabila kondisi diubah menjadi $i > n$, maka perulangan for tidak akan dijalankan. Karena kondisi sudah tidak memenuhi dari awal, dimana inisialisasi i dimulai dari 1 dan saat dicek dalam perulangan ($1 > 5$) adalah false, sehingga program langsung keluar dari perulangan for.
4. Jika pada perulangan for, kondisi step $i++$ diubah menjadi $i--$ apa akibatnya? Mengapa bisa demikian?

```

11
12     for (int i = 1; i <= n; i++) {
13         System.out.print("*");
14     }
15 }
16
17

```

The screenshot shows a Java code editor with the following code:

```

11
12     for (int i = 1; i <= n; i++) {
13         System.out.print("*");
14     }
15 }
16
17

```

The code is a simple for loop that prints asterisks from 1 to n. The step is currently set to $i++$. The editor interface includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS, along with a RUN START button.

- Jika kondisi step $i++$ diubah menjadi $i--$, maka perulangan akan berjalan terus menerus tanpa henti karena program akan berhenti jika mencapai kondisi $i \leq n$, sedangkan jika $i = n$ —kondisi tersebut tidak akan pernah tercapai.
5. Jika pada perulangan for, step $i++$ diubah menjadi $i += 2$, bagaimana pola outputnya jika input $n = 6$? Apa yang menyebabkan perubahan tersebut?

```

11
12     for (int i = 1; i <= n; i+= 2) {
13         System.out.print("*");
14     }
15 }
16
17

```

The screenshot shows a Java code editor with the following code:

```

11
12     for (int i = 1; i <= n; i+= 2) {
13         System.out.print("*");
14     }
15 }
16
17

```

The code is a for loop that prints asterisks at odd indices (1, 3, 5). The step is currently set to $i += 2$. The editor interface includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS, along with a RUN START button. The terminal shows the command PS D:\DASPRO\DA PRO\PraktikumDaspro> & 'C:\Program' and the user prompt 'Masukkan nilai n = 6'.

- Apabila $i++$ diubah menjadi $i += 2$ dan input $n = 6$, maka pola output yang dihasilkan adalah bintang sebanyak 3 karena stepnya bertambah 2 selama kondisi $i \leq n$ terpenuhi.
 - Inisialisasi dimulai dari 1, maka $1 \leq 6$ (true) lalu $1 + 2 = 3$.
 - Nilai $i = 3$, maka $3 \leq 6$ (true) lalu $3 + 2 = 5$.
 - Nilai $i = 5$, maka $5 \leq 6$ (true) lalu $5 + 2 = 7$.
 - Nilai $i = 7$, maka $7 \leq 6$ (false) program berhenti.

Output yang dihasilkan adalah 1,3,5 (bintang sebanyak 3).

Percobaan 2: Bintang Persegi

```
1 package jobsheet8;
2
3 import java.util.Scanner;
4
5 public class square07 {
6     Run main | Run main | Debug main | Run | Debug
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in); Resource le
9         System.out.print("Masukkan nilai N = ");
10        int N = sc.nextInt();
11
12        for (int iOuter = 1; iOuter <= N; iOuter++) {
13            for (int i = 1; i <= N; i++) {
14                System.out.print("*");
15            }
16            System.out.println();
17        }
18    }
19}
```

Masukkan nilai N = 5

PS D:\DASPRO\DA
S

1. Perhatikan perulangan luar. Jika pada sintaks for, inisialisasi iOuter=1 diubah menjadi iOuter=0, apa yang akibatnya? Mengapa bisa demikian?

```
12 for (int iOuter = 0; iOuter <= N; iOuter++) {
13     for (int i = 1; i <= N; i++) {
14         System.out.print("*");
15     }
16     System.out.println();
17 }
```

Masukkan nilai N = 5

➤ Apabila inisialisasi iOuter diubah menjadi iOuter = 0, maka perulangan akan dilakukan 6 kali dan output menghasilkan bintang 6 baris. Karena jika i dimulai dari 0, kondisi yang memenuhi $i \leq N$ adalah sebanyak 6 kali. Jika i mencapai 6 dimana kondisi $6 \leq 5$ adalah false, maka program keluar dari perulangan.

2. 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?

```
12 for (int iOuter = 1; iOuter <= N; iOuter++) {
13     for (int i = 0; i <= N; i++) {
14         System.out.print("*");
15     }
16     System.out.println();
17 }
```

aming\Code\User\workspaceStorage\fec03684ed3b31398a47ad15fc04df
f9ebdc80\bin' 'jobsheet8.square07'
Masukkan nilai N = 5

➤ Jika inisialisasi i diubah jadi i = 0 dalam inner loop, maka jumlah kolom bintang berubah menjadi 6 kolom karena perulangan dimulai dari 0 sampai $i \leq 5$.

3. Apakah perbedaan kegunaan antara perulangan luar dengan perulangan yang berada di dalamnya?

➤ Dalam nested loop, perulangan luar (outer loop) digunakan untuk menentukan banyak baris yang akan dicetak. Sedangkan perulangan dalam (inner loop) digunakan untuk menentukan banyak kolom yang dicetak setiap baris.

4. Mengapa perlu ditambahkan sintaks System.out.println(); di bawah perulangan dalam? Apa akibatnya jika sintaks tersebut dihilangkan?

```

12     for (int iOuter = 1; iOuter <= N; iOuter++) {
13         for (int i = 0; i <= N; i++) {
14             System.out.print(s:"*");
15         }
16     }

```

PROBLEMS 24 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\DASPRO\DA PRO\PraktikumDaspro> & Open folder in new window (ctrl +
CodeDetailsInExceptionMessages' '-cp' 'C:\Users\ANDINI\AppData\Roam
age\fec03684ed3b31398a47ad15fc04dfe6\redhat.java\jdt_ws\PraktikumDa
t8.square07'
Masukkan nilai N = 5

- Sintaks System.out.println(); di bawah perulangan dalam digunakan untuk membuat baris baru setiap perulangannya. Apabila sintaks tersebut dihilangkan, maka output bintang setiap perulangannya akan saling menyambung dan tidak membentuk persegi.

5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 2”

PraktikumDaspro / jobsheet8 / square07.java

sesilia-acc Modifikasi percobaan 2

Code Blame 18 lines (14 loc) · 410 Bytes

```

1 package jobsheet8;
2
3 import java.util.Scanner;
4
5 public class square07 {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8
9         System.out.print("Masukkan nilai N = ");
10        int N = sc.nextInt();
11
12        for (int iOuter = 1; iOuter <= N; iOuter++) {
13            for (int i = 0; i <= N; i++) {
14                System.out.print("*");
15            }
16        }
17    }
18 }

```

Percobaan 3: Bintang Segitiga

```

5 public class triangle07 {
6     public static void main(String[] args) {
7         import java.util.Scanner;
8
9         public class triangle07 {
10             Run main | Debug main | Run | Debug
11             public static void main(String[] args) {
12                 Scanner sc = new Scanner(System.in); Re
13
14                 System.out.print(s:"Masukkan nilai n = ");
15                 int n = sc.nextInt();
16                 int i = 0;
17
18                 while (i <= n) {
19                     int j = 0;
20                     while (j < i) {
21                         System.out.print(s:"*");
22                         j++;
23                     }
24                     i++;
25                 }
26             }
27         }
28     }
29 }

```

.java\jdt_ws\PraktikumDaspro_f9ebdc80\bin'

Masukkan nilai n = 5

PS D:\DASPRO\DA PRO\PraktikumDaspro>

1. Perhatikan, apakah output yang dihasilkan dengan nilai n = 5 sesuai dengan tampilan berikut?

```

*
**
***
*****

```

- Tidak.
2. Jika tidak sesuai, bagian mana saja yang harus diperbaiki/ditambahkan? Jelaskan setiap bagian yang perlu diperbaiki/ditambahkan.
- Bagian yang harus ditambahkan sintaks System.out.println(); di bawah i++ pada outer loop. Sintaks ini digunakan untuk membuat baris baru dalam setiap perulangan yang dilakukan.

```

13     while (i <= n) {
14         int j = 0;
15         while (j < i) {
16             System.out.print(s:"*");
17             j++;
18         }
19         i++;
20         System.out.println();
21     }

```

Output window:

```

ava\jdt_ws\PraktikumDaspro_f9ebdc80\bin' 'jobsh
Masukkan nilai n = 5

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

3. 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 untuk menentukan berapa banyak baris yang akan dicetak dan diulang. Setiap i++ maka program akan berpindah ke baris selanjutnya. Sedangkan variabel j berperan untuk menentukan kolom yang akan dicetak dan diulang setiap barisnya. Setiap j++ maka program akan berpindah ke kolom berikutnya. Selama j < i program akan mencetak bintang di baris yang sama.
 - Variabel j di-set ulang ke 0 di awal iterasi outer loop karena perulangan harus dimulai lagi dari kolom awal. Jika j tidak direset, maka perulangan akan dilakukan dengan nilai j = 0. Dimana jika mencapai kondisi 1 < 0 (false), program akan berpindah ke baris selanjutnya dan pola segitiga tidak akan terbentuk.

```

8
9     System.out.print(s:"Masukkan nilai n = ");
10    int n = sc.nextInt();
11    int i = 0;
12    int j = 0;
13
14    while (i <= n) {
15        while (j < i) {
16            System.out.print(s:"*");
17            j++;
18        }
19        i++;
20        System.out.println();
21    }

```

Output window:

```

redhat.java\jdt_ws\PraktikumDaspro_f9ebdc80\bin' 'jobsheet8.t
Masukkan nilai n = 5

*
*
*
*
*
*
```

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

```

PraktikumDaspro / jobsheet8 / triangle07.java

sesilia-acc Modifikasi percobaan 3

Code Blame 23 lines (19 loc) · 490 Bytes
1 package jobsheet8;
2
3 import java.util.Scanner;
4
5 public class triangle07 {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8
9         System.out.print("Masukkan nilai n = ");
10        int n = sc.nextInt();
11        int i = 0;
12        int j = 0;
13
14        while (i <= n) {
15            while (j < i) {
16                System.out.print("*");
17                j++;
18            }
19            i++;
20            System.out.println();
}

```

Percobaan 4: Studi Kasus Nilai Tugas Proyek Kelompok

```

package jobsheets;
import java.util.Scanner;
public class nilaiKelompok07 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int i = 1, j, nilai;
        float totalNilai, rataNilai;
        while (i <= 6) {
            System.out.println("Kelompok " + i);
            totalNilai = 0;
            for (j = 1; j <= 5; j++) {
                System.out.print("Nilai dari kelompok penilai " + j + " = ");
                nilai = sc.nextInt();
                totalNilai += nilai;
            }
            rataNilai = totalNilai / 5;
            System.out.println("Kelompok " + i + " : " + "Nilai rata - rata = " + rataNilai);
            i++;
        }
    }
}

```

Kelompok 1
Nilai dari kelompok penilai 1 = 88
Nilai dari kelompok penilai 2 = 79
Nilai dari kelompok penilai 3 = 90
Nilai dari kelompok penilai 4 = 82
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
PS D:\DASPRO\PraktikumDaspro>

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 outer loop variabel totalNilai menyimpan value berupa total nilai dari 5 kelompok penilai. Inisialisasi totalNilai = 0 berada di dalam outer loop untuk kembali mereset totalNilai mulai 0 saat dilakukan perulangan berikutnya. Jika inisialisasi dilakukan di luar loop, maka totalNilai terus bertambah value-nya karena perhitungan melanjutkan nilai yang paling akhir digunakan dan tidak dimulai dari 0.
2. Modifikasi program di atas, sehingga dapat mencari kelompok dengan rata-rata nilai tertinggi dan tampilkan nomor kelompok tersebut.

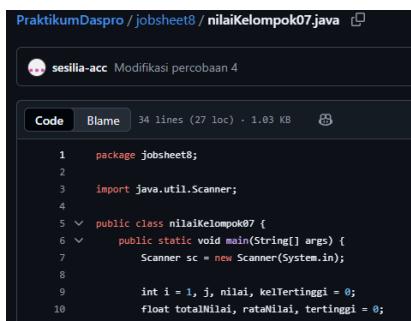
```

1 package jobsheet8;
2
3 import java.util.Scanner;
4
5 public class nilaiKelompok07 {
6     Run/Debug [Run main | Debug main]
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in);
9
10        int i = 1, j, nilai, kelTertinggi = 0;
11        float totalNilai, rataNilai, tertinggi = 0;
12
13        while(i <= 5) {
14            System.out.println("Kelompok " + i);
15
16            totalNilai = 0;
17            for (j = 1; j <= 5; j++) {
18                System.out.print("Nilai dari kelompok penilai " + j + " = ");
19                nilai = sc.nextInt();
20                totalNilai += nilai;
21            }
22            rataNilai = totalNilai / 5;
23            System.out.println("Nilai rata - rata = " + rataNilai);
24
25            if (rataNilai > tertinggi) {
26                tertinggi = rataNilai;
27                kelTertinggi = i;
28            }
29            i++;
30        }
31
32        System.out.print("Kelompok dengan rata - rata nilai tertinggi adalah kelompok = " + kelTertinggi);
33        System.out.println("Nilai rata - rata nilai : " + tertinggi);
34    }
35 }

```

Nilai dari kelompok penilai 5 = 68
 Kelompok 4 : Nilai rata - rata = 77.4
 Kelompok 5
 Nilai dari kelompok penilai 1 = 90
 Nilai dari kelompok penilai 2 = 85
 Nilai dari kelompok penilai 3 = 75
 Nilai dari kelompok penilai 4 = 77
 Nilai dari kelompok penilai 5 = 76
 Kelompok 5 : Nilai rata - rata = 80.6
 Kelompok 6
 Nilai dari kelompok penilai 1 = 67
 Nilai dari kelompok penilai 2 = 90
 Nilai dari kelompok penilai 3 = 87
 Nilai dari kelompok penilai 4 = 86
 Nilai dari kelompok penilai 5 = 79
 Kelompok 6 : Nilai rata - rata = 81.8
 Kelompok dengan rata - rata nilai tertinggi adalah kelompok = 1
 dengan rata - rata nilai : 86.2
 PS D:\DASPRO\Daspro\PraktikumDaspro> []

- Commit and push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 4”



```

PraktikumDaspro / jobsheet8 / nilaiKelompok07.java

sesilia-acc Modifikasi percobaan 4

Code Blame 34 lines (27 loc) · 1.03 KB

```

```

1 package jobsheet8;
2
3 import java.util.Scanner;
4
5 public class nilaiKelompok07 {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8
9         int i = 1, j, nilai, kelTertinggi = 0;
10        float totalNilai, rataNilai, tertinggi = 0;

```

TUGAS!

- Buatlah program untuk menghitung dan menampilkan jumlah kuadrat bilangan 1 s.d n. Gunakan perulangan bersarang. Berikut output yang diharapkan jika n pada rentang 1 s.d 5.

```

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

```

```

6  public static void main(String[] args) {
7      Scanner sc = new Scanner(System.in);    Resource leak: 'sc' is never closed
8
9      System.out.print("Masukkan nilai n = ");
10     int nilai = sc.nextInt();
11
12     int i = 1;
13     while (i <= nilai) {
14         System.out.print("\n" + "-----");
15         int jumlah = 0;
16
17         int j = 1;
18         while (j <= i) {
19             int kuadrat = j * j;
20             jumlah += kuadrat;
21             if (j < i) {
22                 System.out.print(j + " " + "+");
23             } else if (j == i) {
24                 System.out.print(j + " ");
25             }
26             j++;
27         }
28         System.out.print(" = " + jumlah);
29         i++;
30     }
31 }

```

Masukkan nilai n = 5

n = 1 Jumlah kuadrat = 1 = 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

Commit dan push program Anda ke Github dengan pesan “Tugas 1”



- Buatlah program untuk mencetak tampilan persegi angka seperti di bawah ini berdasarkan input n (nilai n minimal 3). Contoh n = 3, dan n = 5

```

6  public static void main(String[] args) {
7      Scanner sc = new Scanner(System.in);    Resource leak: 'sc' is never closed
8
9      System.out.print("Masukkan n = ");
10     int n = sc.nextInt();
11
12     if (n < 3) {
13         System.out.println("Masukkan angka yang lebih besar");
14         return;
15     }
16
17     int i = 1;
18     while (i <= n) {
19
20         int j = 1;
21         while (j <= n) {
22             if (i == 1 || i == n || j == 1 || j == n) {
23                 System.out.print(n + " ");
24             } else {
25                 System.out.print(" " );
26             }
27             j++;
28         }
29         System.out.println();
30         i++;
31     }

```

Masukkan n = 5

Masukkan n = 3

PS D:\DASPRO\DA

Commit dan push program Anda ke Github dengan pesan “Tugas 2”!



- Sebuah jaringan kafe “Kopi Senja” memiliki beberapa cabang di berbagai lokasi kota. Untuk memantau kinerja operasional harian, manajemen membutuhkan data sederhana mengenai aktivitas penjualan di setiap cabang. Setiap hari, setiap cabang melayani sejumlah pelanggan, dan setiap pelanggan memesan satu atau lebih item, baik berupa makanan maupun minuman. Buatlah program untuk mencatat dan menghitung banyak pelanggan dan total item yang terjual pada masing-masing cabang. Karena sistem yang

digunakan masih sederhana dan tidak menyimpan data historis, program hanya perlu mencatat secara langsung berapa banyak pelanggan yang dilayani dan berapa total item yang terjual di masing-masing cabang, tanpa perlu menyimpan nama pelanggan, jenis menu, atau detail lainnya. Program akan meminta input jumlah cabang terlebih dahulu. Lalu untuk setiap cabang, operator memasukkan jumlah pelanggan yang datang hari ini. Selanjutnya, untuk setiap pelanggan, operator memasukkan jumlah item yang dipesan. Program kemudian menghitung dan menampilkan ringkasan penjualan per cabang, serta total keseluruhan dari seluruh cabang pada akhir sesi. Berikut adalah contoh input dan output program:

```
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
```

```
3 import java.util.Scanner;
4
5 public class kafesenja07 {
6     Run | Debug | Run main | Debug main
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in); Resource leak: 'sc' is never closed
9
10        System.out.print("Jumlah cabang kafe = ");
11        int cabang = sc.nextInt();
12        System.out.println("\n==== Input Penjualan Per Cabang ===");
13
14        int totalPelanggan = 0, totalItem = 0;
15
16        int i = 1;
17        while (i <= cabang) {
18            System.out.println("\n--- Cabang " + i + " ---");
19            int ItemCabang = 0;
20            System.out.print("Jumlah pelanggan = ");
21            int PelangganCabang = sc.nextInt();
22
23            for (int j = 1; j <= PelangganCabang; j++) {
24                System.out.print("- Pelanggan " + j + " memesan berapa item? ");
25                int item = sc.nextInt();
26                ItemCabang += item;
27            }
28
29            totalPelanggan += PelangganCabang;
30            totalItem += ItemCabang;
31
32            System.out.println("Cabang " + i + ":");
33            System.out.print(" Pelanggan = " + PelangganCabang + " orang");
34            System.out.print("\n- Item terjual = " + ItemCabang);
35            System.out.println();
36
37        }
38
39        System.out.println("\nTotal seluruh cabang :");
40        System.out.println(" Pelanggan = " + totalPelanggan + " orang");
41        System.out.println(" Item terjual = " + totalItem + " item");
42
43    }
44}
```

```
Jumlah cabang kafe = 2
 === Input Penjualan Per Cabang ===

--- Cabang 1 ---
Jumlah pelanggan = 3
- Pelanggan 1 memesan berapa item? 4
- Pelanggan 2 memesan berapa item? 2
- 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? 2
- Pelanggan 4 memesan berapa item? 1
Cabang 2 :
- Pelanggan = 4 orang
- Item terjual = 11

Total seluruh cabang :
Pelanggan = 7 orang
Item terjual = 18 item
PS D:\DASPRO\DA PRO\PraktikumDaspro>
```

Commit dan push program Anda ke Github dengan pesan “Tugas 3”!