

LAPORAN PRAKTIKUM  
ALGORITMA PEMROGRAMAN PEKAN LIMA  
*FOR STATEMENT*

disusun Oleh:

Zahira Nur Asyifa

2411532015

Dosen Pengampu: Dr. Wahyudi, S.T, M.T

Asisten Praktikum: Rahmad Dwirizki Olders



DEPARTEMEN INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS ANDALAS

2025

## KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah SWT karena atas rahmat dan karunia-Nya, laporan praktikum Algoritma dan Pemrograman dengan topik Perulangan For ini dapat diselesaikan dengan baik. Laporan ini disusun untuk memenuhi tugas praktikum pekan kelima mata kuliah Algoritma dan Pemrograman, sekaligus sebagai sarana pembelajaran dalam memahami konsep dasar perulangan menggunakan bahasa Java.

Melalui praktikum ini penulis dapat mempelajari cara kerja dan penerapan struktur perulangan for, baik dalam bentuk sederhana maupun bersarang(*Nested Loop*). Penulis menyadari bahwa laporan ini masih jauh dari sempurna, baik dari segi penjelasan maupun penyajian. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan demi perbaikan di masa mendatang. Akhir kata, penulis mengucapkan terima kasih kepada dosen pengampu, asisten praktikum, serta semua pihak yang telah memberikan bimbingan dan dukungan sehingga laporan ini dapat terselesaikan. Semoga laporan ini dapat memberikan manfaat bagi penulis sendiri maupun pembaca.

Padang, 2025

Penulis

## DAFTAR ISI

<b>KATA PENGANTAR.....</b>	<b>i</b>
<b>DAFTAR ISI .....</b>	<b>ii</b>
<b>DAFTAR GAMBAR .....</b>	<b>iii</b>
<b>BAB I.....</b>	<b>4</b>
<b>PENDAHULUAN .....</b>	<b>4</b>
<b>1.1    Latar Belakang .....</b>	<b>4</b>
<b>1.2    Tujuan .....</b>	<b>4</b>
<b>1.3    Manfaat.....</b>	<b>Error! Bookmark not defined.</b>
<b>BAB II .....</b>	<b>4</b>
<b>PEMBAHASAN.....</b>	<b>5</b>
<b>2.1    Class PerulanganFor1 dan PerulanganFor2 .....</b>	<b>5</b>
<b>2.2    Class PerulanganFor3 dan PerulanganFor4 .....</b>	<b>8</b>
<b>2.3    Class NestedFor0 .....</b>	<b>12</b>
<b>2.4    Class NestedFor1 .....</b>	<b>13</b>
<b>2.5    Class NestedFor2 .....</b>	<b>14</b>
<b>BAB III.....</b>	<b>17</b>
<b>PENUTUP .....</b>	<b>17</b>
<b>3.1    Kesimpulan .....</b>	<b>17</b>
<b>3.2    Saran .....</b>	<b>17</b>
<b>DAFTAR PUSTAKA .....</b>	<b>18</b>

## DAFTAR GAMBAR

Gambar 2.1 Membuat package baru .....	5
Gambar 2. 2 Membuat nama package.....	5
Gambar 2. 3 Membuat class baru .....	6
Gambar 2. 4 Syntax Class PerulanganFor1 .....	6
Gambar 2. 5 Output PerulanganFor1 .....	7
Gambar 2. 6 Syntax PerulanganFor2 .....	7
Gambar 2. 7 Ouput PerulanganFor2 .....	8
Gambar 2. 8 Membuat class PerulanganFor3 .....	8
Gambar 2. 9 Syntax PerulanganFor3 .....	9
Gambar 2. 10 Output PerulanganFor3 .....	9
Gambar 2. 11 Syntax PerulanganFor4 .....	10
Gambar 2. 12 Output PerulanganFor4 .....	11
Gambar 2. 13 Membuat class NestedFor0 .....	12
Gambar 2. 14 Syntax NestedFor0.....	12
Gambar 2. 15 Output NestedFor0.....	13
Gambar 2. 16 Membuat class NestedFor1 .....	13
Gambar 2. 17 syntax NestedFor1 .....	14
Gambar 2. 18 Output NestedFor1.....	14
Gambar 2. 19 class NestedFor2.....	15
Gambar 2. 20 Syntax NestedFor2.....	15
Gambar 2. 21 Output NestedFor2.....	16

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Perulangan pada java terbagi menjadi tiga jenis yaitu : For, While dan Do While. Pada praktikum kali ini menggunakan perulangan for yaitu perulangan yang digunakan ketika jumlah yang akan kita gunakan diketahui. Pada perulangan for menggunakan 3 statement, statement1 dieksekusi sekali sebelum blok kode dieksekusi, Statement2 mendefinisikan kondisi untuk mengeksekusi blok kode, Statement3 dieksekusi (setiap kali) setelah blok kode dieksekusi.

Ada contoh lain dari perulangan for yang dimana jika suatu perulangan berada di dalam badan perulangan lain, perulangan tersebut disebut perulangan bersarang (nested for).

### **1.2 Tujuan**

1. Memahami konsep dasar perulangan for
2. Melatih kemampuan menerapkan nested for

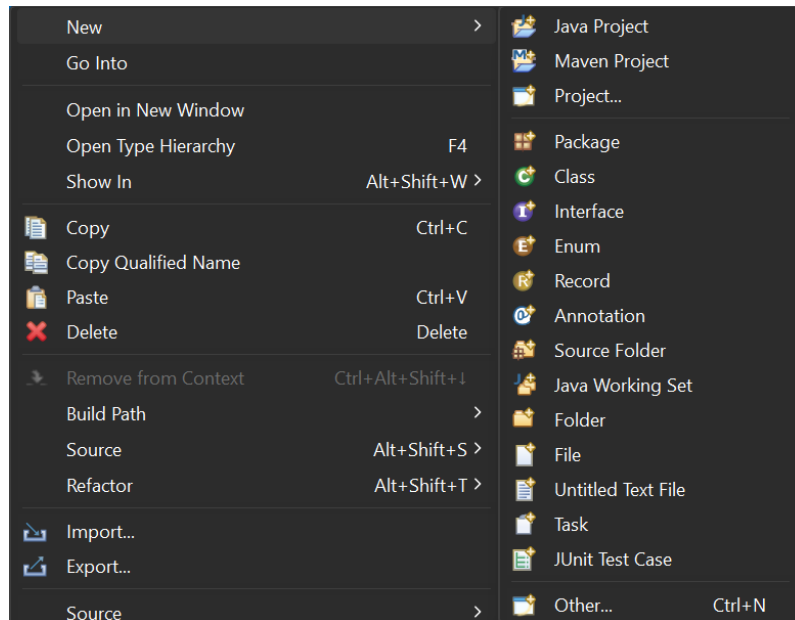
### **1.3 Manfaat**

1. Mahasiswa dapat menulis dan memahami kode perulangan
2. Mahasiswa dapat menganalisis dan memecahkan masalah yang melibatkan perulangan

## BAB II PEMBAHASAN

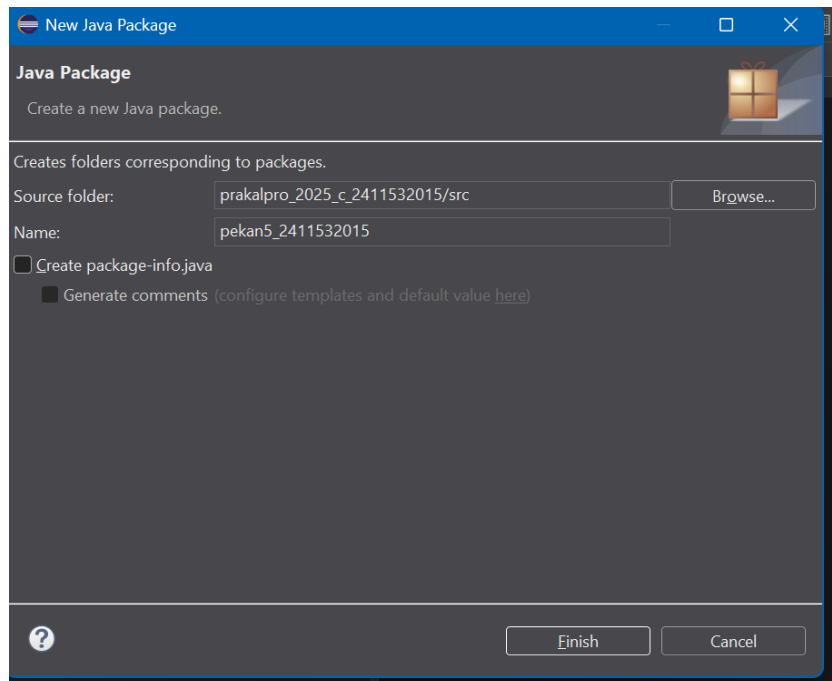
### 2.1 *Class* PerulanganFor1 dan PerulanganFor2

1. Buat *package* baru dengan klik kanan pada src, pilih new lalu *package*



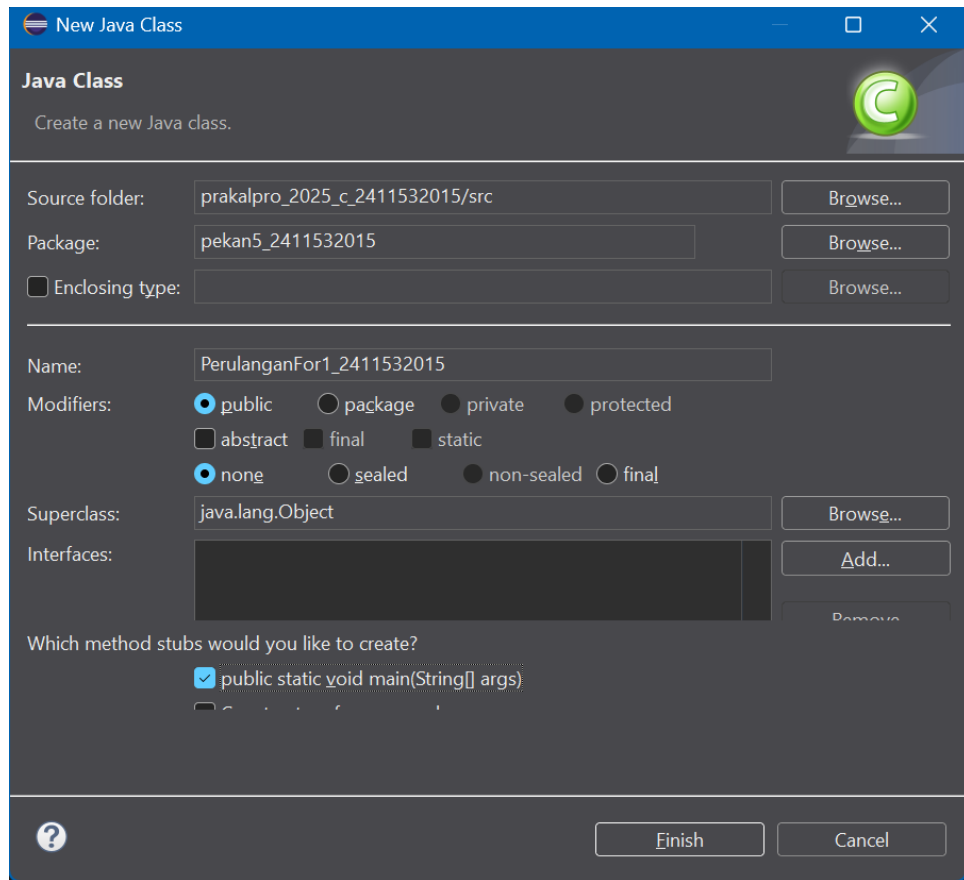
Gambar 2.1 Membuat package baru

2. Buat *package* dengan nama pekan 5



Gambar 2. 2 Membuat nama package

3. Buat *class* baru dengan nama “PerulanganFor1”



Gambar 2. 3 Membuat *class* baru

4. Buat *syntax* seperti ini

```
package pekan5_2411532015;

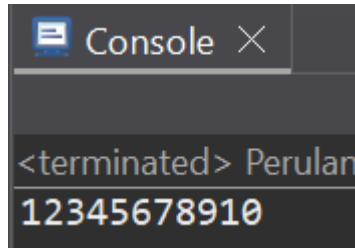
public class PerulanganFor1_2411532015 {

    public static void main(String[] args) {
        for (int i = 1; i<=10; i++) {
            System.out.print(i);
        }
    }
}
```

Gambar 2. 4 *Syntax Class PerulanganFor1*

- `int i=1` nilai awal variabel `i` adalah 1
- `i<=10`, selama `i` kurang atau sama dengan 10 perulangan terus berjalan
- `i++`, setiap satu perulangan selesai, nilai `i` akan bertambah

5. Ketika dijalankan maka outputnya akan seperti ini



Gambar 2. 5 Output PerulanganFor1

6. Buat class baru dengan nama “PerulanganFor2” dengan syntax yang kurang lebih sama seperti sebelumnya. Perbedaannya terdapat pada spasi setelah `i`

```

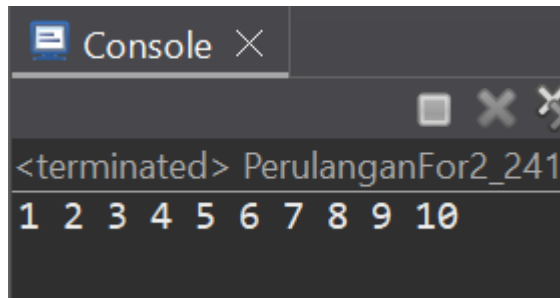
1 package pekan5_2411532015;
2
3 public class PerulanganFor2_2411532015 {
4
5     public static void main(String[] args) {
6         for (int i = 1; i<=10; i++) {
7             System.out.print(i+" ");
8         }
9
10    }
11
12 }
13

```

Gambar 2. 6 Syntax PerulanganFor2



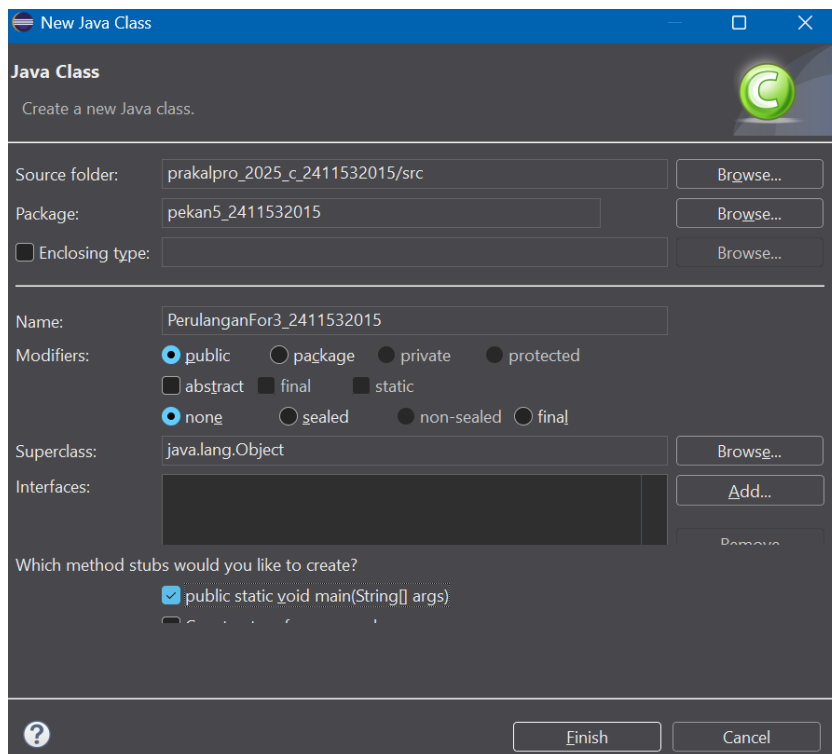
7. Berikut outputnya



Gambar 2. 7 Ouput PerulanganFor2

## 2.2 Class *PerulanganFor3* dan *PerulanganFor4*

1. Selanjutnya buat lagi *class* baru dengan nama “*PerulanganFor3*”;



Gambar 2. 8 Membuat *class* *PerulanganFor3*

2. Masukkan *syntax* seperti ini

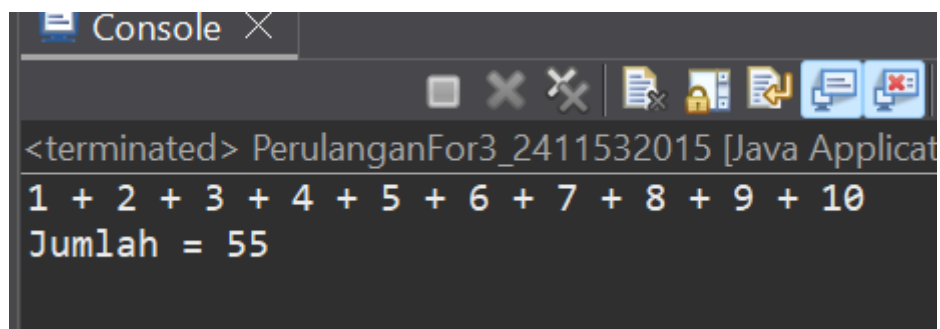
```
package pekan5_2411532015;

public class PerulanganFor3_2411532015 {

    public static void main(String[] args) {
        int jumlah = 0;
        for (int i=1; i<=10; i++) {
            System.out.print(i);
            if (i<10) {
                System.out.print(" + ");
            }
            jumlah+=i;
        }
        System.out.println();
        System.out.println("Jumlah = "+jumlah);
    }
}
```

Gambar 2. 9 Syntax PerulanganFor3

- Perulangan berjalan dari I = 1 sampai 10
  - If(i<10) agar tanda + tidak muncul di akhir
  - jumlah+=I agar setiap angka dijumlahkan ke variabel jumlah
3. Output nantinya akan seperti ini. Perbedaan dengan *class* sebelumnya adalah jika input kurang dari 2.75 maka akan ditampilkan “Anda Tidak Lulus”



```
<terminated> PerulanganFor3_2411532015 [Java Applicat
1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10
Jumlah = 55
```

Gambar 2. 10 Output PerulanganFor3

4. Buat class baru dengan nama “PerulanganFor4” dengan berikut. Pada class ini menggunakan scanner yang membuatnya berbeda dengan class sebelumnya yang menetapkan batas langsung.

```
package pekan5_2411532015;
import java.util.Scanner;

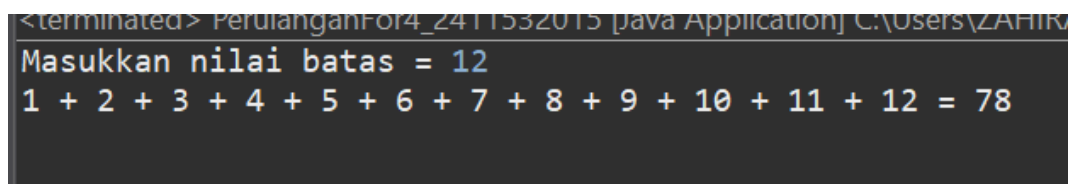
public class PerulanganFor4_2411532015 {

    public static void main(String[] args) {
        int jumlah = 0;
        int batas;
        Scanner input = new Scanner(System.in);
        System.out.print("Masukkan nilai batas = ");
        batas= input .nextInt();
        input.close();
        for (int i=1;i<=batas;i++) {
            System.out.print(i);
            jumlah = jumlah+i;
            if(i<batas) {
                System.out.print(" + ");
            }else {
                System.out.print(" = ");
            }
        }
        System.out.println(jumlah);
    }
}
```

Gambar 2. 11 Syntax PerulanganFor4

- `int batas` akan menyimpan angka batas yang dimasukkan pengguna
- `for (int i = 1; i <= batas; i++)` artinya program menghitung dari 1 sampai batas yang dimasukkan pengguna
- `jumlah = jumlah + i` agar menambahkan angka `i` ke total jumlah
- `if(i<batas)` agar membari batas + diantara angka dan = diakhir

5. Berikut outputnya, misal input 12

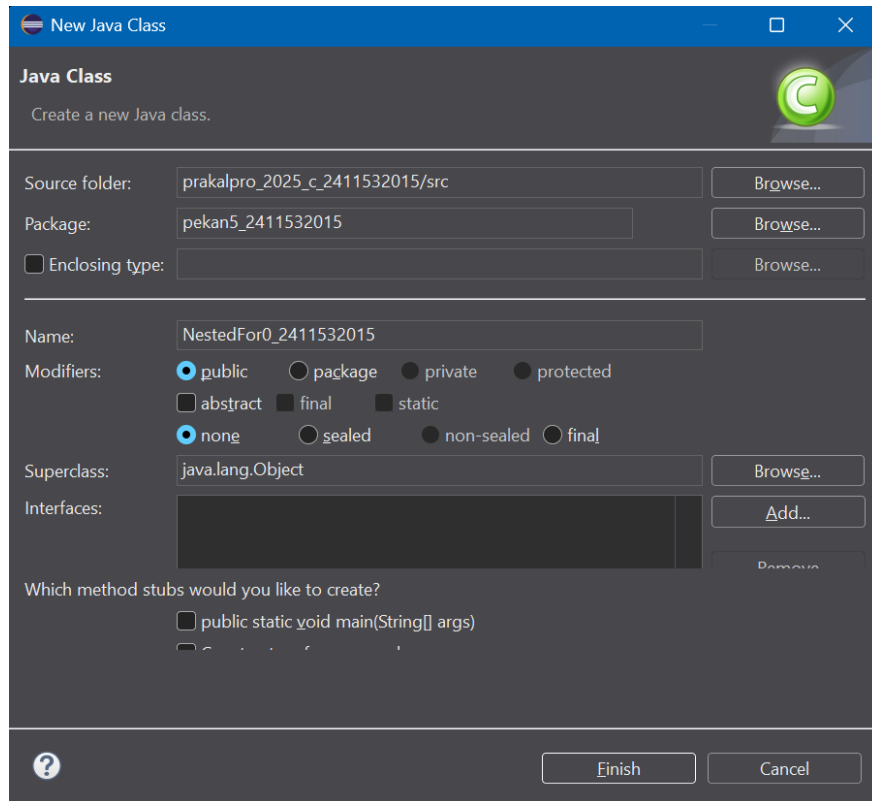


```
<terminated> PerulanganFor4_2411532015 [Java Application] C:\Users\ZAHIR
Masukkan nilai batas = 12
1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10 + 11 + 12 = 78
```

*Gambar 2. 12 Output PerulanganFor4*

## 2.3 Class NestedFor0

1. Selanjutnya buat *class* dengan nama “NestedFor0



Gambar 2. 13 Membuat *class* NestedFor0

2. Masukkan *syntax* berikut

```
package pekan5_2411532015;

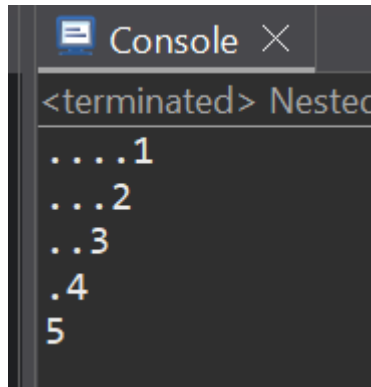
public class NestedFor0_2411532015 {
    public static void main(String[] args) {
        for (int line = 1; line<=5; line++) {
            for (int j = 1; j<= (-1 * line + 5); j++) {
                System.out.print(".");
            }
            System.out.print(line);
            System.out.println();
        }
    }
}
```

Gambar 2. 14 *Syntax* NestedFor0

- for pertama mengatur baris yang akan dicetak, program akan menjalankan isi perulangan 5 kali

- for kedua menentukan berapa banyak titik yang dicetak setiap baris sesuai rumus
- setelah titik selesai dicetak, program mencetak angka barisnya (line)

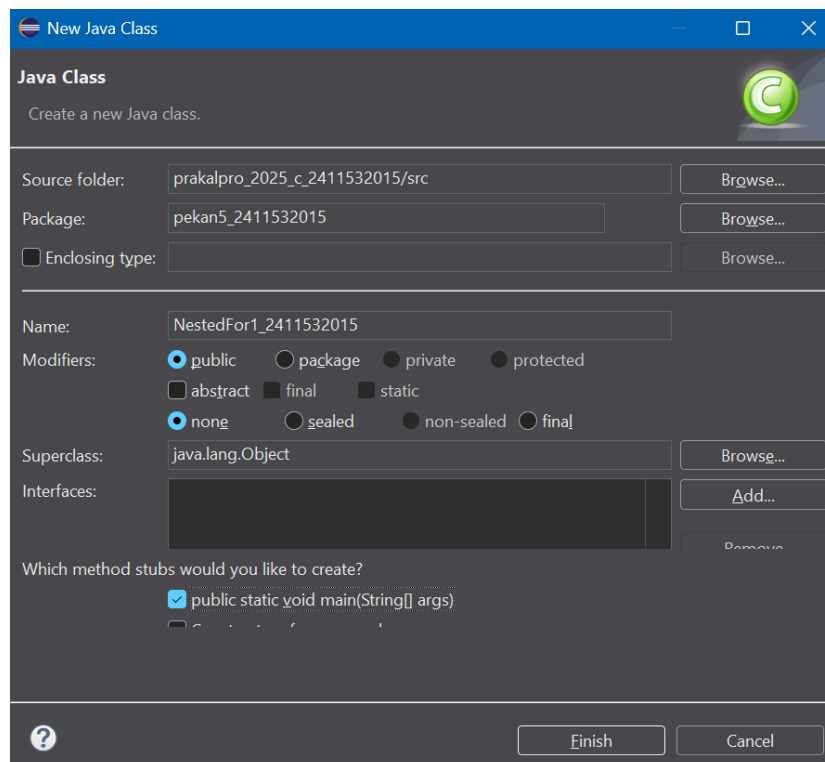
3. Output nantinya akan seperti ini



Gambar 2. 15 Output NestedFor0

## 2.4 Class NestedFor1

1. Buat *class* baru dengan nama “NestedFor1”



Gambar 2. 16 Membuat *class* NestedFor1

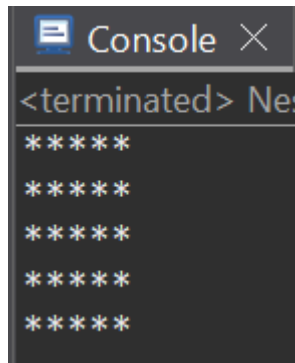
2. Buat *syntax* seperti ini

```
1 package pekan5_2411532015;
2
3 public class NestedFor1_2411532015 {
4
5     public static void main(String[] args) {
6         for (int i = 1; i<=5; i++) {
7             for (int j = 1; j<=5; j++) {
8                 System.out.print("*");
9             }
10            System.out.println();
11        }
12    }
13 }
14
15 }
16
```

Gambar 2. 17 *syntax* NestedFor1

- for pertama mengatur jumlah baris, nantinya akan ada 5 baris
- for kedua untuk mengatur jumlah bintang di setiap baris, setiap perulangan pertama berjalan sekali, perulangan ini akan mencetak 5 bintang

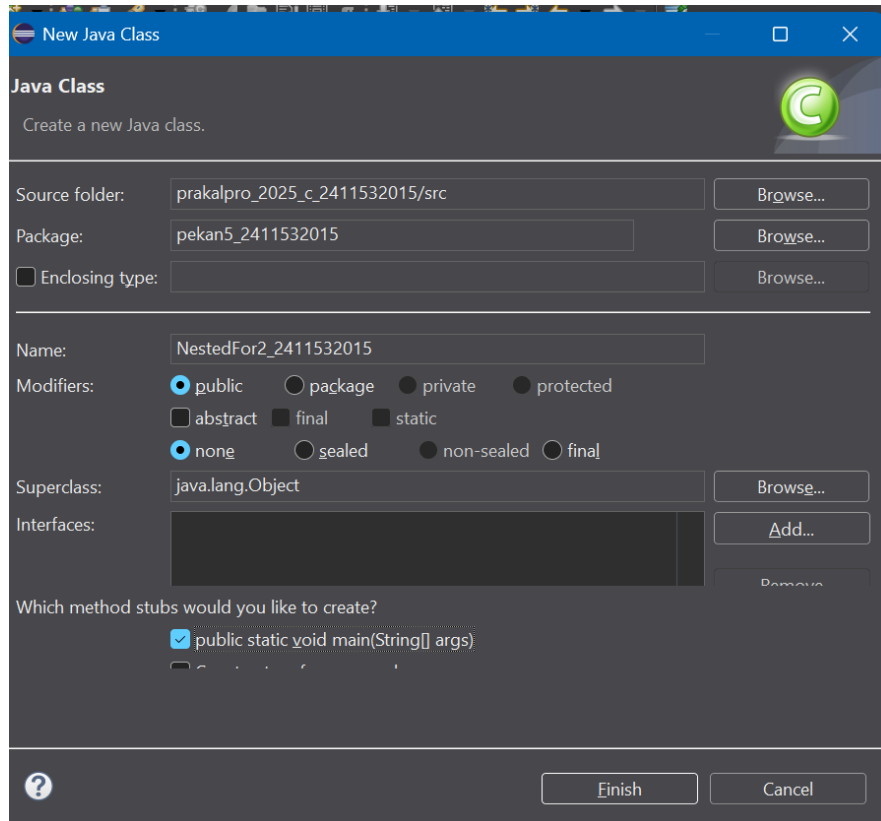
3. Output nya seperti ini



Gambar 2. 18 Output *NestedFor1*

## 2.5 Class NestedFor2

1. Buat *class* baru dengan nama “NestedFor2”



Gambar 2. 19 class NestedFor2

## 2. Buat syntax seperti ini

```

1 package pekan5_2411532015;
2
3 public class NestedFor2_2411532015 {
4
5     public static void main(String[] args) {
6         for (int i = 0; i<=5; i++) {
7             for (int j = 0; j<=5; j++) {
8                 System.out.print(i+j+" ");
9             }
10            System.out.println();
11        }
12    }
13 }
14
15 }
16

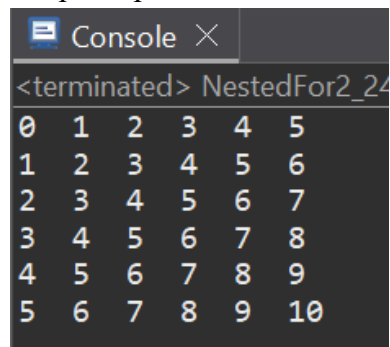
```

Gambar 2. 20 Syntax NestedFor2



- for pertama artinya variabel i mulai dari 0 sampai 5, jadi akan ada 6 baris output
- for kedua artinya untuk setiap nilai i, variabel j juga akan berulang dari 0 sampai 5
- `System.out.print(i+j+" ")` untuk setiap perulangan, program mencetak hasil penjumlahan i+j

### 3. Output seperti ini



```

Console X
<terminated> NestedFor2_24
0 1 2 3 4 5
1 2 3 4 5 6
2 3 4 5 6 7
3 4 5 6 7 8
4 5 6 7 8 9
5 6 7 8 9 10

```

*Gambar 2. 21 Output NestedFor2*

## **BAB III**

### **PENUTUP**

#### **3.1 Kesimpulan**

Berdasarkan praktikum yang telah dilakukan, dapat disimpulkan bahwa penggunaan struktur perulangan for merupakan salah satu konsep dasar yang sangat penting dalam pemrograman, terutama untuk mengontrol eksekusi kode yang berulang dengan jumlah iterasi tertentu. Melalui berbagai percobaan kode seperti *PerulanganFor1* hingga *PerulanganFor4*, mahasiswa dapat memahami cara mencetak deret angka, menghitung jumlah bilangan, serta menampilkan hasil penjumlahan secara bertahap. Sementara itu, percobaan *NestedFor0* hingga *NestedFor2* memperlihatkan bagaimana perulangan bersarang dapat digunakan untuk membuat pola visual dan tabel sederhana dengan logika dua dimensi.

#### **3.2 Saran**

Dalam penerapan perulangan for, disarankan agar setiap mahasiswa terlebih dahulu merancang alur logika program secara sistematis sebelum menuliskannya dalam kode. Perencanaan ini penting untuk menghindari kesalahan logika dan hasil keluaran yang tidak sesuai.

## DAFTAR PUSTAKA

- [1] malasngoding, “Tutorial Java Part 9 : Perulangan Pada Java”. [Daring]. Tersedia pada: <https://www.malasngoding.com/perulangan-pada-java/>
- [2] nextgen, “Perulangan For Java”. [Daring]. Tersedia pada: <https://nextgen.co.id/perulangan-for-java/>
- [3] programiz, “Nested Loop in Java”. [Daring]. Tersedia pada: <https://www.programiz.com/java-programming/nested-loop>