**LAPORAN**

**PRAKTIKUM DASPRO PERTEMUAN 11**

****

**Nama :**

Ahmad Dzul Fadhli Hanna

**NIM :**

2341720106

**Kelas :**

TI-1H

**Absen :**

03

**Tujuan :**

1. Mahasiswa memahami konsep perulangan bersarang
2. Mahasiswa dapat menjelaskan format penulisan perulangan bersarang (nested loop)
3. Mahasiswa dapat mengimplementasikan flowchart perulangan bersarang menggunakan bahasa pemrograman Java

**Alat dan Bahan**

* PC/Laptop
* JDK
* Java IDE

**Praktikum**

1. **Percobaan 1**

**Praktikum : Review perulangan yang lalu**

Codingan

import java.util.Scanner;

public class Star03 {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Masukkan nilai N = ");

int N = sc.nextInt();

for(int i = 1; i <= N; i++) {

System.out.print("\*");

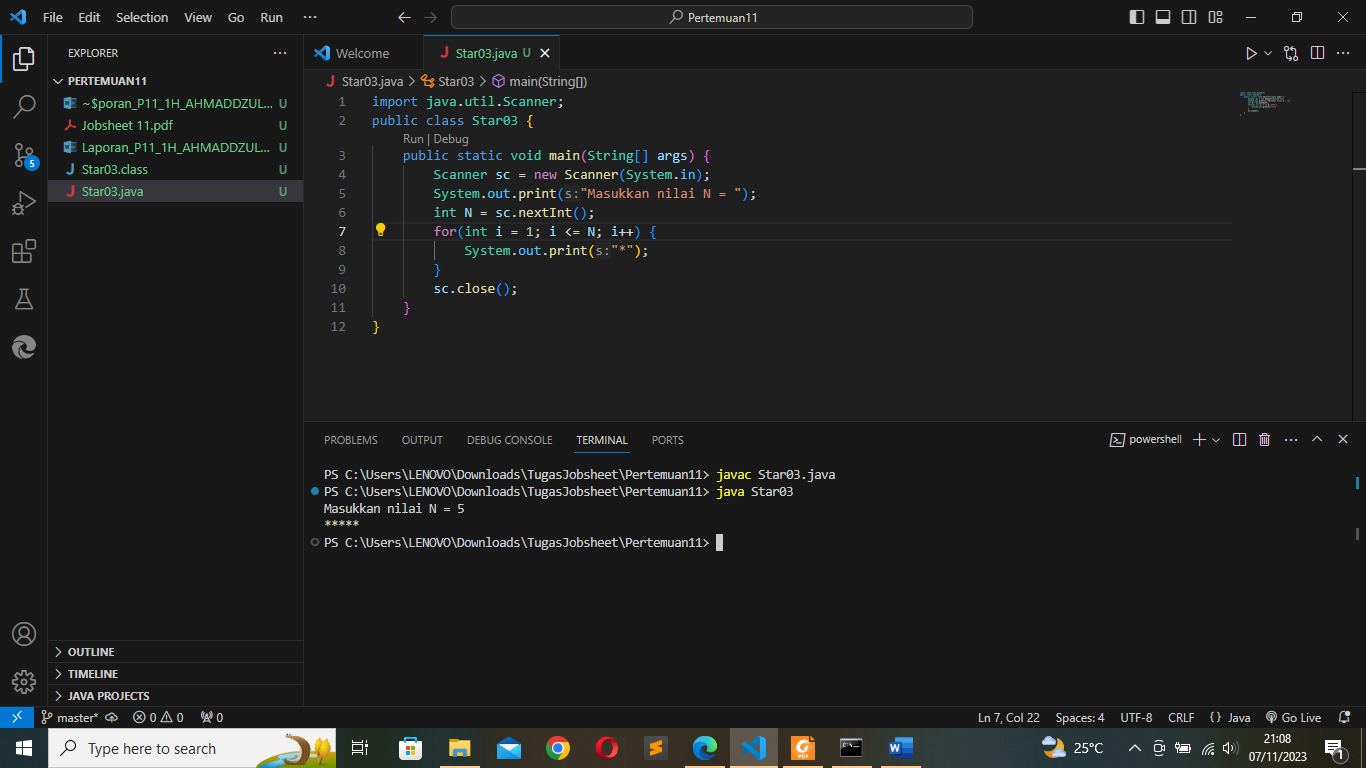
}

sc.close();

}

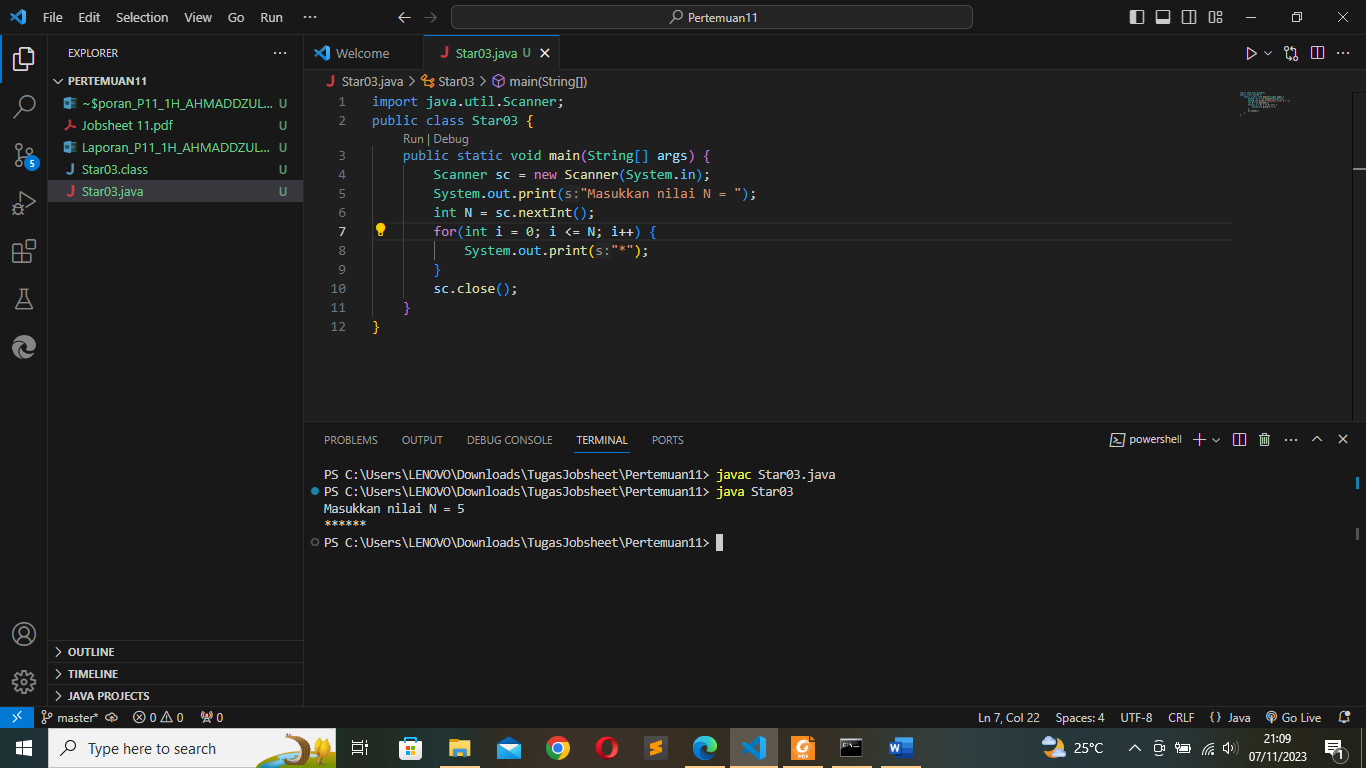
}

Hasil Running



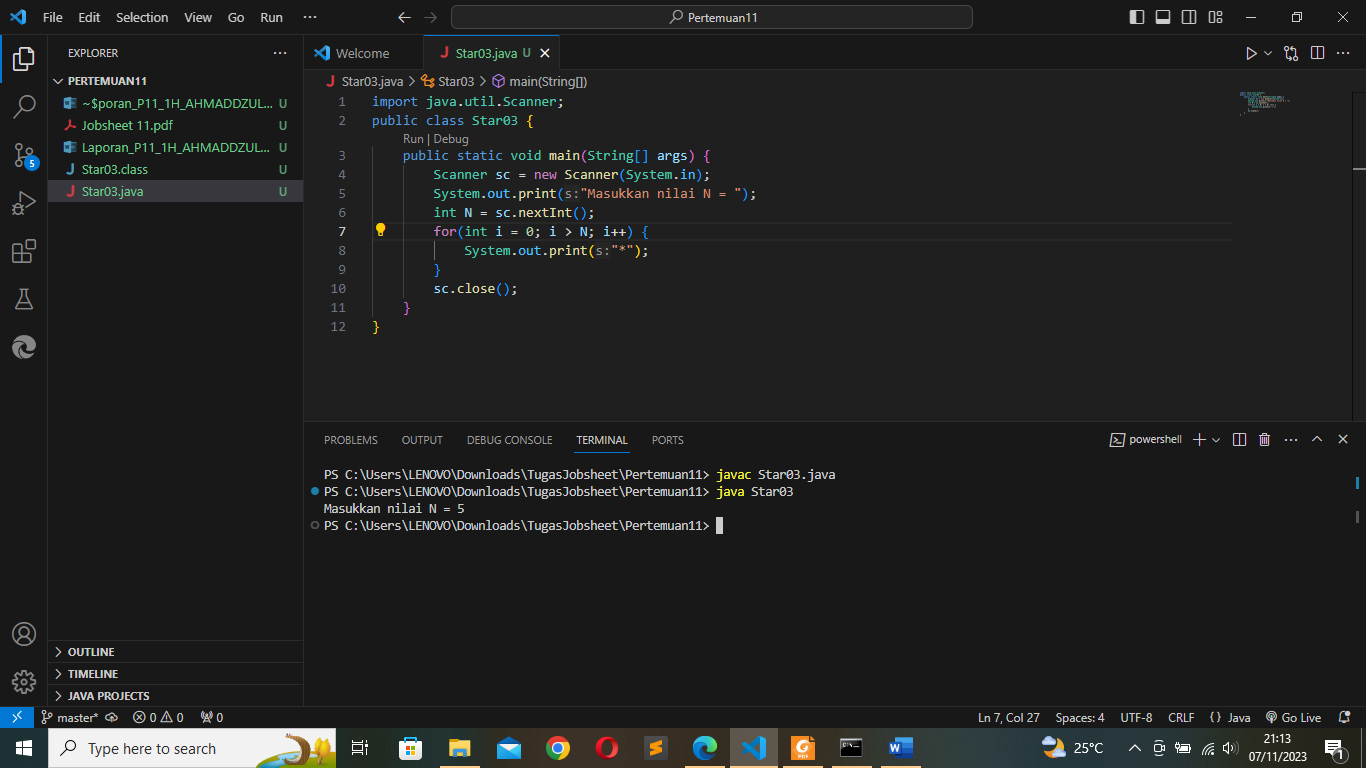
**Pertanyaan**

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

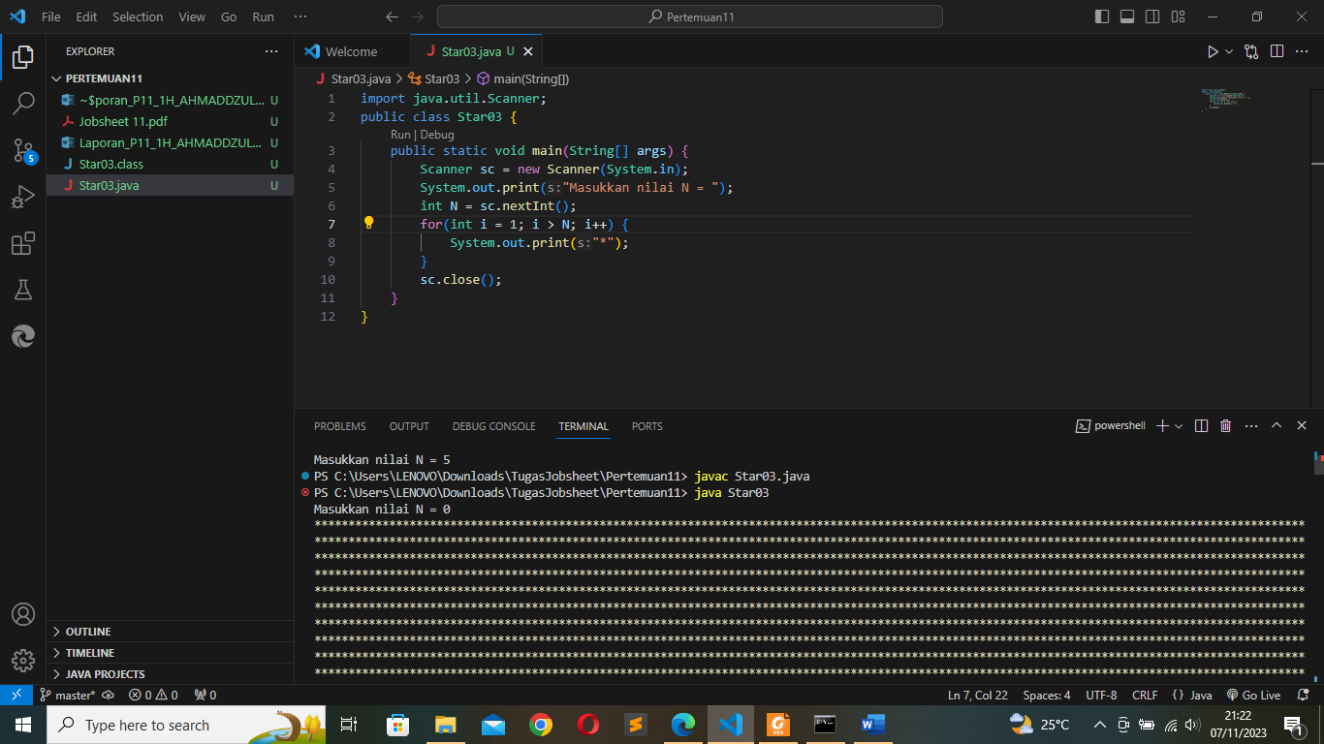


Jika i diubah nilainya menjadi 0, maka perulangan akan dimulai dari 0 dan selanjutnya.Misal jika input nilai N adalah 5, maka perulangan akan dijalankan ketika nilai i 0,1,2,3,4,5. Jumlah nilai i ada 6 sehingga perullangan akan terjadi 6 kali. Bisa dilihat dari jumlah output bintang ada 6 yang inputnya 5.

1. Jika pada perulangan for, kondisi i <= N diubah menjadi i > N, apa akibatnya? Mengapa bisa demikian?

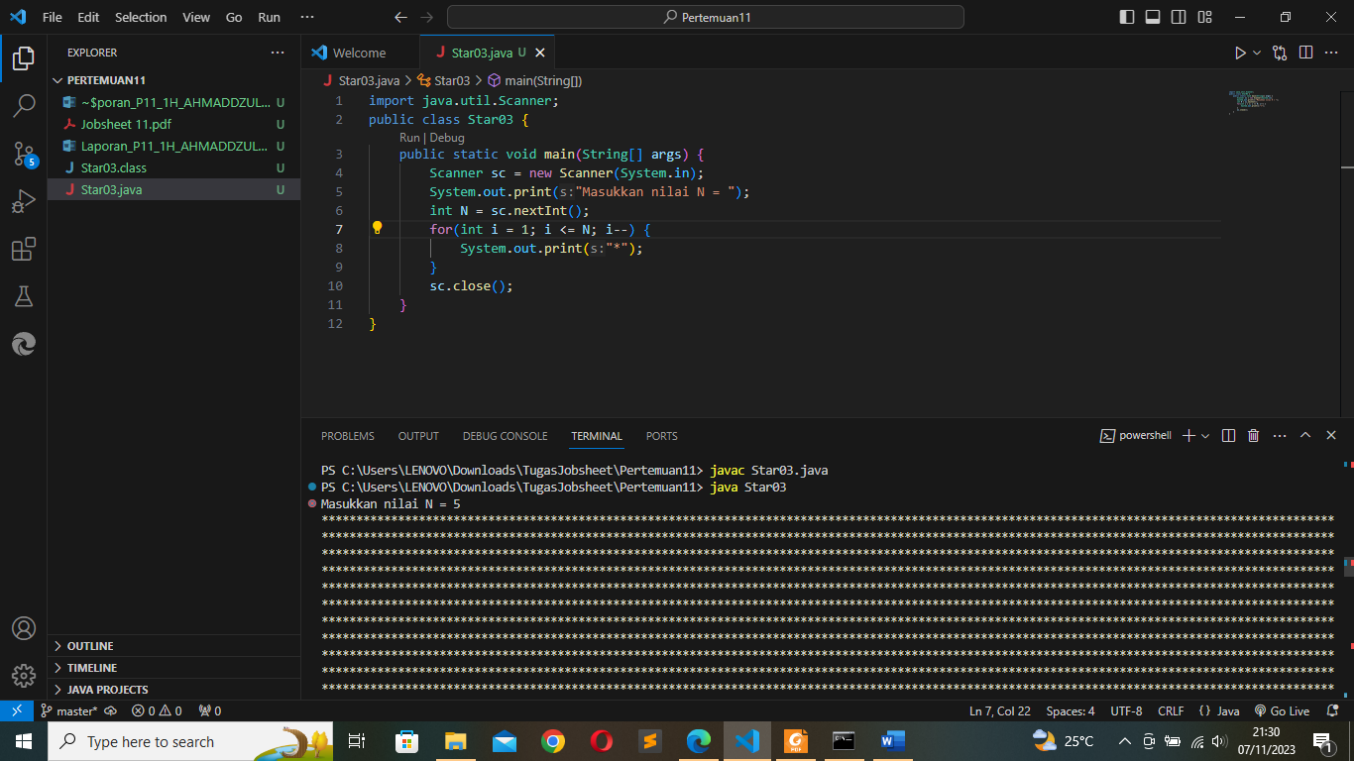


Maka program akan melakukan perulangan sebanyak i > N yang artinya perulangan akan terjadi sebanyak 0. Kecuali jika input nilai N adalah 0, maka perulangan tanpa batas.

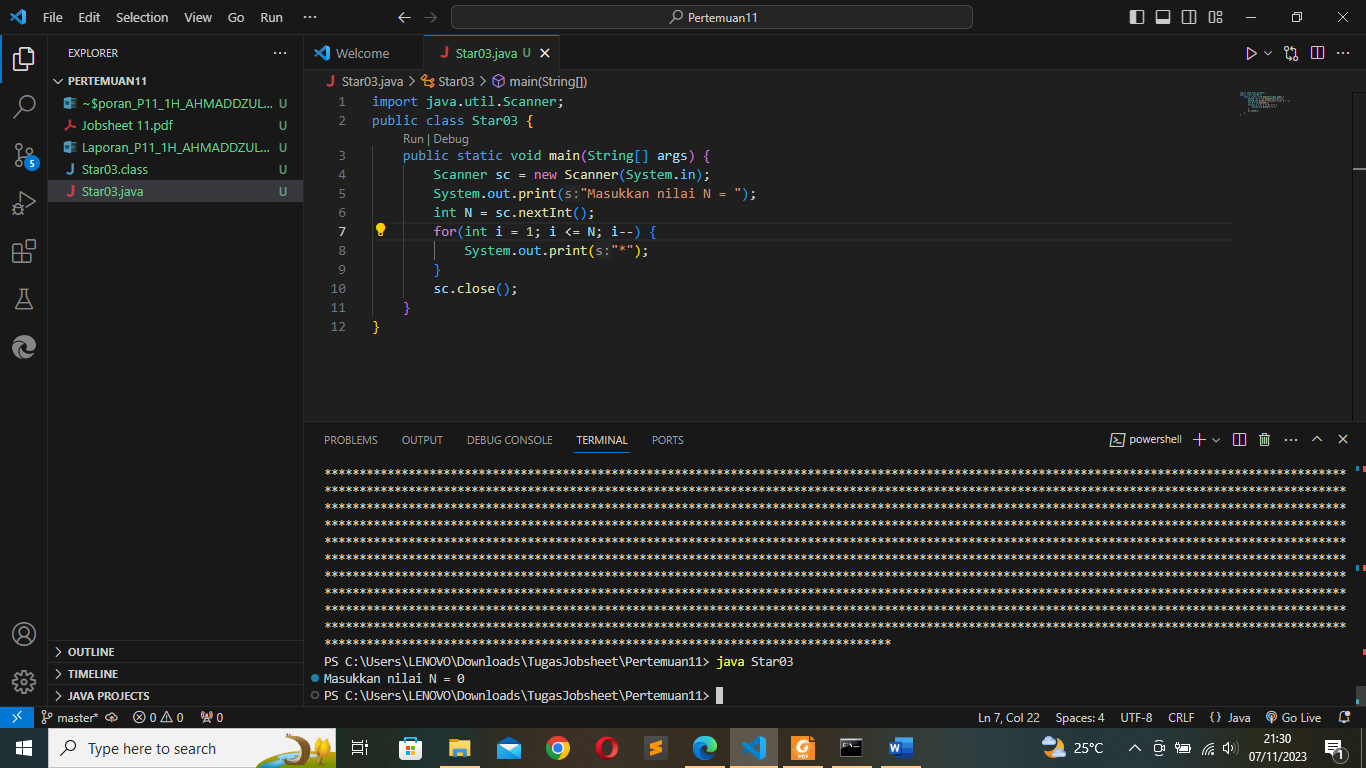


Itu karena i > N artinya perulangan akan berjalan hingga nilai i lebih besar dari nilai N.

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



Maka perulangan akan berjalan terus hingga i <= N, dan jika pada stataement 3 diubah menjadi i--, maka perulangan akan terus berjalan tanpa henti. Karena nilai i akan selalu kurang dari N jika input N adalah bilangan positif lebih atau sama dengan 1.



Dan jika input N adalah kurang dari 1, maka program akan melakukan perulangan sebanyak 0 kali.

1. **Percobaan 2 : Bintang Persegi**

**Praktikum**

Codingan

import java.util.Scanner;

public class Square03 {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Masukkan nilai N = ");

int N = sc.nextInt();

for (int iOuter = 1; iOuter <= N; iOuter++ ) {

for (int i = 1; i<= N; i++) {

System.out.print("\*");

}

System.out.println();

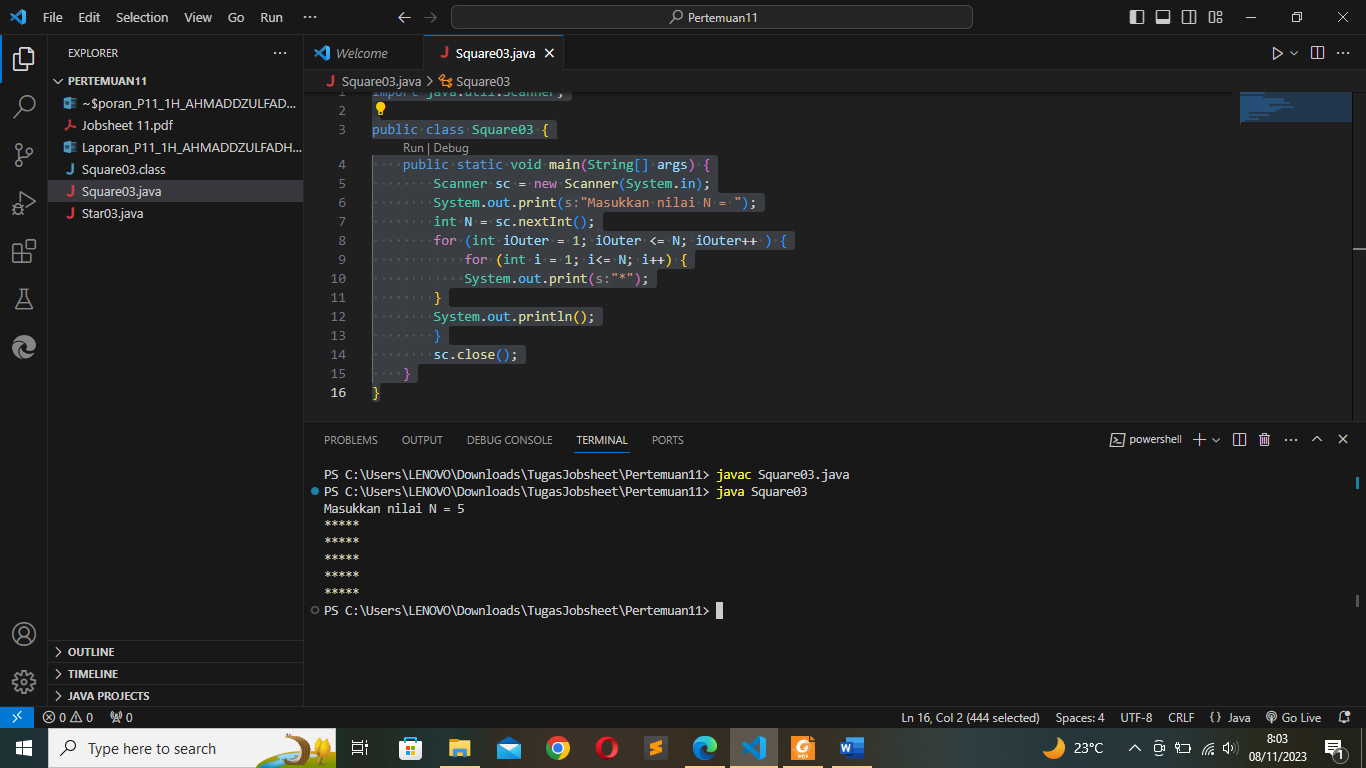
}

sc.close();

}

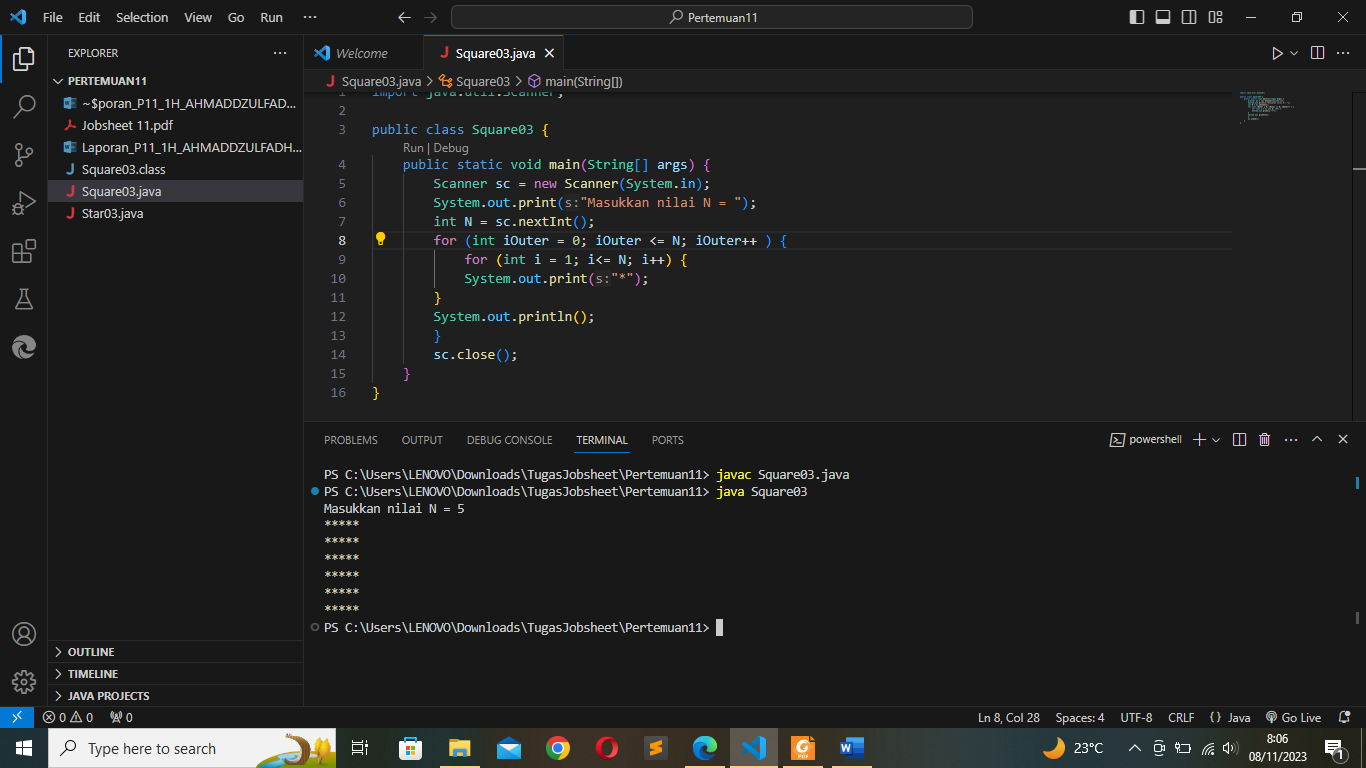
}

Hasi Running



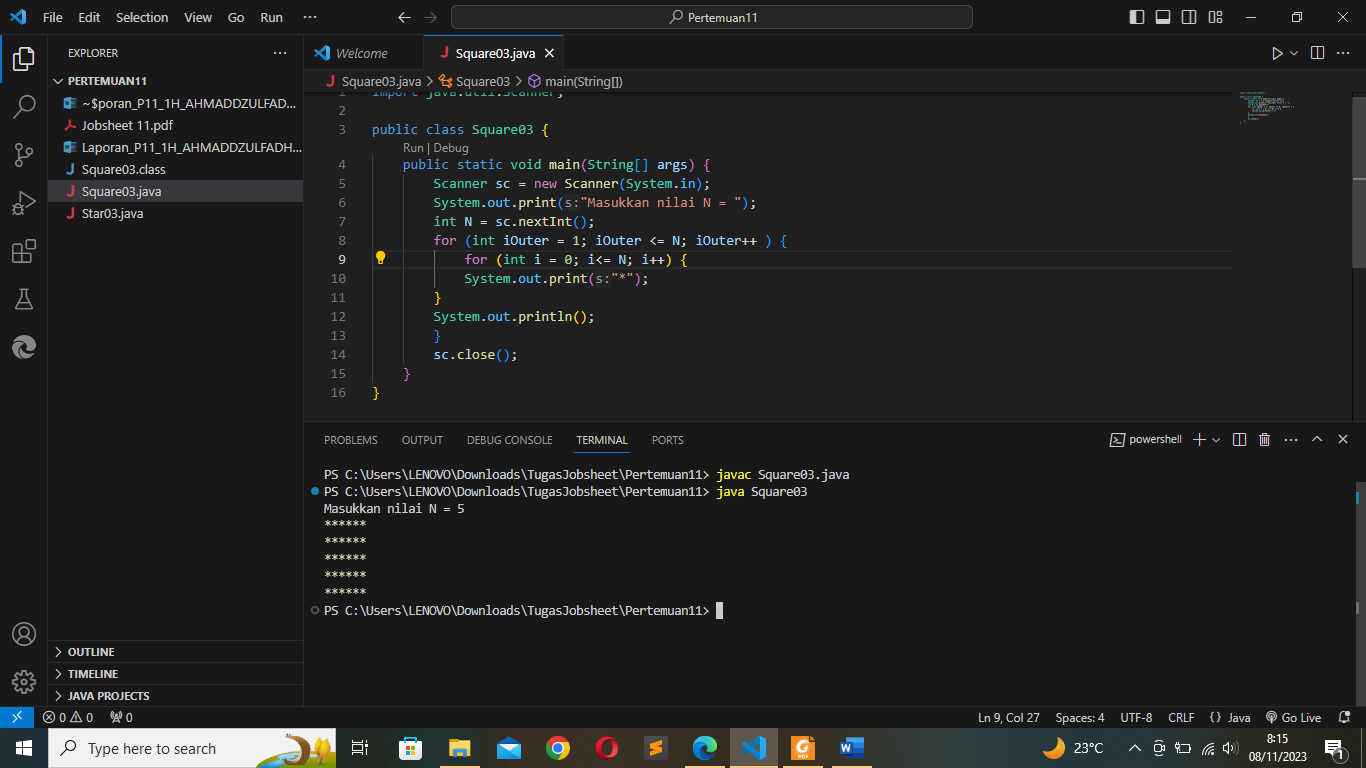
**Pertanyaan**

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



Jika iOuter diubah nilainya menjadi 0, maka perulangan akan dimulai dari 0 dan selanjutnya.Misal jika input nilai N adalah 5, maka perulangan akan dijalankan ketika nilai iOutput 0,1,2,3,4,5. Jumlah nilai iOutput ada 6 sehingga perullangan akan terjadi 6 kali. Bisa dilihat dari jumlah output baris bintang ada 6 yang inputnya 5.

1. 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?



Jika i diubah nilainya menjadi 0, maka perulangan akan dimulai dari 0 dan selanjutnya.Misal jika input nilai N adalah 5, maka perulangan akan dijalankan ketika nilai i 0,1,2,3,4,5. Jumlah nilai i ada 6 sehingga perullangan akan terjadi 6 kali. Bisa dilihat dari jumlah output kolom bintang ada 6 yang inputnya 5.

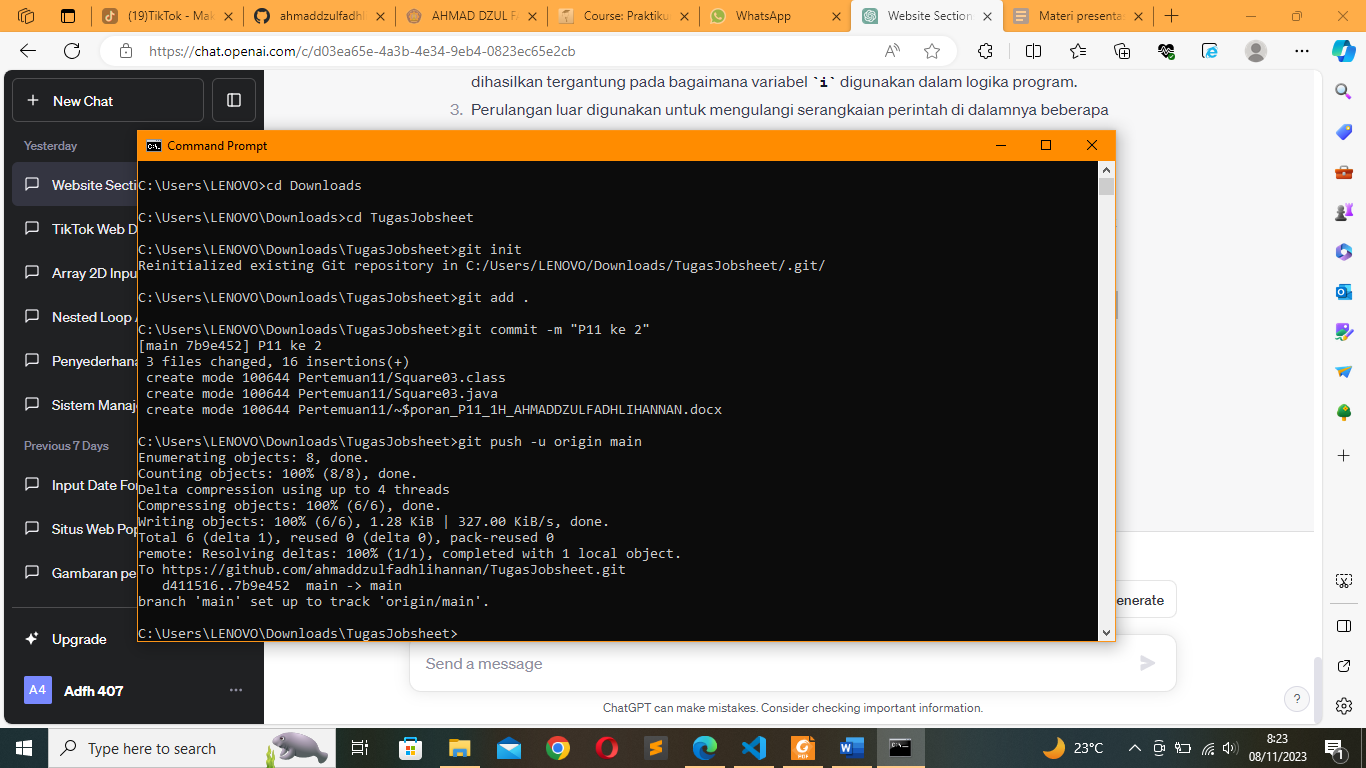
1. Jadi, apakah perbedaan kegunaan antara perulangan luar dengan perulangan yang berada didalamnya?

Perulangan luar digunakan untuk mengulangi serangkaian perintah di dalamnya beberapa kali, sementara perulangan dalam digunakan untuk menjalankan serangkaian perintah tertentu secara berulang tergantung pada kondisi yang diberikan di dalam perulangan tersebut. Perulangan luar memungkinkan pengulangan keseluruhan blok perintah, sedangkan perulangan dalam memungkinkan pengulangan perintah tertentu di dalam blok perintah.

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

System.out.println(); digunakan untuk mencetak baris kosong setelah mencetak bintang-bintang dalam satu baris. Jika sintaks tersebut dihilangkan, maka semua bintang akan dicetak dalam satu baris tanpa jeda baris, sehingga outputnya akan tampak seperti satu baris panjang dari bintang-bintang.

1. Silakan commit dan push ke repository Anda.





1. **Percobaan 3 : Bintang Segitiga**

**Praktikum**

1. Codingan

public class Numbers03 {

public static void main(String[] args) {

int[][] myNumbers = new int [3][];

myNumbers[0] = new int[5];

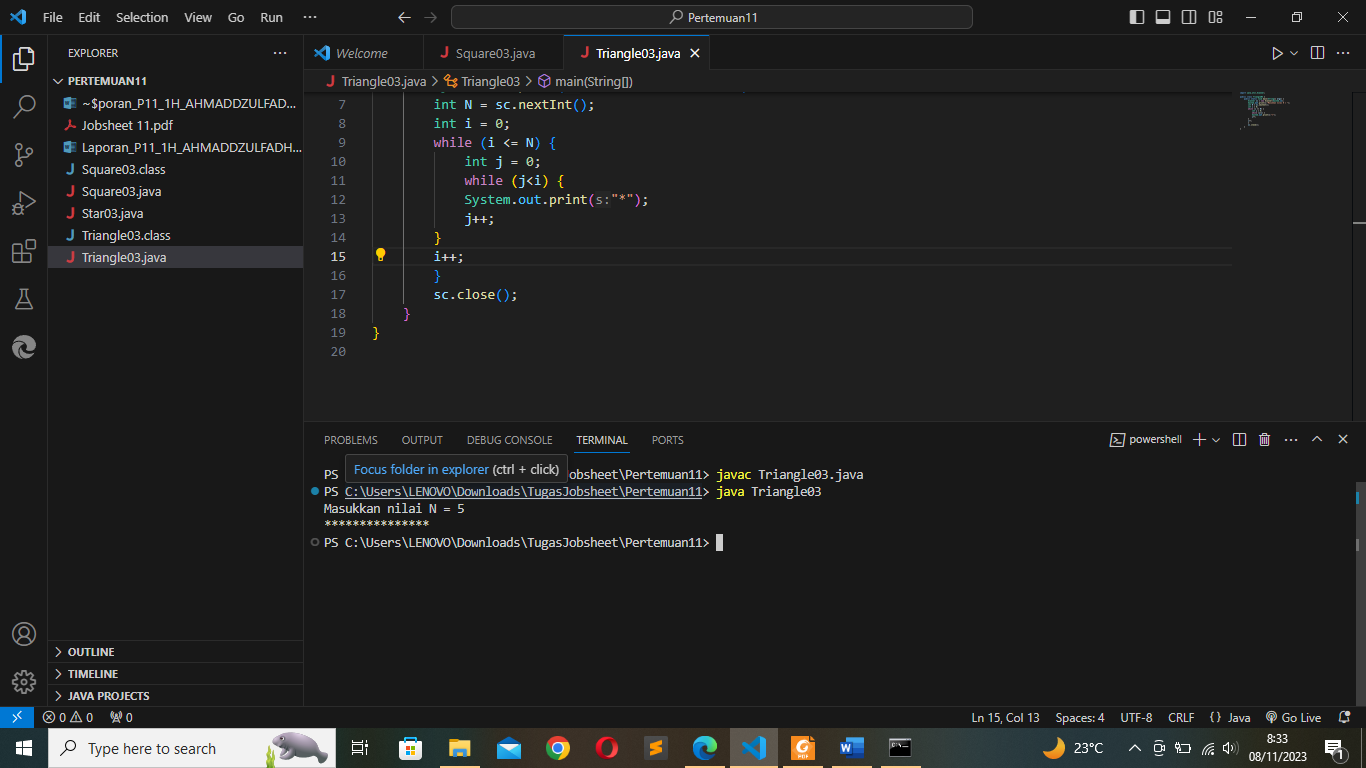
myNumbers[1] = new int[3];

myNumbers[2] = new int[1];

}

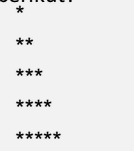
}

Hasil running

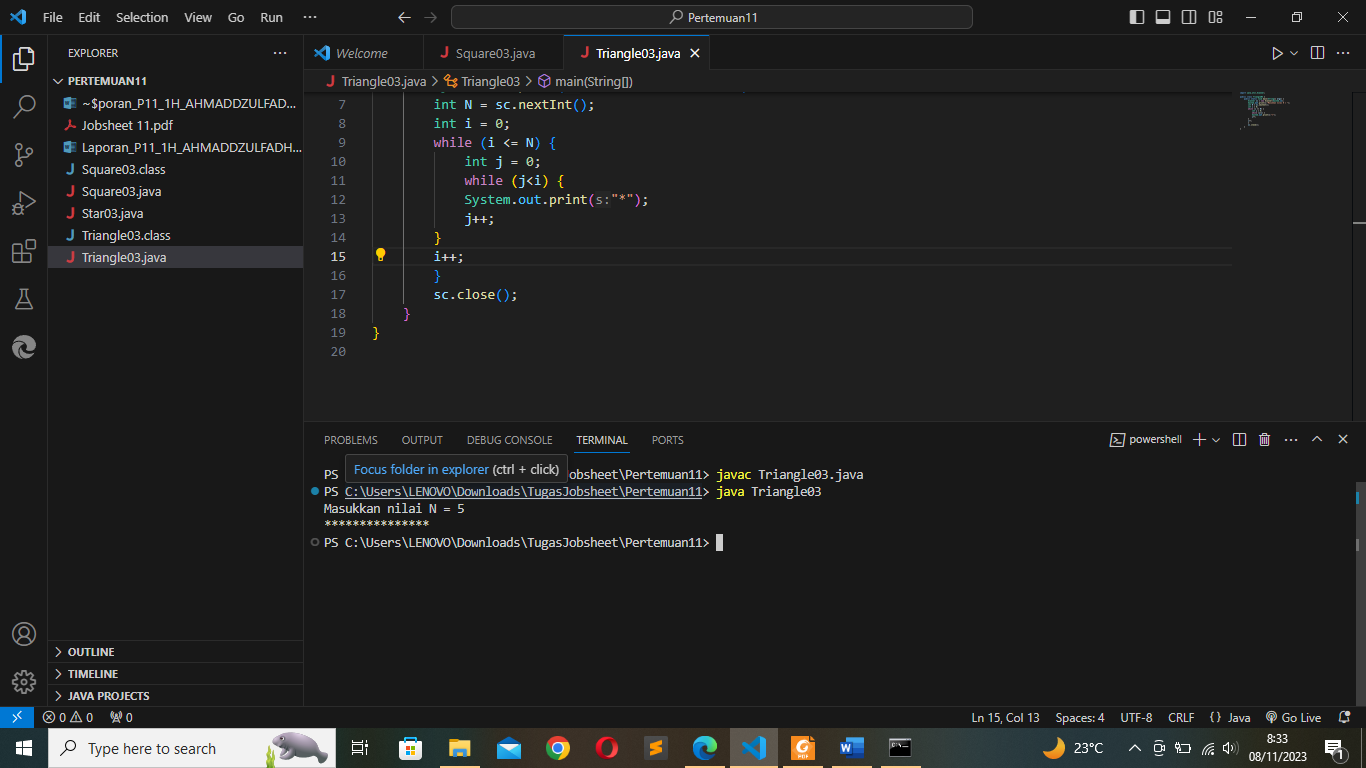


**Pertanyaan**

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



Tidak sesuai



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

Dengan menambahkan System.out.println(); pada perulangan outer dan setelah perulangan inner untuk menambahkan baris baru setelah perulangan inner dilakukan. Berikut adalah kode program yang sudah dimodifikasi :

import java.util.Scanner;

public class Triangle03 {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Masukkan nilai N = ");

int N = sc.nextInt();

int i = 0;

while (i <= N) {

int j = 0;

while (j<i) {

System.out.print("\*");

j++;

}

**System.out.println();**

i++;

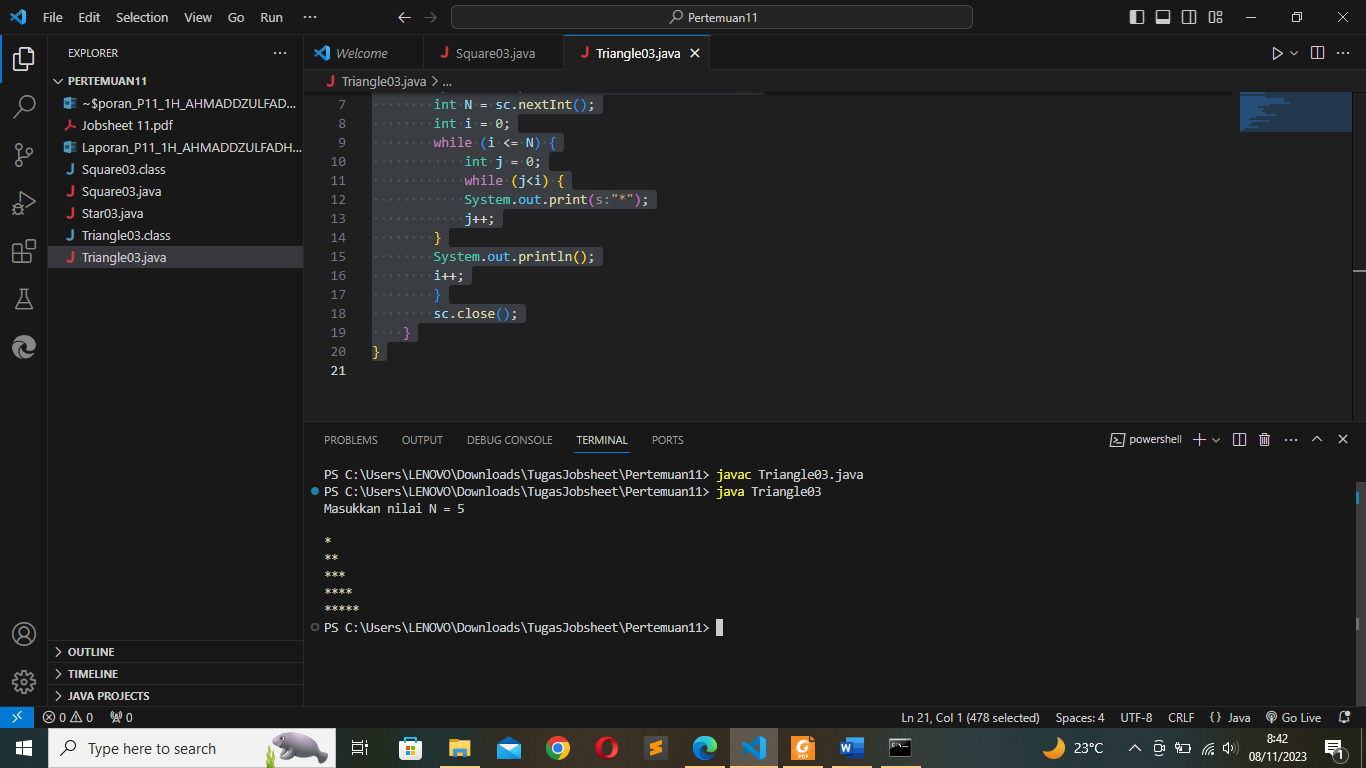
}

sc.close();

}

}

Hasil running



* 1. **Percobaan 4 : Kuis Tebak Angka**

**Praktikum**

Codingan

import java.util.Scanner;

import java.util.Random;

public class Quiz03 {

public static void main(String[] args) {

Random random = new Random();

Scanner input = new Scanner(System.in);

char menu = 'y';

do {

int number = random.nextInt(10) + 1;

boolean success = false;

do {

System.out.print("Tebak angka (1-10): ");

int answer = input.nextInt();

input.nextLine();

success = (answer == number);

} while (!success);

System.out.print("Apakah Anda ingin mengulang permainan (Y/y)? ");

menu = input.nextLine().charAt(0);

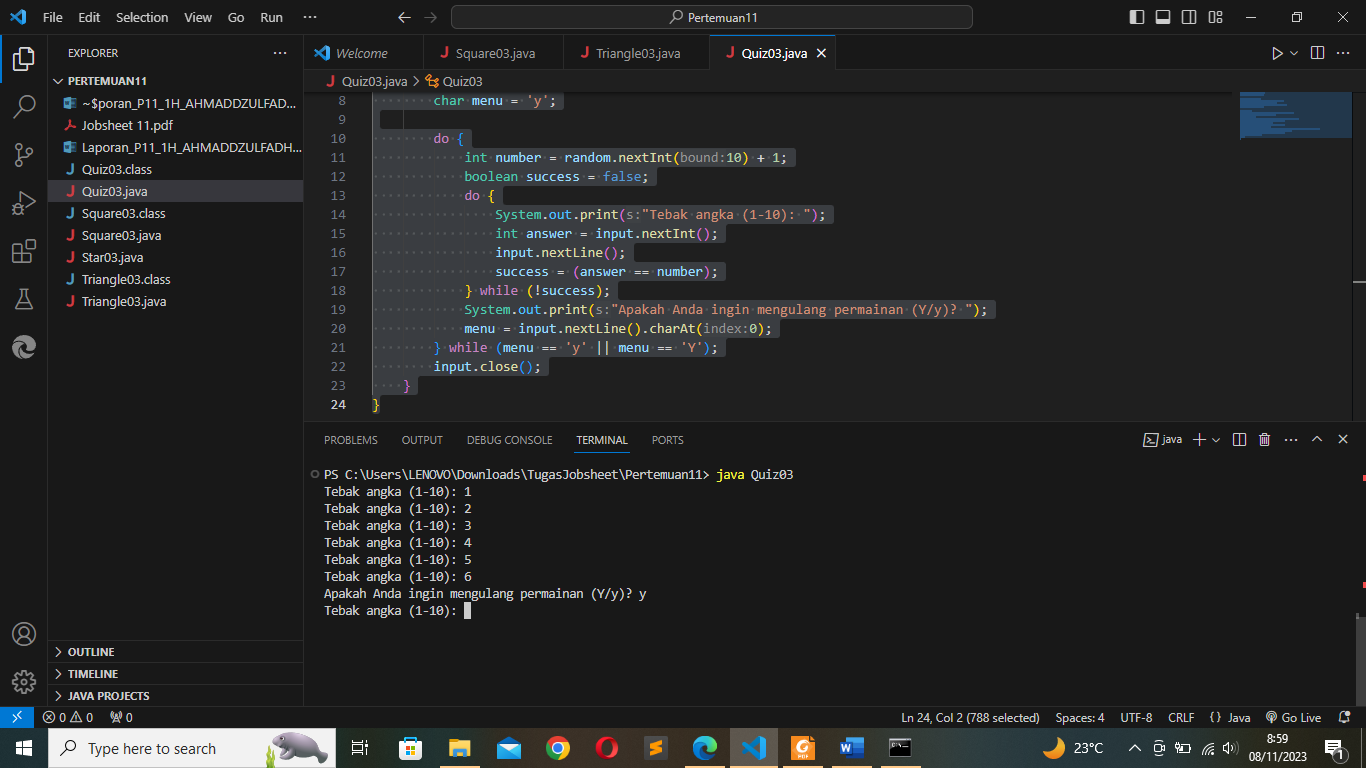
} while (menu == 'y' || menu == 'Y');

input.close();

}

}

Hasil running



**Pertanyaan**

1. Jelaskan alur program di atas!

* Program memulai dengan mendeklarasikan objek random dan input untuk menghasilkan angka acak dan menerima input pengguna.
* Program kemudian memulai permainan dengan memilih angka acak antara 1 hingga 10.
* Pengguna diminta untuk menebak angka.
* Jika tebakan benar, pengguna ditanya apakah mereka ingin mengulangi permainan.
* Jika pengguna setuju untuk mengulangi permainan, permainan dimulai kembali dengan angka acak baru.

1. Apa yang harus dilakukan untuk tidak melanjutkan (tidak mengulangi) permainan tersebut?

Dengan menambahkan kondisi if (menu != 'y' && menu != 'Y'), program akan mencetak pesan "Permainan dihentikan. Terima kasih telah bermain!" saat pengguna memilih untuk tidak melanjutkan permainan dengan memasukkan pilihan selain 'y' atau 'Y'. Hal ini akan menghentikan permainan setelah loop utama selesai dieksekusi.

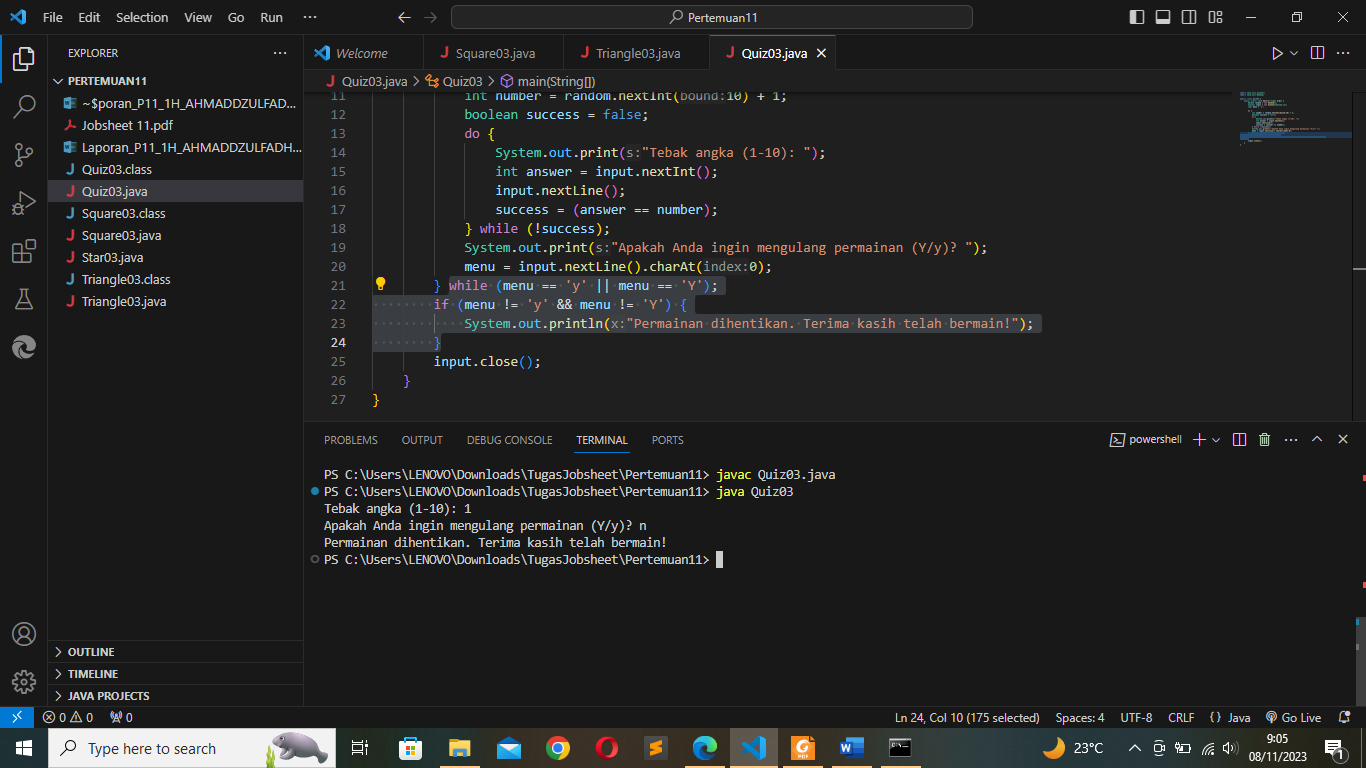
while (menu == 'y' || menu == 'Y');

**if (menu != 'y' && menu != 'Y') {**

**System.out.println("Permainan dihentikan. Terima kasih telah bermain!");**

**}**

Running



1. Modifikasi program di atas, sehingga bisa menampilkan informasi mengenai : input nilai tebakan yang dimasukan oleh user apakah lebih kecil atau lebih besar jawaban/number yang di random!

Kode yang dimodifikasi

import java.util.Scanner;

import java.util.Random;

public class Quiz03 {

public static void main(String[] args) {

Random random = new Random();

Scanner input = new Scanner(System.in);

char menu = 'y';

do {

int number = random.nextInt(10) + 1;

boolean success = false;

do {

System.out.print("Tebak angka (1-10): ");

int answer = input.nextInt();

input.nextLine();

if (answer < number) {

System.out.println("Tebakan terlalu kecil!");

} else if (answer > number) {

System.out.println("Tebakan terlalu besar!");

} else {

success = true;

}

} while (!success);

System.out.print("Apakah Anda ingin mengulang permainan (Y/y)? ");

menu = input.nextLine().charAt(0);

} while (menu == 'y' || menu == 'Y');

if (menu != 'y' && menu != 'Y') {

System.out.println("Permainan dihentikan. Terima kasih telah bermain!");

}

input.close();

}

}

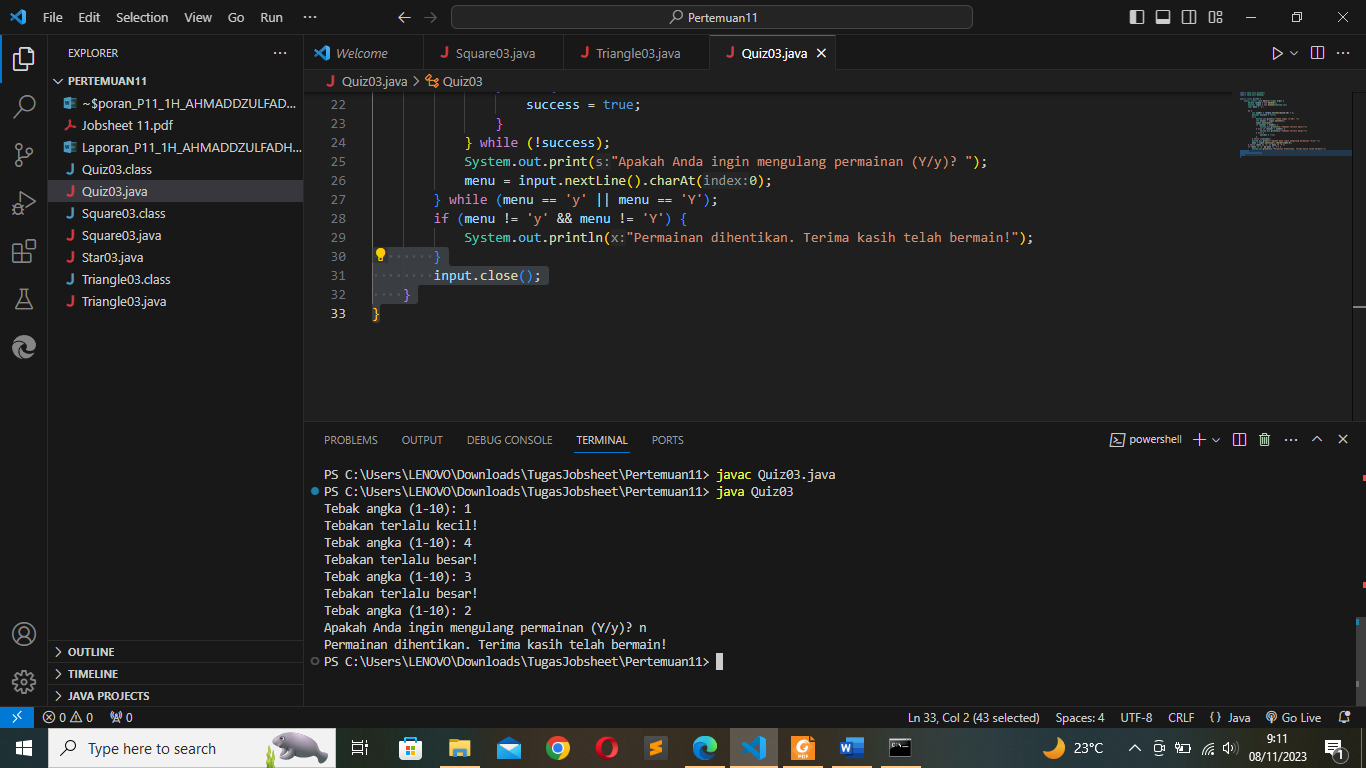
}

input.close();

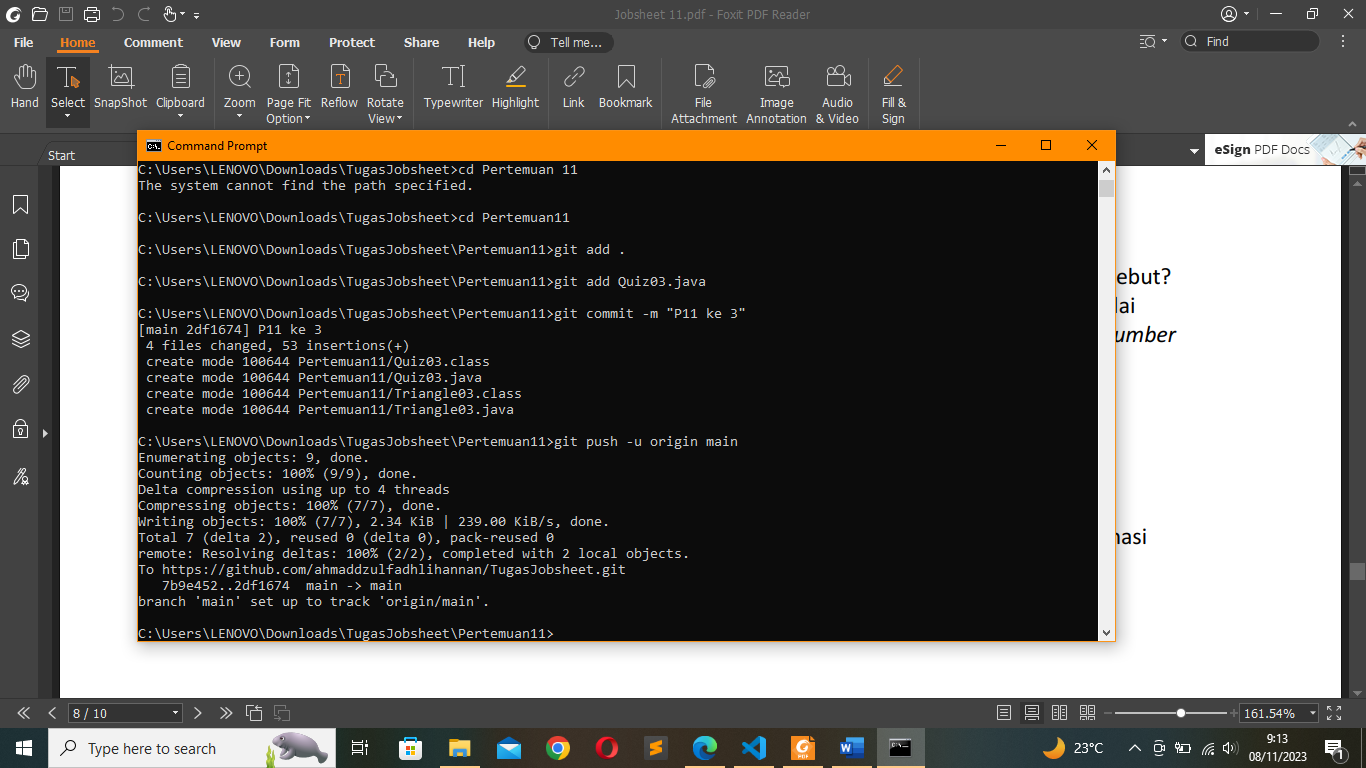
}

}

Hasil running



1. Silakan commit dan push ke repository Anda.





* 1. **Percobaan 5: Mengisi dan menampilkan array**

**Praktikum**

Codingan

import java.util.Scanner;

public class NestedLoop\_2341720106 {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

double[][] temps = new double[5][7];

for (int i = 0; i < temps.length; i++) {

System.out.println("Kota ke-" + i);

for (int j = 0; j < temps[0].length; j++) {

System.out.print("Hari ke-" + (j + 1) + ": ");

temps[i][j] = sc.nextDouble();

}

System.out.println();

}

for (int i = 0; i < temps.length; i++) {

System.out.print("Kota ke-" + (i + 1) + ": ");

for (int j = 0; j < temps[0].length; j++) {

System.out.print(temps[i][j] + " ");

}

System.out.println();

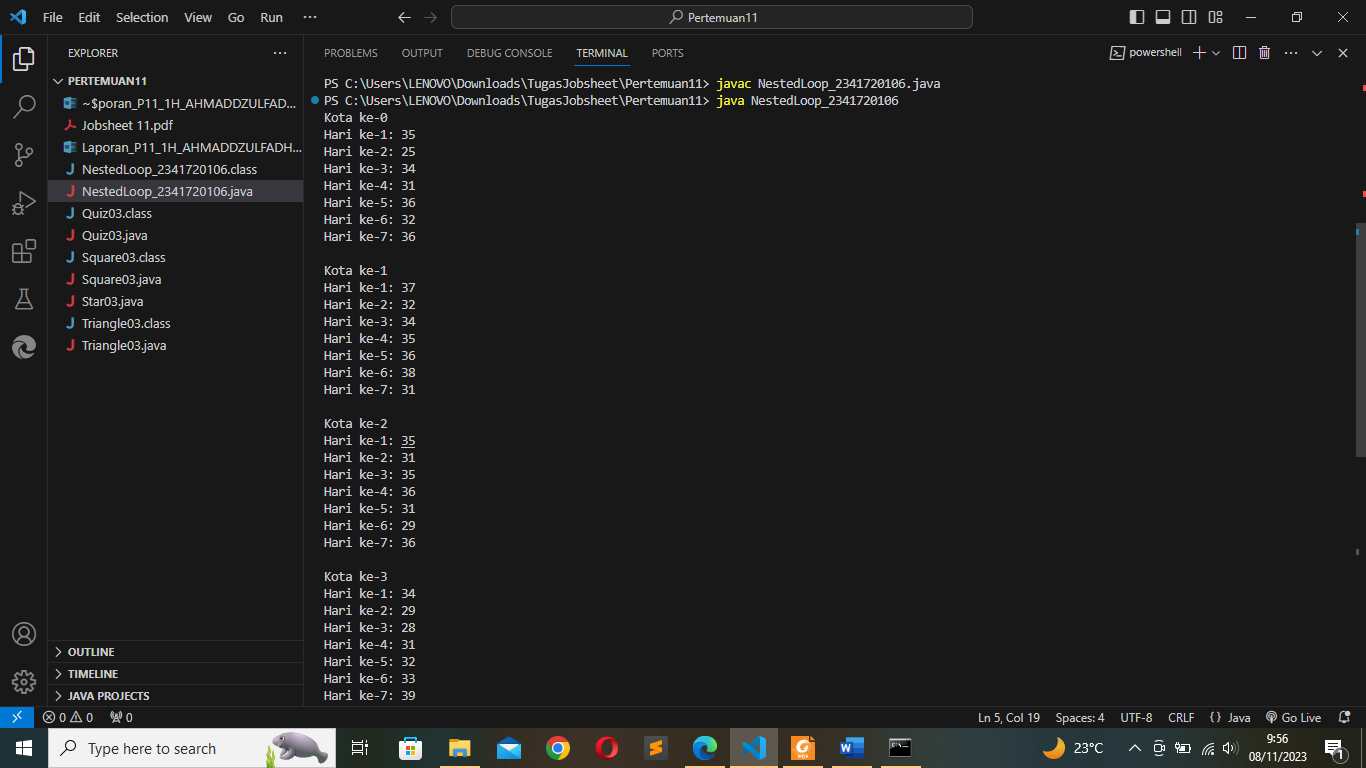
}

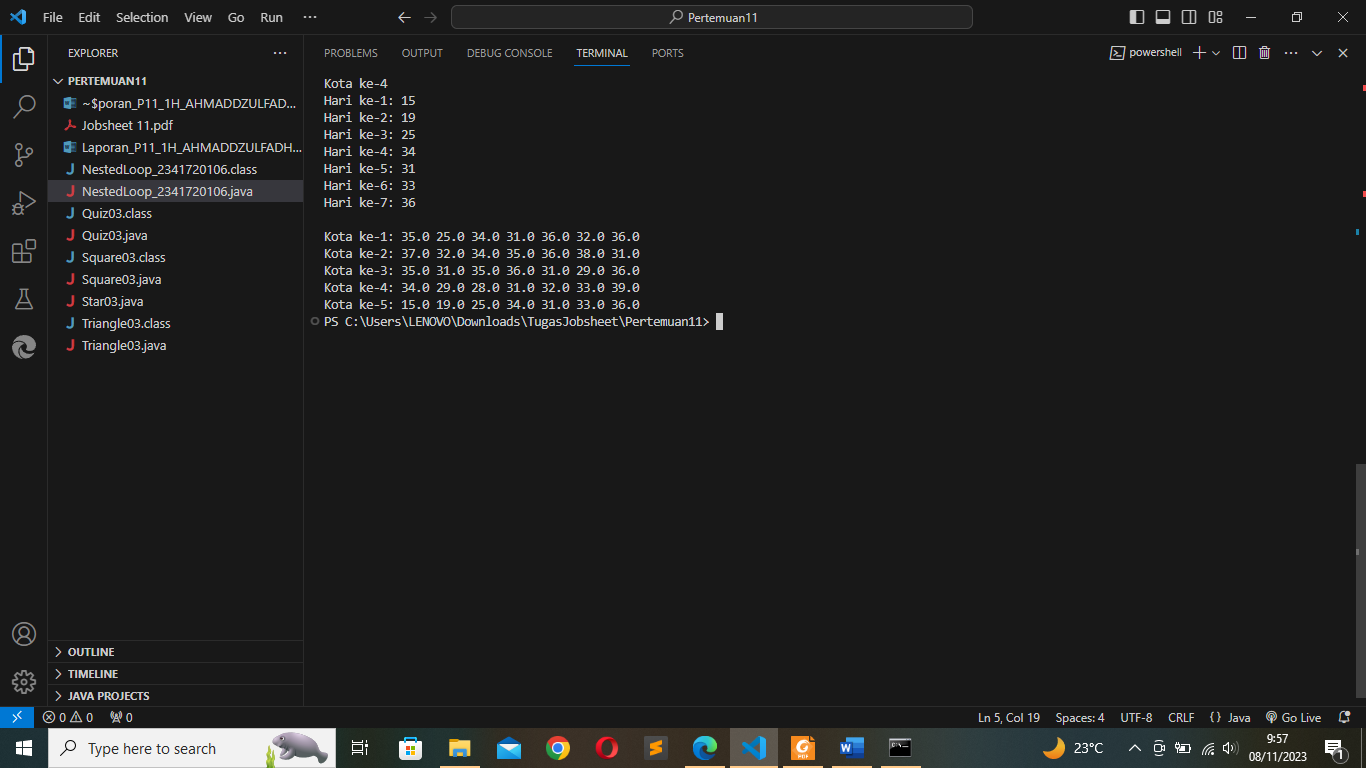
sc.close();

}

}

Hasil Running





**Pertanyaan**

1. Jelaskan alur program di atas!

* Program dimulai dengan mendeklarasikan objek Scanner dan array 2 dimensi temps untuk menyimpan data suhu.
* Program meminta pengguna memasukkan data suhu untuk setiap hari dalam seminggu untuk setiap kota.
* Setelah semua data suhu dimasukkan, program menampilkan data suhu untuk setiap kota dalam bentuk tabel 2 dimensi.

1. Silakan modifikasi program di atas pada bagian untuk menampilkan array menggunakan foreach!

for (double[] tempArray : temps) {

for (double temp : tempArray) {

System.out.print(temp + " ");

}

System.out.println();

}

1. Modifikasi program di atas sehingga bisa menampilkan nilai rata-rata masing-masing kota!

Modifikasi dengan menambahkan perulangan berikut setelah menampilkan data suhu kota.

for (int i = 0; i < temps.length; i++) {

double sum = 0;

for (int j = 0; j < temps[i].length; j++) {

sum += temps[i][j];

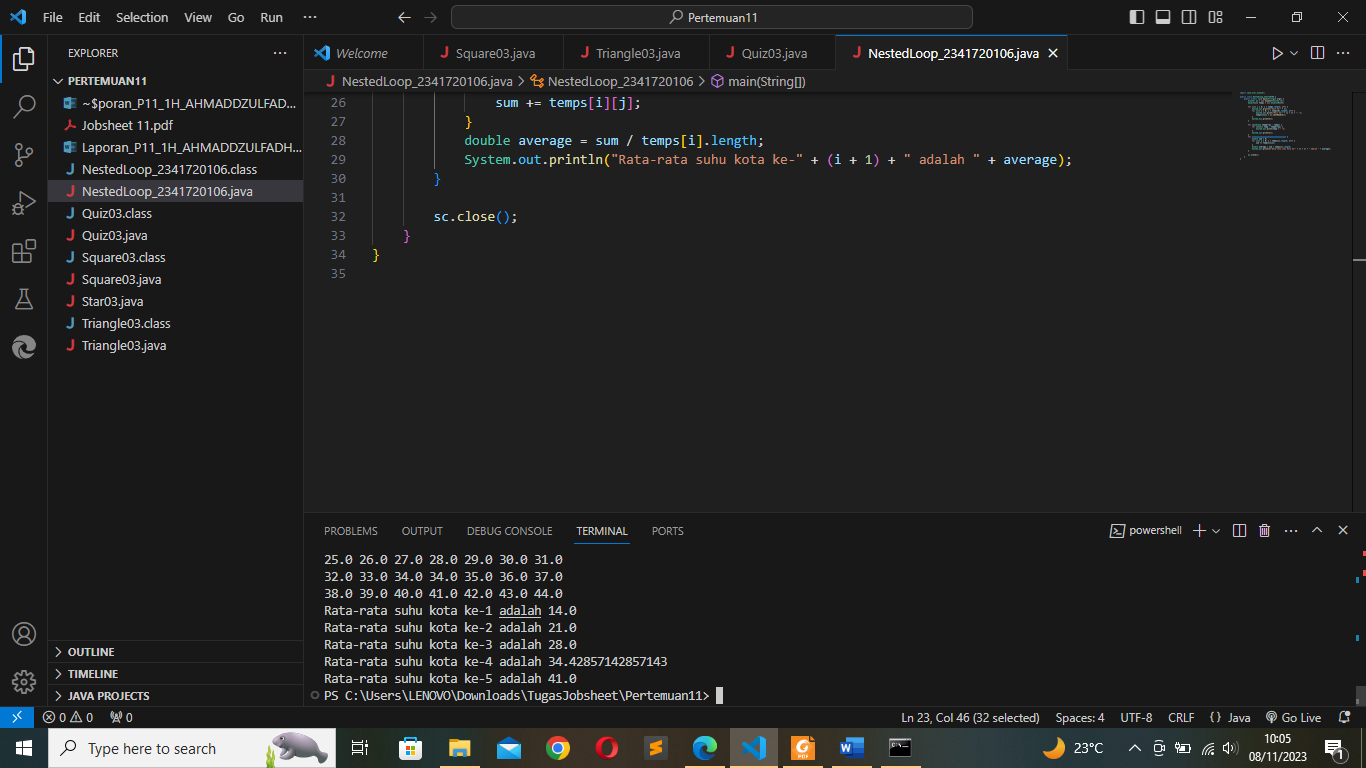
}

double average = sum / temps[i].length;

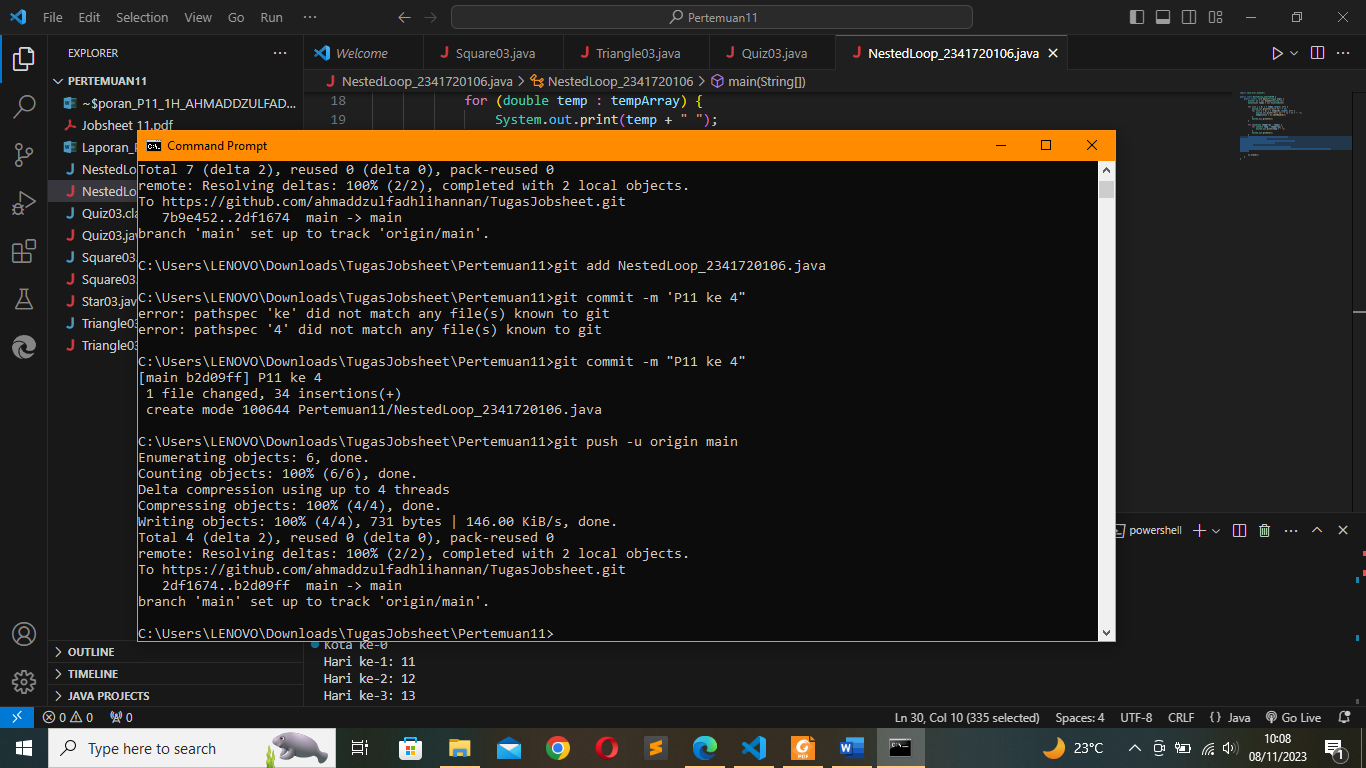
System.out.println("Rata-rata suhu kota ke-" + (i + 1) + " adalah " + average);

}

Hasil running



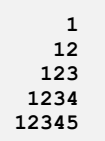
1. Silakan commit dan push ke repository Anda.





**Tugas Mandiri dan Kelompok**

1. Buatlah program untuk mencetak tampilan segitiga angka seperti di bawah ini berdasarkan input N (nilai N minimal 3). Contoh N = 5



Codingan

import java.util.Scanner;

public class TMandiri11ke1 {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Masukkan nilai N: ");

int N = sc.nextInt();

if (N < 3) {

System.out.println("N minimal harus 3.");

} else {

for (int i = 1; i <= N; i++) {

// Spasi

for (int j = 1; j <= N - i; j++) {

System.out.print(" ");

}

// Angka

for (int k = 1; k <= i; k++) {

System.out.print(k);

}

System.out.println();

}

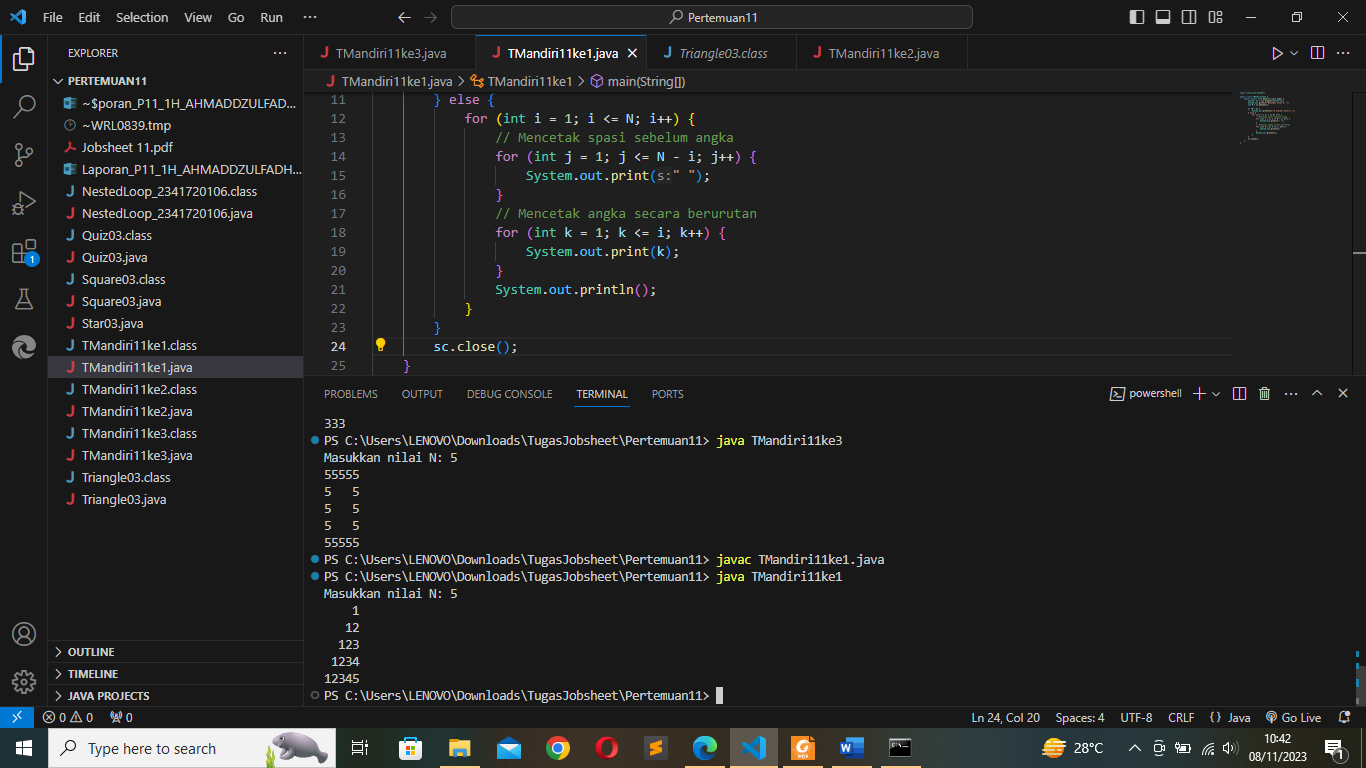
}

sc.close();

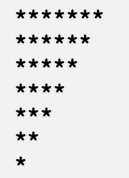
}

}

Hasil running



1. Buatlah program untuk mencetak tampilan segitiga bintang seperti di bawah ini berdasarkan input N (nilai N minimal 5). Contoh N = 7



Codingan

import java.util.Scanner;

public class TMandiri11ke2 {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Masukkan nilai N: ");

int N = sc.nextInt();

if(N < 5) {

System.out.println("N minimal harus 5.");

} else {

for(int i = N; i >= 1; i--) {

for(int j = 1; j <= i; j++) {

System.out.print("\*");

}

System.out.println();

}

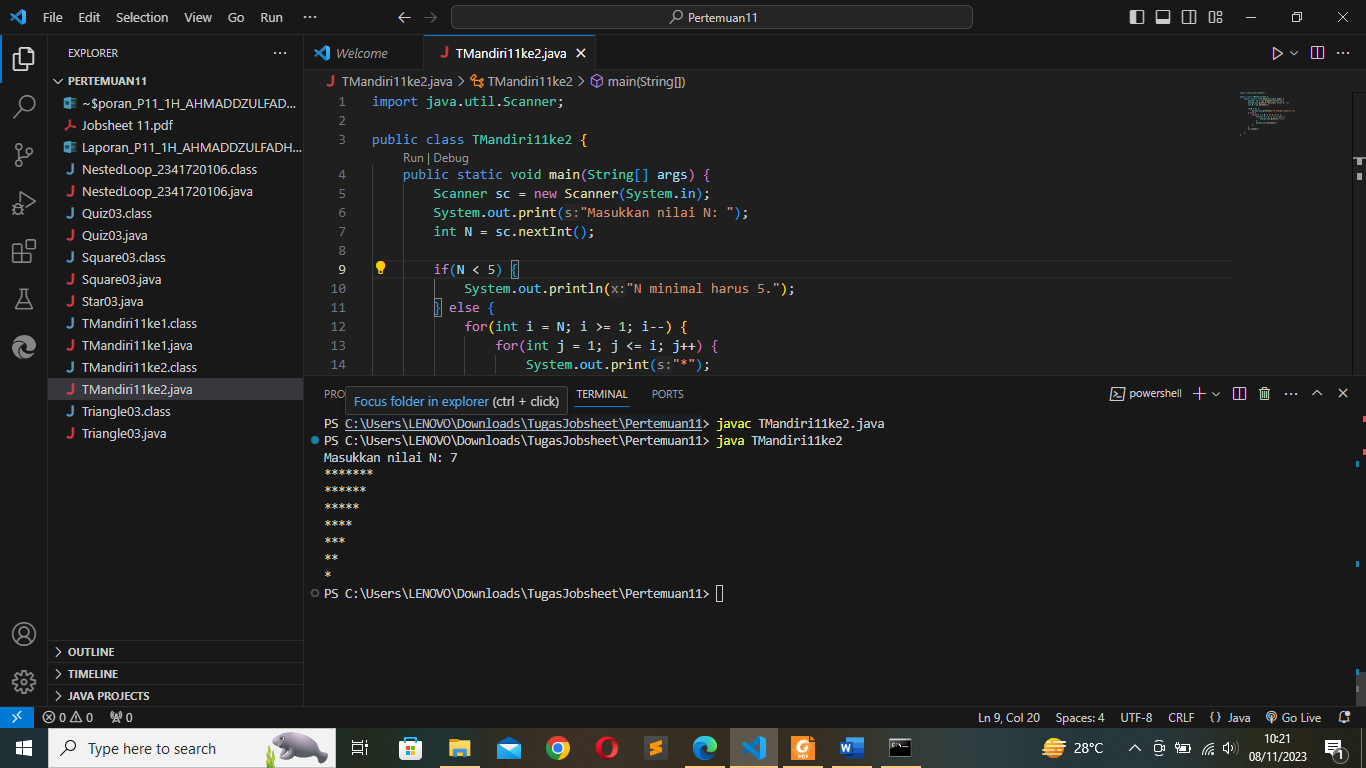
}

sc.close();

}

}

Hasil running



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



Codingan

import java.util.Scanner;

public class TMandiri11ke3 {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Masukkan nilai N: ");

int N = sc.nextInt();

if (N < 3) {

System.out.println("N minimal harus 3.");

} else {

// Baris pertama

for (int i = 0; i < N; i++) {

System.out.print(N);

}

System.out.println();

// Baris tengah

for (int i = 0; i < N - 2; i++) {

System.out.print(N);

for (int j = 0; j < N - 2; j++) {

System.out.print(" ");

}

System.out.println(N);

}

// Baris terakhir

for (int i = 0; i < N; i++) {

System.out.print(N);

}

System.out.println();

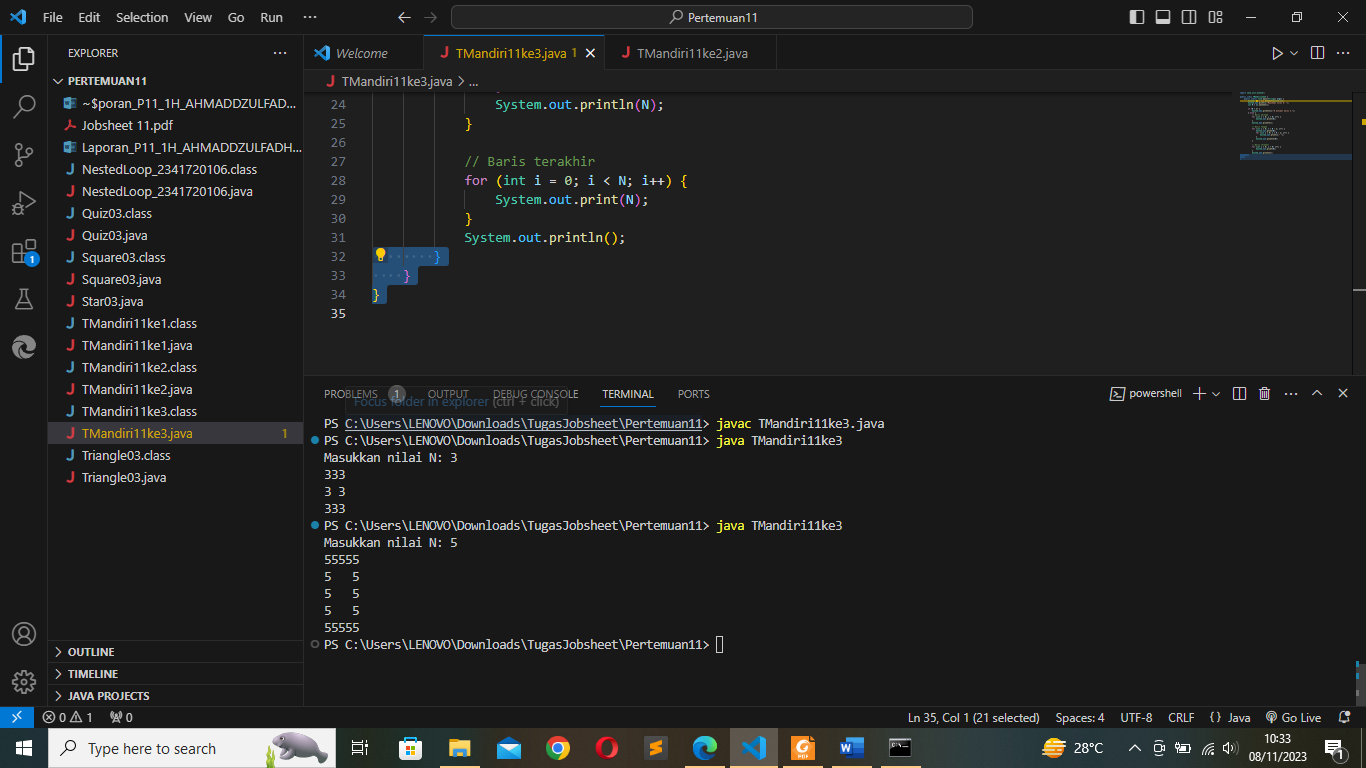
}

sc.sclose();

}

}

Hasil running



1. Implementasikan flowchart dari fitur-fitur yang telah Anda buat pada tugas teori sebelumnya tentang nested loop!
2. Jangan lupa, semoga kode program harus di-push ke repository Anda