

Nama : Khania Puji Auliya

NIM : 254107020236

Prodi : D-IV Teknik Informatika

Kelas : 1G

Percobaan 1

Pertanyaan

1. Jika pada perulangan for, inisialisasi  $i = 1$  diubah menjadi  $i = 0$ , apa akibatnya? Mengapa bisa demikian?
2. Jika pada perulangan for, kondisi  $i \leq n$  diubah menjadi  $i < n$ , bagaimana bentuk outputnya jika input  $n=5$ ? Mengapa hasilnya berbeda?
3. Jika pada perulangan for, kondisi  $i \leq n$  diubah menjadi  $i > n$ , apa akibatnya? Mengapa bisa demikian?
4. Jika pada perulangan for, kondisi step  $i++$  diubah menjadi  $i--$  apa akibatnya? Mengapa bisa demikian?
5. Jika pada perulangan for, step  $i++$  diubah menjadi  $i += 2$ , bagaimana pola outputnya jika input  $n = 6$ ? Apa yang menyebabkan perubahan tersebut?

Jawab:

1. Jika  $i = 1$  diubah menjadi  $i = 0$  akibatnya jumlah Bintang akan terus bertambah satu.
  - Semua loop berjalan dari  $i = 1$  sampai  $i = n$ , maka total iterasi  $n$  kali
  - Setelah diubah menjadi  $i = 0$  sampai  $i = n$ , maka total iterasi menjadi  $n + 1$  kali, karena dimulai dari angka 0.
2. Pada kondisi  $i \leq n$ , hasilnya akan mengoutput 5 bintang  
Pada kondisi  $i < n$ , hasilnya akan mengoutput 4 bintang  
Karena kondisi  $i < n$  mengehentikan perulangan sebelum mencapai nilai  $n$ , sehingga satu iterasi tidak muncul.
3. Jika kondisi  $i \leq n$  diubah menjadi  $i > n$  akibatnya perulangan tidak bisa berjalan sama sekali, karena  $i$  dimulai dari 1 yang dimana tidak ada bilangan bulat yang lebih kecil dari 1, dan karena kondisi awal sudah false, loop akan langsung berhenti.
4. Jika kondisi step  $i++$  diubah menjadi  $i--$ , maka perulangan menjadi infinite loop (tidak berhenti), karena  $i--$  membuat nilai  $i$  semakin kecil ( $1, 0, -1, -2, \dots$ ),  $i$  tidak pernah lebih besar dari  $n$  sehingga kondisi  $i \leq n$  akan selamanya true.
5. Jika kondisi step  $i++$  diubah menjadi  $i += 2$ , maka  $i$  bertambah dua-dua ( $1, 3, 5$ , berhenti)  
Penyebab perubahannya  $i += 2$  membuat loop hanya mengeksekusi untuk nilai ganjil, dan perulangan menjadi lebih sedikit karena kenaikannya meloncat 2 langkah per iterasi.

## Percobaan 2

### Pertanyaan

1. Perhatikan perulangan luar. Jika pada sintaks for, inisialisasi iOuter=1 diubah menjadi iOuter=0, apa yang akibatnya? Mengapa bisa demikian?
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?
3. Apakah perbedaan kegunaan antara perulangan luar dengan perulangan yang berada di dalamnya?
4. Mengapa perlu ditambahkan sintaks System.out.println(); di bawah perulangan dalam? Apa akibatnya jika sintaks tersebut dihilangkan?
5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 2”

Jawab:

1. Jika inisialisasi iOuter=1 diubah menjadi iOuter=0 akibatnya jumlah baris Bintang bertambah 1 baris, karena semua loop berjalan dari 1 sampai n, sehingga menghasilkan n baris, dan setelah diubah menjadi 0 sampai n, loop berjalan sebanyak n + 1 kali.
2. Jika inisialisasi pada perulangan dalam diubah dari i = 1 menjadi i = 0, maka dalam setiap baris akan muncul satu bintang tambahan. Penyebabnya adalah perubahan jumlah perulangan: awalnya loop berjalan dari 1 hingga n, sehingga menghasilkan n bintang per baris. Setelah diubah menjadi 0 hingga n, loop akan berjalan satu kali lebih banyak, sehingga total bintang yang tercetak menjadi n+1.
3. Perulangan luar berperan menentukan berapa banyak baris yang akan dibuat pada pola persegi. Setiap kali iterasi pada perulangan luar berarti satu baris baru ditampilkan. Sebaliknya, perulangan dalam bertugas menentukan jumlah bintang yang akan muncul dalam setiap baris tersebut. Jadi, perulangan luar mengatur tinggi pola, sedangkan perulangan dalam mengatur lebarnya.
4. Perintah System.out.println() diperlukan agar setelah seluruh bintang dalam satu baris dicetak, kursor berpindah ke baris berikutnya. Jika perintah ini dihapus, semua bintang dari seluruh iterasi akan dicetak dalam satu garis yang sama tanpa perpindahan baris. Akibatnya, bentuk persegi yang seharusnya tercipta akan berubah menjadi deretan bintang panjang dalam satu baris saja.

5.

```

1 package jobsheet8;
2
3 import java.util.Scanner;
4
5 public class Square15 {
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        for (int iOuter = 1; iOuter <= n; iOuter++) {
12            for (int i = 1; i <= n; i++) {
13                System.out.print("*");
14            }
15        }
16    }
17 }

```

## Percobaan 3

### Pertanyaan

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

`*
 **
 ***
 ****
 *****`
- Jika tidak sesuai, bagian mana saja yang harus diperbaiki/ditambahkan? Jelaskan setiap bagian yang perlu diperbaiki/ditambahkan.
- 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?
- Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 3”

Jawab:

- Tidak sama
- Bagian yang harus ditambahkan yaitu `System.out.println();` dibawah perulangan dalam, agar menghasilkan output yang sesuai, sintaks tersebut ditambahkan karena dapat membuat baris baru (new line) setiap kali perulangan luar selesai menjalankan perulangan dalam.

3. Peran variabel i: Variabel i berfungsi sebagai penghitung untuk perulangan luar. Nilai i menentukan baris ke berapa yang sedang dicetak. Semakin besar nilai i, semakin banyak bintang yang ditampilkan dalam satu baris. Dengan kata lain, i digunakan untuk mengatur jumlah baris sekaligus ukuran segitiga yang bertambah setiap barisnya.

Peran variabel j: Variabel j berfungsi sebagai penghitung untuk perulangan dalam. Nilai j mengatur berapa banyak bintang yang dicetak pada baris tertentu. Dalam satu iterasi outer loop, j akan berjalan dari 0 hingga kurang dari i, sehingga jumlah bintang per baris selalu menyesuaikan dengan nilai i.

Variabel j berfungsi sebagai penghitung untuk perulangan dalam. Nilai j mengatur berapa banyak bintang yang dicetak pada baris tertentu. Dalam satu iterasi outer loop, j akan berjalan dari 0 hingga kurang dari i, sehingga jumlah bintang per baris selalu menyesuaikan dengan nilai i. Jika tidak direset baris kedua dan selanjutnya tidak mencetak bintang sama sekali atau perulangan dalam tidak berjalan dengan benar karena nilai j sudah mencapai nilai batas dari iterasi sebelumnya.

The screenshot shows a GitHub commit page for a repository named 'PraktikumDaspro'. The commit is titled 'Commit b60431a' and was made by 'khniaaaly' on Oct 28. The commit message is 'Modifikasi Percobaan 3'. The code file 'Triangle15.java' has been modified, with line 18 being added. The code is as follows:

```
15 15      System.out.print("*");
16 16      j++;
17 17  }
18 +     System.out.println();
19 19      i++;
20 21  }
```

Comments: 0

4.

## Percobaan 4

### Pertanyaan

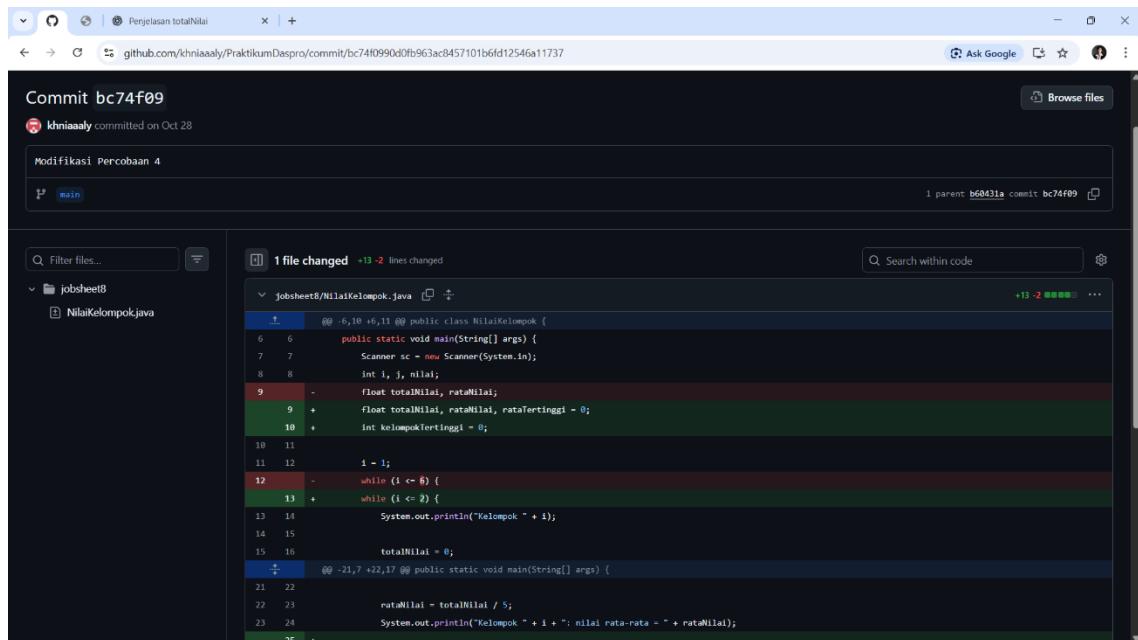
1. Jelaskan apa yang terjadi pada variabel totalNilai di setiap iterasi outer loop dan mengapa inisialisasinya (`total = 0`) berada di dalam outer loop, bukan di luar.
2. Modifikasi program di atas, sehingga dapat mencari kelompok dengan rata-rata nilai tertinggi dan tampilkan nomor kelompok tersebut.

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

Jawab:

- totalNilai harus di-reset di dalam outer loop agar perhitungan setiap kelompok berdiri sendiri, tidak tercampur dengan kelompok yang sudah diproses sebelumnya.

2.



# TUGAS

The screenshot shows the Eclipse IDE interface with the following details:

- File Explorer (Left):** Shows the project structure with files like `cm2_khania.java`, `cm2_15.java`, and `Tugas1.java`.
- Open Editors (Top):** Displays three editor tabs: `J cm2_khania.java 1, U`, `J cm2_15.java U`, and `J Tugas1.java 3 X`.
- Code Editor (Center):** The `Tugas1.java` file is open, containing the following Java code:

```
package jobsheets;
import java.util.Scanner;

public class Tugas1 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n, i, jumlah;
        for (n = 1; n <= 5; n++) {
            jumlah = 0;
            System.out.print("n = " + n + " jumlah kuadrat = ");
            for (i = 1; i <= n; i++) {
                jumlah += i * i;
                System.out.print(i + " ");
                if (i < n) {
                    System.out.print(" + ");
                }
            }
            System.out.println(" = " + jumlah);
        }
    }
}
```
- Bottom Navigation Bar:** Includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS.
- Bottom Status Bar:** Shows the path `D:\PraktikumDaspro\PraktikumDaspro`, the status `Java Ready`, and the message `Khanie Puji Aullya (1 month ago)`.

1.

The screenshot shows a Java development environment with the following details:

- File Explorer:** Shows files like cm2\_khana.java, cm2\_15.java, and Tugas2.java.
- Open Editors:** Shows multiple tabs for cm2\_khana.java, cm2\_15.java, and Tugas2.java.
- PRAKTIKUMDASPRO:** A folder containing various assignment files (e.g., cm2\_khana.java, cm2\_15.java) and a README file.
- Code Editor:** The Tugas2.java file is open, displaying code for a program that prints a diamond pattern based on user input n. The code uses Scanner to read input and System.out.println to print the pattern.
- Terminal:** Shows the command PS D:\PraktikumDaspro\PraktikumDaspro> & 'C:\jdk-24.0.1\jdk-24.0.1\bin\java.exe' -XX:+ShowCodeDetailsInExceptionMessages -cp 'C:\Users\Evie Puji Fitriana\AppData\Roaming\CodeBlocks\Workspaces\storage\15049c17c15ba297f4f2bb1ecf1\redhat.java\dt\_ws\PraktikumDaspro\_25915f8a\bin' 'jobsheet8.Tugas2'.
- Output:** Shows the execution result: Metode main di luar n (minimal 3): 3 3 3 3 3 3 3 3

2.

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

- File Explorer:** Shows various Java files including cm2\_khania.java, cm2\_15.java, Tugas3.java, jobsheet8.java, jobsheet5.java, jobsheet7.java, jobsheet9.java, Nilaikelomp..java, Square15.java, Star15.java, Triangle15.java, Tugas1.java, Tugas2.java, and Tugas3.java.
- Code Editor:** The active file is Tugas3.java. The code is as follows:

```
public class Tugas3 {
    public static void main(String[] args) {
        jumlahCabang = sc.nextInt();
        for (int i = 1; i < jumlahCabang; i++) {
            System.out.println("Masukkan jumlah pelanggan hari ini: ");
            int jumlahPelanggan = sc.nextInt();
            int totalItemCabang = 0;
            for (int j = 1; j <= jumlahPelanggan; j++) {
                System.out.print("Masukkan jumlah item yang dipesan oleh pelanggan ke-" + j + ": ");
                int item = sc.nextInt();
                totalItemCabang += item;
            }
            System.out.println("Cabang ke- " + i + " melayani " + jumlahPelanggan + " pelanggan.");
            System.out.println("Total item terjual di cabang ini: " + totalItemCabang);
            totalPelangganSemua += jumlahPelanggan;
            totalItemSemua += totalItemCabang;
        }
        System.out.println("\n===== RINGKASAN PENJUALAN SELURUH CABANG =====");
        System.out.println("Total pelanggan dari semua cabang : " + totalPelangganSemua);
        System.out.println("Total item terjual dari semua cabang : " + totalItemSemua);
        System.out.println("===== ");
    }
}
```

- Output Console:** Displays the execution results:

```
Masukkan jumlah item yang dipesan oleh pelanggan ke-5: 3
Cabang ke-2 melayani 5 pelanggan.
total item terjual di cabang ini: 13
=====
===== RINGKASAN PENJUALAN SELURUH CABANG =====
total pelanggan dari semua cabang : 7
Total item terjual dari semua cabang : 20
=====
```

3.