

## Pertanyaan 1.

```
jobsheet8 > J star25.java > ⚡ star25 > main(String[])
1 package jobsheet8;
2
3 import java.util.Scanner;
4
5 public class star25 {
6     Run | Debug
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in);
9
10        System.out.print(s: "Masukan nilai n = ");
11        int n = sc.nextInt();
12
13        for (int i = 1; i <= n; i++) {
14            System.out.print(s: "* ");
15        }
16    }
17}
```

### 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. Jumlah Bintang bertambah 1 karena variable  $I$  dimulai dari 0.
2. Jumlah Bintang berkurang 1 karena batas variabel menjadi 4
3. Hasil output tidak muncul karena nilai  $I$  tidak memenuhi program for. Maka program tidak mengulang.
4. Program looping tanpa henti. Jumlah Bintang karena perulangan tidak dibatas
5. Bintang muncul hanya 3, karena data  $I$  ditambah 2 ( $I += 2$ ) maka Bintang muncul saat variable  $I$  di angka 1, 3, 5.

## Pertanyaan 2

```
jobsheet8 > J square25.java > 🏃 square25 > ⚙ main(String[])
1 package jobsheet8;
2
3 import java.util.Scanner;
4
5 public class square25 {
6     Run | Debug
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in);
9
10        System.out.print(s: "Masukan Nilai N = ");
11        int N = sc.nextInt();
12
13        for (int iOuter1 = 1; iOuter1 <= N; iOuter1++) {
14            for (int i = 1; i <= N; i++) {
15                System.out.print(s: "* ");
16            }
17            System.out.println();
18        }
19    }
20 }
```

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. Jumlah baris Bintang akan bertambah 1 baris karena outer loop dimulai dari 0
2. Jumlah kolom Bintang akan bertambah 1 kolom karena inner loop dimulai dari 0
3. Perulangan luar (outer loop) berfungsi untuk menambahkan baris, dan perulangan dalam (inner loop) berfungsi untuk menambah jumlah kolom,
4. Berfungsi untuk meng-enter kolom Bintang. Jika dihilangkan, output Bintang selalu lurus.
5. Sepertinya tidak perlu dimodifikasi.

### Pertanyaan 3

```
1 package jobsheet8;
2
3 import java.util.Scanner;
4
5 public class triangle25 {
6     Run| Debug
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in);
9
10        System.out.print("Masukan nilai N = ");
11        int n = sc.nextInt();
12
13        int i = 0;
14
15        while (i < n) {
16            int j = 0;
17            while (j < i) {
18                System.out.print("* ");
19                j++;
20            }
21            System.out.println(); //jobsheet lupa menambahkan output di loop outer
22            i++;
23        }
24    }
25
26 }
```

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

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

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

### Jawab

1. Tidak. Namun sudah diubah pada gambar diatas.
2. Saya menambahkan program print di bagian inner loop agar setelah variable  $j$  terpenuhi, output akan meng-enter urutan bintangnya.
3. Variable  $I$  berfungsi untuk membuat outer loop, sedangkan variable  $j$  untuk inner loop. Jika  $j$  tidak di reset, pola bentuknya menjadi 1 2 2 2 bintang. Ini disebabkan karena jumlah  $j$  mengikuti variable  $I$ .  $j$  hanya menambah 2 bintang lalu Kembali lagi ke outer loop
4. sudah

## Pertanyaan 4

```
jobsheet8 > J nilaiKelompok25.java > T main(String[] args)
1 package jobsheet8;
2
3 import java.util.Scanner;
4
5 public class nilaiKelompok25 {
6     Run|Debug
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in);
9
10        int i, j, nilai;
11        double rataNilai, totalNilai;
12
13        i = 1;
14        while (i <= 6) {
15            System.out.println("Kelompok " + i);
16
17            totalNilai = 0;
18            rataNilai = 0;
19            for (j = 1; j <= 5; j++) {
20                System.out.print("\tnilai dari kelompok Penilai " + j + ": ");
21                nilai = sc.nextInt();
22
23                totalNilai += nilai;
24            }
25
26            rataNilai = totalNilai / 5;
27            System.out.println("Kelompok " + i +": nilai rata-rata =" + rataNilai);
28            i++;
29        }
30    }
31}
```

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

1. Deklarasikan dulu variabel *totalNilai* menjadi double, lalu reset nilai *totalNilai* di outerloop. Inisialisasi 0 untuk mereset nilai dan jika berada di inner loop, total nilai akan selalu direset, dan menyebabkan rata nilai menjadi tidak sesuai.

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

if (rataNilai > tertinggi){
    tertinggi = rataNilai;
    hebat = i;
}

i++;
}

System.out.println("nilai tertinggi diraih oleh kelompok: " + tertinggi + " oleh kelompok " + hebat);
}

```

S OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
GitHub Utilities\PraktikumDaspro> PraktikumDaspro> & "C:\Program Files\Java\jdk-25\bin\java.exe" "-agentlib:jdwp=transport=dt
:\Users\IronPC\appData\Roaming\Code\User\workspaceStorage\6b5f3dfda40556cd6a75b7e1599bc213\vrehat\java\jd_w\PraktikumDaspr
Nilai dari kelompok Penilai 5: 82
k 2: nilai rata-rata -86.2
k 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
k 3: nilai rata-rata -88.2
k 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
k 4: nilai rata-rata -77.4
k 5
Nilai dari kelompok Penilai 1: 89
Nilai dari kelompok Penilai 2: 82
Nilai dari kelompok Penilai 3: 81
Nilai dari kelompok Penilai 4: 77
Nilai dari kelompok Penilai 5: 83
k 5: nilai rata-rata -80.6
k 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
k 6: nilai rata-rata -88.8
```

2. tertinggi diraih oleh kelompok 6

3. sudah