

SENIN 3 NOVEMBER 2025

**LAPORAN HASIL PRAKTIKUM
Jobsheet 8**



Disusun oleh:

Rafif Rizdan Prastana

Kelas 1H/IT

254107020052

**PROGRAM STUDI D-IV TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2025**

2. Praktikum

2.1 Percobaan 1: Review Perulangan 1

Waktu Percobaan: 30 menit

Percobaan ini ditujukan untuk me-review kembali perulangan yang telah dibahas pada pertemuan sebelumnya. Pada percobaan 1 akan dibuat program untuk membuat tampilan * sebanyak n kali ke arah samping.

1. Buat folder baru pada repositori lokal Anda, beri nama jobsheet8
2. Buat file baru StarNoPresensi.java
3. Buat fungsi/method main() di dalamnya.
4. Karena program membutuhkan input dari keyboard, maka perlu import class Scanner. Jadi tambahkan sintaks import di baris atas sendiri program.
5. Di dalam fungsi main() yang telah dibuat, deklarasikan objek Scanner dengan nama sc
6. Pada baris selanjutnya, tampilkan instruksi untuk memasukkan nilai yang akan disimpan ke variabel n.

```
1 System.out.print("Masukkan nilai n = ");
2 int n = sc.nextInt();
```

7. Pada baris selanjutnya, buat sintaks perulangan dengan for seperti di bawah ini.

```
1 for (int i=1; i<=n; i++) {
2     System.out.println("*");
3 }
```

8. Compile dan jalan program.
9. Amati hasilnya, maka hasilnya harusnya akan serupa dengan tampilan di bawah ini.

```
Masukkan nilai n = 5
*****
rizzdann@192 ~ %
```

10. Commit program Anda ke Github dengan pesan “Percobaan 1”

jobsheet8 Public

main 1 Branch 0 Tags

rizzdann Percobaan 1 dd2e6cc · now 1 Commit

Star22.java Percobaan 1 now

README

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 $i++ 2$, bagaimana pola outputnya jika input $n = 6$? Apa yang menyebabkan perubahan tersebut?

Jawaban

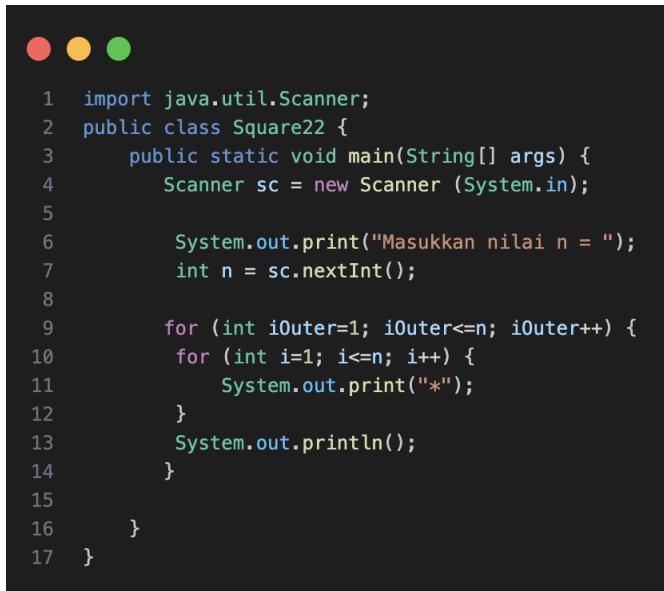
1. loop akan mulai dari 0 bukan 1, sehingga menjadi 0 1 2 3 4 5
2. loop menjadi kurang 1, karena angka 5 tidak terbaca
3. pada loop pertama $i = 1$, dan jika $i < n$ maka false
4. perulangan tidak akan berhenti karena $i \leq 5$ dan jika 1 – jawaban akan selalu true
5. hasil output adalah 1 3 5 karena $i++ 2$ menjadi lompat 2 langkah

2.2 Percobaan 2: Bintang Persegi

Waktu Percobaan: 30 menit

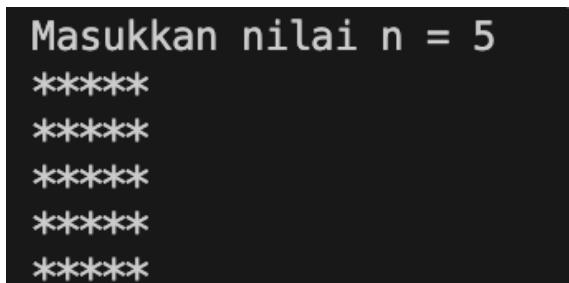
1. Pada percobaan ke-2 akan dilakukan percobaan tentang nested loop. Kasus yang akan diselesaikan adalah untuk membuat tampilan bujursangkar *, dengan panjang sisi sebanyak n . Misalkan n dimasukan 5, maka hasilnya adalah
2. Kalau diamati lebih lanjut, sebenarnya mirip dengan kasus percobaan 1. Jika di percobaan 1, misal input n bernilai 5, maka yang akan dihasilkan adalah ***** (bisa dianggap ini sebagai inner loop yang mencetak 5 bintang *****), maka untuk kasus percobaan 2 ini, hasil dari percobaan 1 tersebut hanya perlu diulang lagi sebanyak n kali. Yaitu dengan menambahkan outer loop untuk mengulangi proses inner loop sebanyak n kali
3. Buat file baru `SquareNoPresensi.java`
4. . Karena program membutuhkan input dari keyboard, maka perlu import class `Scanner`. Jadi tambahkan sintaks import di baris atas sendiri program.
5. Buat method `main()`, dan isikan kode program yang sama dengan isi method `main()` di percobaan 1.
6. Compile dan jalankan program. Pastikan program berjalan seperti saat menjalankan percobaan 1.
7. Perhatikan sintaks perulangan yang digunakan untuk mencetak * sebanyak n kali ke arah samping. Di langkah 8 di atas kode `for` (kotak merah) dijadikan sebagai inner loop.

- Perulangan pada inner loop dilakukan sebanyak n kali untuk menghasilkan output seperti pada poin 1. Maka perlu ditambahkan perulangan luar (outer loop).



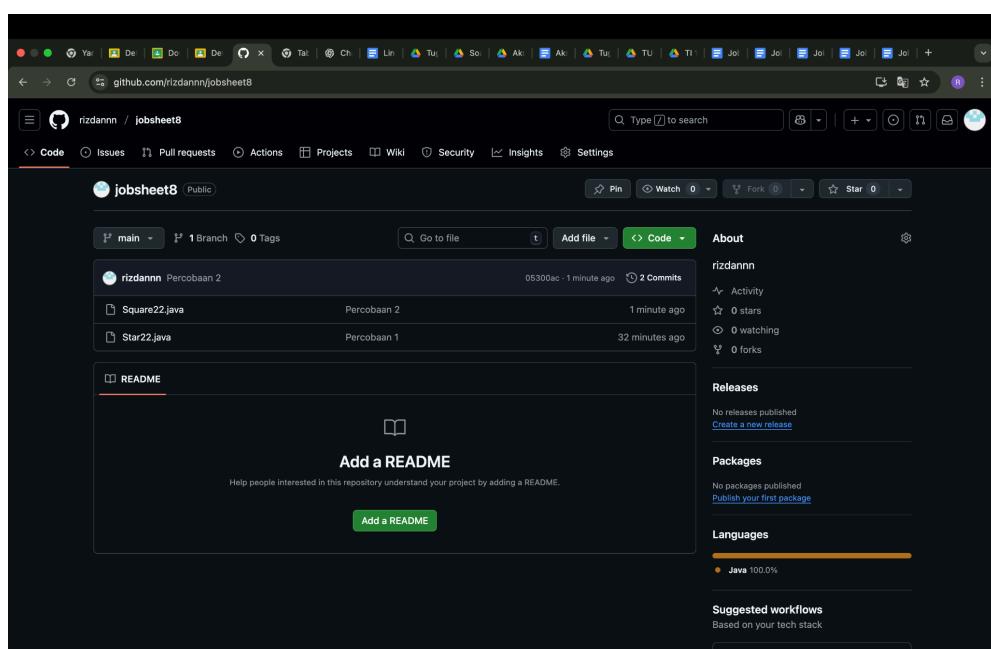
```
1 import java.util.Scanner;
2 public class Square22 {
3     public static void main(String[] args) {
4         Scanner sc = new Scanner (System.in);
5
6         System.out.print("Masukkan nilai n = ");
7         int n = sc.nextInt();
8
9         for (int iOuter=1; iOuter<=n; iOuter++) {
10             for (int i=1; i<=n; i++) {
11                 System.out.print("*");
12             }
13             System.out.println();
14         }
15     }
16 }
17 }
```

- Simpan perubahan, compile dan jalankan program.
- Amati hasilnya, maka hasilnya harusnya akan serupa dengan tampilan di bawah ini.



```
Masukkan nilai n = 5
*****
*****
*****
*****
*****
*****
```

- Commit program Anda ke Github dengan pesan “Percobaan 2”

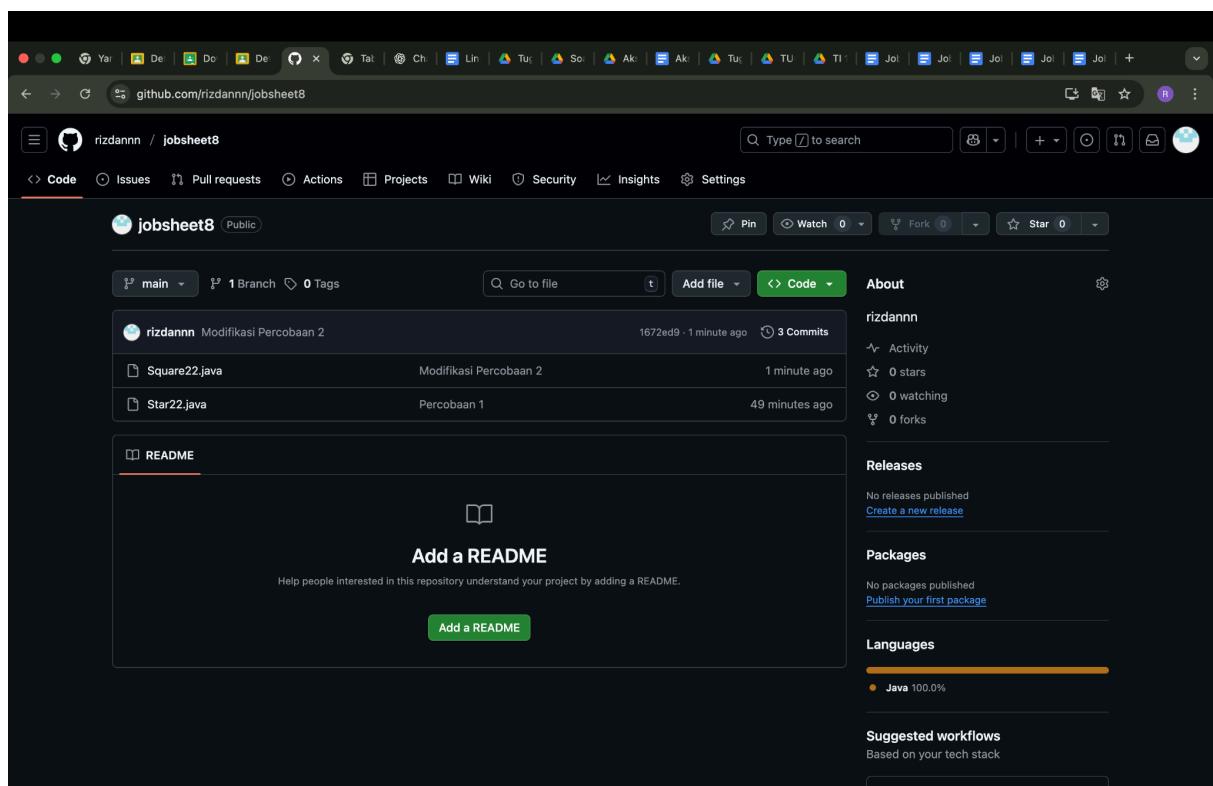


Pertanyaan

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?

Jawaban

1. penghitungan manjadi 0 sampai 5 dan menjadi 6 baris yang sebelumnya 5 baris
2. jumlah kolom bertambah 1
3. outer loop untuk baris sedangkan inner loop untuk kolom
4. hasil cetak akan menyambung semua di 1 baris
5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 2”



2.3 Percobaan 3: Bintang Segitiga

Waktu Percobaan: 45 menit

1. Pada percobaan ke-3 akan dilakukan percobaan segitiga *, dengan sama siku dengan tinggi sebesar n. Misalkan n dimasukan 5, maka hasilnya adalah:

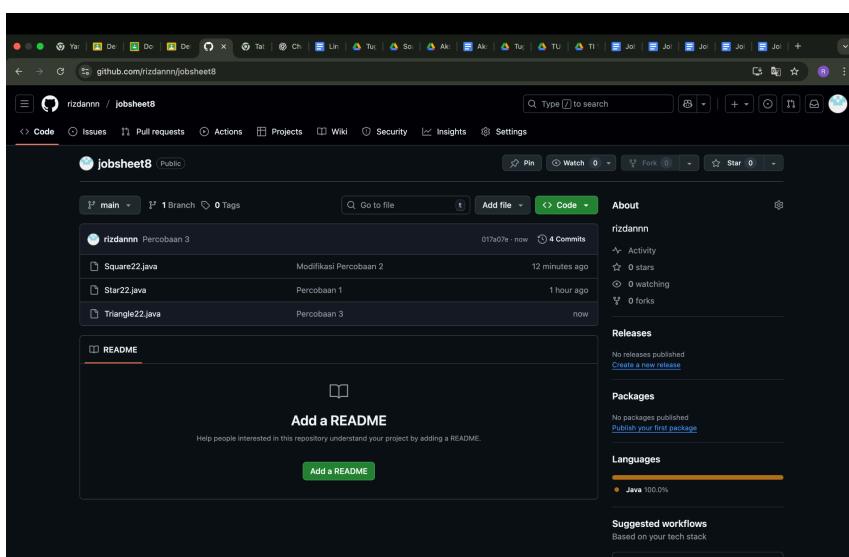
```
55605a301b87fb13a29100d/recurse
Masukkan nilai N = 5

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

2. Buat file baru TriangleNoPresensi.java
3. Karena program membutuhkan input dari keyboard, maka perlu import class Scanner.
4. Buat method **main()**, dan isikan kode program berikut kedalam method **main()**.

```
1 System.out.print("Masukkan nilai N = ");
2 int n = sc.nextInt();
3 int i = 0;
4
5 while(i <= n) {
6     int j = 0;
7     while(j < i) {
8         System.out.print("*");
9         j++;
10    }
11    System.out.println("");
12    i++;
13 }
```

5. Compile dan jalankan program. Amati apa yang terjadi.
6. Commit program Anda ke Github dengan pesan “Percobaan 3”



2.4 Percobaan 4: Studi Kasus Nilai Tugas Proyek Kelompok

Waktu Percobaan: 75 menit

Dalam pembelajaran kolaboratif, dosen menerapkan sistem peer review untuk menilai tugas kelompok. Setiap kelompok akan dinilai oleh beberapa kelompok lain sebagai bentuk umpan balik sejawat. Untuk setiap kelompok akan dinilai oleh beberapa kelompok lainnya. Pada percobaan ini, program akan memproses nilai untuk kelas yang berjumlah 6 kelompok. Masing-masing kelompok akan dinilai oleh 5 kelompok yang lain, dalam skala 1-100. Program akan menghitung nilai akhir setiap kelompok sebagai rata-rata dari semua penilai. Berikut adalah flowchart yang menunjukkan algoritma dari penyelesaian masalah tersebut.

Berdasarkan flowchart tersebut, buat program menggunakan bahasa pemrograman Java.

Langkah-langkah Percobaan

1. Buat file baru NilaiKelompokNoPresensi.java
2. Buatlah struktur dasar program Java yang terdiri dari fungsi main().
3. Tambahkan library Scanner di bagian atas (luar) class
4. Buat deklarasi Scanner dengan nama variabel sc di dalam fungsi main()
5. Deklarasikan variabel sesuai pada flowchart
6. Buat struktur perulangan untuk inner loop menggunakan for untuk memasukkan 5 nilai dari masing-masing kelompok penilai

```
1  for (int j = 1; j <= 5; j++) {
```

7. Di dalam perulangan for tersebut, tambahkan perintah untuk memasukkan 5 nilai dari kelompok penilai dan menghitung total nilainya. Jangan lupa untuk memberikan nilai awal 0 pada totalNilai sebelum perulangan.

```
1  totalNilai = 0;
2      for (int j = 1; j <= 5; j++) {
3          System.out.print(" Nilai dari Kelompok Penilai " + j + ": ");
4          nilai = sc.nextInt();
5          totalNilai += nilai;
6      }
7
```

8. Setelah proses perulangan memasukkan 5 nilai dan diperoleh total nilai, maka hitung rata-rata nilai dengan rumus
9. Selanjutnya adalah mengulang proses memasukkan nilai tersebut untuk 6 kelompok. Buat struktur perulangan untuk outer loop menggunakan while.

```
1  totalNilai = 0;
2  for (int j = 1; j <= 5; j++) {
3      System.out.print(" Nilai dari Kelompok Penilai " + j + ": ");
4      nilai = sc.nextInt();
5      totalNilai += nilai;
6  }
7
8  rataNilai = totalNilai / 5;
9  System.out.println("Rata-rata nilai kelompok " + i + ": " + rataNilai);
10 System.out.println();
11
12     i++;
13 }
```

10. Tambahkan narasi keterangan untuk kebutuhan masukan dan luaran
11. Compile dan run program.
12. Amati hasilnya, apakah program telah menjalankan perintah memasukkan 5 nilai untuk 6 kelompok dengan tampilan serupa dengan di bawah ini:

```
Nilai dari Kelompok Penilai 2: 79
Nilai dari Kelompok Penilai 3: 90
Nilai dari Kelompok Penilai 4: 82
Nilai dari Kelompok Penilai 5: 85
Rata-rata nilai kelompok 1: 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
Rata-rata nilai kelompok 2: 86.2

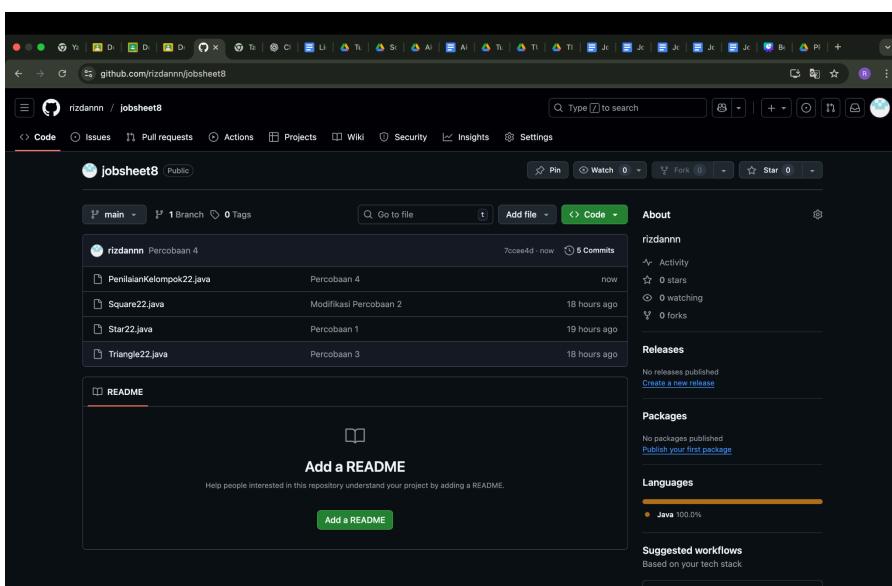
Kelompok 3
Nilai dari Kelompok Penilai 1: 90
Nilai dari Kelompok Penilai 2: 91
Nilai dari Kelompok Penilai 3: 86
Nilai dari Kelompok Penilai 4: 84
Nilai dari Kelompok Penilai 5: 90
Rata-rata nilai kelompok 3: 88.2

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
Rata-rata nilai kelompok 4: 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
Rata-rata nilai kelompok 5: 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
Rata-rata nilai kelompok 6: 88.8
```

13. Commit program Anda ke Github dengan pesan “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”

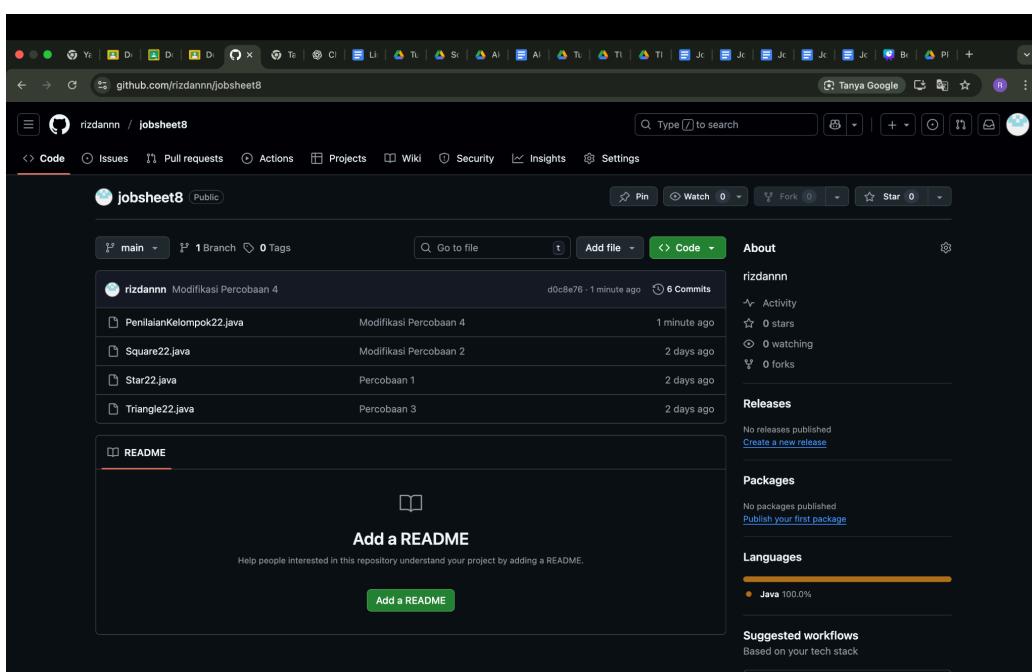
Jawaban

1. fungsi `total = 0` adalah untuk mereset nilai menjadi 0 kembali, maka jika `total = 0` diluar loop maka nilai akan terus ditambahkan.



```
1 double rataTertinggi = 0;
2 int kelompokTerbaik = 0;
3
4
5 if (rataNilai > rataTertinggi) {
6     rataTertinggi = rataNilai;
7     kelompokTerbaik = i - 1;
8 }
9
10 System.out.println("\nKelompok dengan rata-rata tertinggi adalah Kelompok ke-"
11                     + kelompokTerbaik + " dengan rata-rata " + rataTertinggi);
```

2.



3.

3. Tugas

Waktu Percobaan : 120 Menit

1. 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.



```
1 import java.util.Scanner;
2
3 public class Kuadrat22 {
4     public static void main(String[] args) {
5         Scanner input = new Scanner(System.in);
6
7         System.out.print("Masukkan nilai n: ");
8         int n = input.nextInt();
9
10        for (int x = 1; x <= n; x++) {
11            int total = 0;
12            System.out.print("n = " + x + " -> jumlah kuadrat = ");
13
14            for (int i = 1; i <= x; i++) {
15                int kuadrat = i * i;
16                total += kuadrat;
17
18                System.out.print(kuadrat);
19                if (i < x) {
20                    System.out.print(" + ");
21                }
22            }
23            System.out.println(" = " + total);
24        }
25    }
26
27 }
```

Output:

```
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
○ rizdann@192 jobsheet8-1 % █
```

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

```
● ● ●
1 import java.util.Scanner;
2
3 public class Tugas2 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         System.out.print("Masukkan nilai n (Minimal 3) = ");
8         int n = sc.nextInt();
9
10        if (n < 3) {
11            System.out.println("Nilai harus minimal 3");
12            sc.close();
13            return;
14        }
15
16        for (int i = 1; i <= n; i++) {
17            for (int j = 1; j <= n; j++) {
18                if ( i == 1 || i == n || j == 1 || j == n ) {
19                    System.out.print(n + " ");
20                } else {
21                    System.out.print(" ");
22                }
23            }
24            System.out.println();
25        }
26
27        sc.close();
28    }
29 }
```

Output:

```
Masukkan nilai n (Minimal 3) = 3
3 3 3
3   3
3 3 3
rizdann@192 jobsheet8-1 % cd /User
va -agentlib:jdwp=transport=dt_soc
tion\ Support/Code/User/workspaceS
Masukkan nilai n (Minimal 3) = 5
5 5 5 5 5
5      5
5      5
5      5
5 5 5 5 5
rizdann@192 jobsheet8-1 % █
```

3. 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:



```
1 import java.util.Scanner;
2 public class Tugas3 {
3     public static void main(String[] args) {
4         Scanner sc = new Scanner(System.in);
5
6         System.out.print("Jumlah Cabang kafe: ");
7         int jumlahCabang = sc.nextInt();
8
9         int totalPelanggan = 0;
10        int totalItem = 0;
11
12        System.out.println("\n==== Input Penjualan Per Cabang ====");
13
14        for (int i = 1; i <= jumlahCabang; i++) {
15            System.out.println("\n-- Cabang " + i + " --");
16            System.out.print("Jumlah pelanggan: ");
17            int jumlahPelanggan = sc.nextInt();
18
19            totalPelanggan += jumlahPelanggan;
20            int totalItemCabang = 0;
21
22            for (int j = 1; j <= jumlahPelanggan; j++) {
23                System.out.print(" - Pelanggan " + j + " memesan beberapa item? ");
24                int item = sc.nextInt();
25                totalItemCabang += item;
26            }
27
28            totalItem += totalItemCabang;
29
30            System.out.println("Cabang " + i + ": ");
31            System.out.println("- Pelanggan : " + totalPelanggan + " orang");
32            System.out.println("- Item terjual : " + totalItem);
33        }
34
35        System.out.println("\nTotal seluruh cabang: ");
36        System.out.println("Pelanggan: " + totalPelanggan + " orang");
37        System.out.println("Item terjual: " + totalItem + " Item");
38    }
39}
```

Output:

```
Jumlah Cabang kafe: 2

==== Input Penjualan Per Cabang ===

- - - Cabang 1 - - -
Jumlah pelanggan: 3
- Pelanggan 1 memesan beberapa item? 2
- Pelanggan 2 memesan beberapa item? 4
- Pelanggan 3 memesan beberapa item? 1
Cabang 1:
- Pelanggan : 3 orang
- Item terjual : 7

- - - Cabang 2 - - -
Jumlah pelanggan: 4
- Pelanggan 1 memesan beberapa item? 3
- Pelanggan 2 memesan beberapa item? 5
- Pelanggan 3 memesan beberapa item? 1
- Pelanggan 4 memesan beberapa item? 2
Cabang 2:
- Pelanggan : 7 orang
- Item terjual : 18

Total seluruh cabang:
Pelanggan: 7 orang
Item terjual: 18 Item
rizzdann@192 jobsheet8-1 %
```