

LAPORAN PRATIKUM
ALGORITMA DAN PEMROGRAMAN
PENERAPAN PERULANGAN DALAM JAVA



disusun Oleh:
Annisa Layli Ramadhani
2511532024

Dosen Pengampu: Wahyudi. Dr.. S.T.M.T
Asisten Pratikum: Rahmad Dwirizki Olders

DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS
TAHUN 2025

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah SWT karena dengan rahmat-Nya laporan praktikum Algoritma dan Pemrograman tanggal 27 Oktober 2025 dapat diselesaikan. Laporan ini membahas materi tentang Penerapan Perulangan dalam Java sebagai dasar penting dalam memahami pemrograman. Penulis mengucapkan terima kasih kepada dosen, asisten, dan rekan-rekan yang telah membantu selama praktikum. Penulis menyadari laporan ini masih jauh dari sempurna, sehingga kritik dan saran sangat diharapkan. Semoga laporan ini bermanfaat bagi pembaca.

Padang, 27 Oktober 2025

Penulis

DAFTAR ISI

KATA PENGANTAR	ii
DAFTAR ISI.....	iii
BAB I.....	4
1.1 Latar Belakang	4
1.2 Tujuan	4
1.3 Manfaat	4
BAB II	6
2.1 Dasar Teori	6
2.1.1 Perulangan.....	6
2.1.2 Perulangan For.....	6
2.1.3 Perulangan Nested For	7
2.2 Langkah Kerja.....	8
2.2.1 Pembuatan <i>Package</i> dan <i>Class</i> Pekan 5	8
2.2.2 Program Pertama: Perulangan For 1	9
2.2.3 Program Kedua: Perulangan For 2.....	10
2.2.4 Program Ketiga: Perulangan For 3.....	10
2.2.5 Program Keempat: Perulangan For 4.....	11
2.2.6 Program Kelima: Nested For 0.....	12
2.2.7 Program Keenam: Nested For 1	12
2.2.8 Program Ketujuh: Nested For 2	13
2.2.9 <i>Commit</i> dan <i>Push</i> ke GitHub.....	13
BAB III.....	16
3.1 Kesimpulan	16
3.2 Saran.....	16
DAFTAR PUSTAKA	17

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam pemrograman, perulangan (looping) merupakan salah satu konsep dasar yang berfungsi untuk menjalankan satu tau beberapa perintah secara berulang hingga kondisi tertentu terpenuhi. Bahasa pemrograman Java menyediakan beberapa jenis perulangan, salah satunya adalah perulangan for. Struktur perulangan ini sering digunakan karena sederhana dan mudah dipahami, terutama ketika jumlah iterasi sudah diketahui. Selain itu, terdapat juga bentuk nested for atau perulangan bersarang, yaitu penggunaan satu perulangan di dalam perulangan lainnya. Konsep ini sangat berguna untuk memproses data yang berbentuk dua dimensi, seperti matriks, tabel, atau pola bintang.

Pemahaman terhadap perulangan for dan nested for penting bagi programmer pemula agar mampu menulis kode yang efisien, menghemat waktu eksekusi program, serta menghindari penulisan instruksi yang berulang secara manual. Melalui pembelajaran dan perulangan ini, mahasiswa dapat memahami logika kontrol alur program secara lebih mendalam dan mengaplikasikannya pada berbagai kasus pemrograman yang kompleks.

1.2 Tujuan

Tujuan dari pelaksanaan praktikum antara lain sebagai berikut:

- 1.2.1 Memahami konsep dasar perulangan (looping) dalam bahasa pemrograman Java.
- 1.2.2 Mampu menggunakan struktur perulangan for secara tepat sesuai kebutuhan program.
- 1.2.3 Mengetahui cara kerja nested for dalam pemrosesan data dua dimensi.

1.3 Manfaat

Manfaat dari pelaksanaan praktikum antara lain sebagai berikut:

- 1.3.1 Memudahkan penyusunan program yang membutuhkan proses berulang.
- 1.3.2 Menghemat waktu penulisan kode karena tidak perlu menulis perintah berulang secara manual.
- 1.3.3 Membantu dalam pembuatan pola, perhitungan matematis, dan pengolahan data.

BAB II

PEMBAHASAN

2.1 Dasar Teori

2.1.1 Perulangan

Perulangan adalah salah satu konsep dasar dalam pemrograman yang berfungsi untuk mengeksekusi satu atau beberapa perintah secara berulang-ulang selama kondisi tertentu masih terpenuhi. Dengan adanya perulangan, kita tidak perlu menulis baris kode yang sama berkali-kali, karena komputer akan secara otomatis menjalankan instruksi tersebut berulang sesuai dengan syarat atau batas yang telah ditentukan. Konsep ini sangat penting dalam pengembangan program yang efisien, terstruktur, dan mudah dikelola.

Dalam bahasa pemrograman Java, perulangan termasuk dalam struktur kendali alur (control structure) yang digunakan untuk mengatur urutan eksekusi program. Tujuan utama penggunaan perulangan adalah untuk menghemat waktu penulisan kode, mempermudah pemrosesan data dalam jumlah banyak, serta meningkatkan efisiensi program. Misalnya, jika sebuah program harus menampilkan angka dari 1 sampai 100, kita tidak perlu menulis 100 baris kode, cukup menggunakan satu perulangan untuk mengatur pengulangannya.

2.1.2 Perulangan For

Perulangan for merupakan salah satu jenis struktur perulangan dalam bahasa pemrograman Java yang digunakan ketika jumlah pengulangan sudah diketahui secara pasti. Struktur ini sangat umum digunakan karena bentuknya sederhana dan mudah dipahami. Perulangan for menjalankan blok kode secara berulang berdasarkan kondisi tertentu, biasanya

menggunakan variabel penghitung (counter) yang bertambah atau berkurang setiap kali perulangan dijalankan.

Struktur umum perulangan for dalam Java adalah sebagai berikut:

```
For          (inisialisasi;          kondisi;
increment/decrement){
    // blok kode yang akan diulang
}
```

Penjelasan setiap bagian:

1. Inisialisasi – digunakan untuk mendefinisikan variabel awal yang akan digunakan dalam perulangan. Biasanya berupa variabel penghitung, misalnya `int i = 0;`
2. Kondisi – berfungsi sebagai syarat untuk menentukan apakah perulangan akan terus dijalankan atau dihentikan. Selama kondisi bernilai benar (`true`), perulangan akan terus berjalan.
3. Increment/Decrement – digunakan untuk menambah atau mengurangi nilai variabel penghitung setiap kali satu siklus perulangan selesai dijalankan.

2.1.3 Perulangan Nested For

Perulangan nested for adalah bentuk perulangan di mana satu perulangan for berada didalam perulangan for lainnya. Struktur ini digunakan ketika suatu proses membutuhkan dua tingkat pengulangan atau lebih, seperti dalam pembuatan pola, pengolahan data dua dimensi (misalnya matriks atau tabel), dan perulangan yang memiliki hubungan antar variabel.

Struktur dasar nested for dalam Java ditulis sebagai berikut:

```
For (inisialisasi1; kondisi1; increment1){
    For (inisialisasi2; kondisi2; increment2){
        //blok kode yang dijalankan berulang di
        dalam perulangan luar
    }
}
```

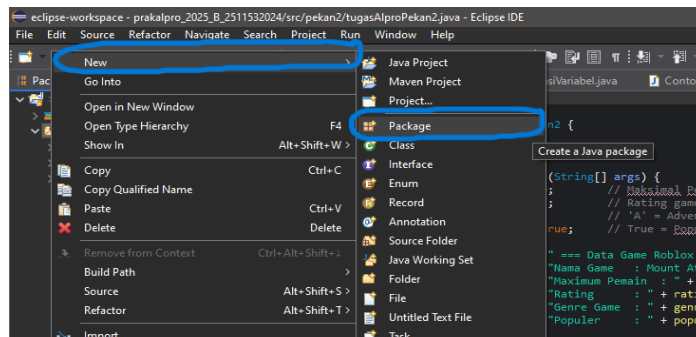
Penjelasan:

1. Perulangan luar (outer loop) – mengatur berapa kali seluruh blok perulangan di dalamnya akan dijalankan
2. Perulangan dalam (inner loop) – akan dieksekusi sepenuhnya setiap kali perulangan luar berjalan satu kali.

2.2 Langkah Kerja

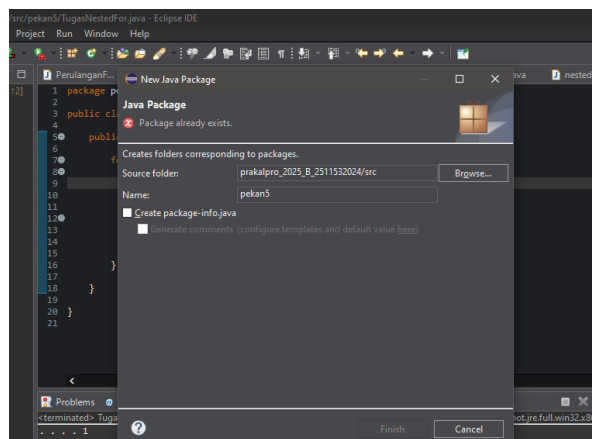
2.2.1 Pembuatan *Package* dan *Class* Pekan 5

1. Buka aplikasi eclipse, lalu buat package baru pada bagian src klik kanan, lalu pilih “New” dan klik tulisan Package.



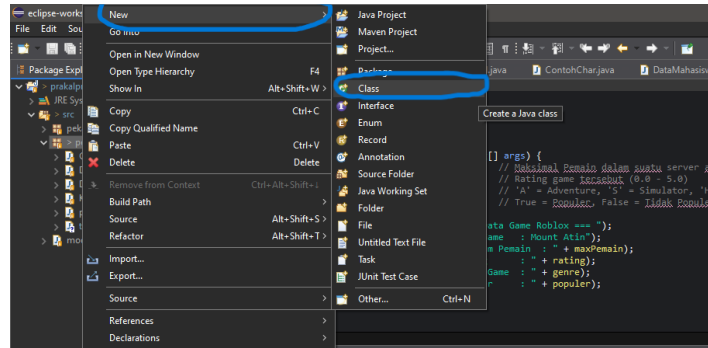
2.2.1 pembuatan package

2. Kemudian buat nama packagenya tanpa spasi, huruf kapital, karakter khusus. Beri nama “pekan5”. Lalu klik finish.



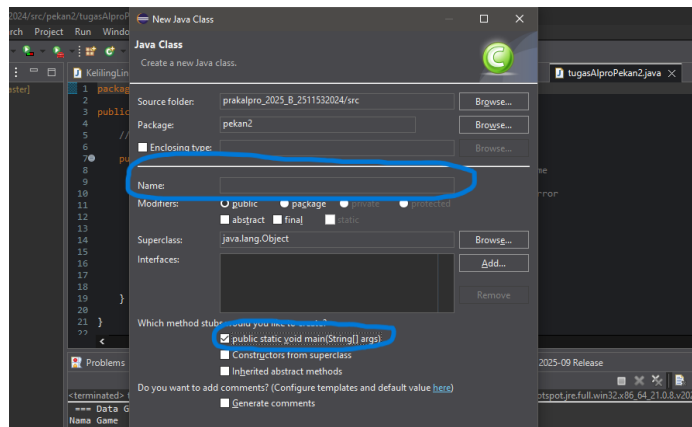
2.2.2 pembuatan package

3. Ketika package sudah jadi. Klik kanan pada bagian “pekan5”. Lalu klik “new” dan pilih bagian “Class” untuk memulai membuat program.



2.2.3 pembuatan class

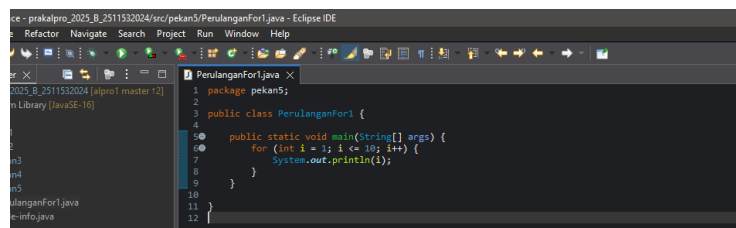
4. Buat nama class program yang akan dibuat pada bagian nama tanpa spasi dan menggunakan huruf kapital pada awal kata, lalu klik bagian “public static void main(string[]args)”. Kemudian klik “finish”.



2.2.4 pembuatan class

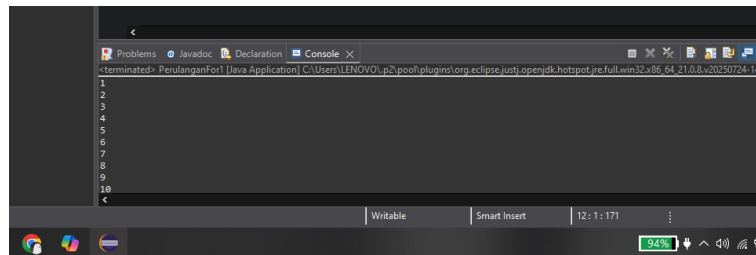
2.2.2 Program Pertama: Perulangan For 1

1. Masukkan sintaks berikut.



2.2.5 sintaks perulangan for 1

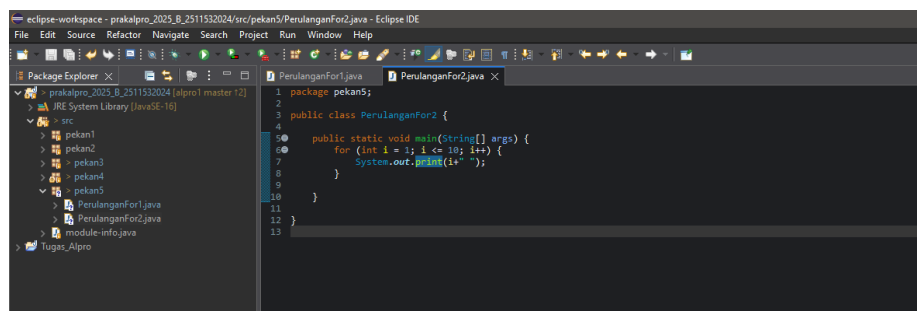
2. Lalu klik bagian “run” untuk melihat hasil sintaks yang di inputkan. Lalu akan menghasilkan output seperti berikut.



2.2.6 running kode dan output sintaks

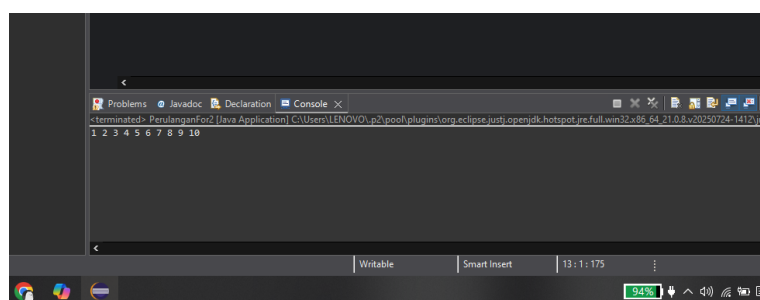
2.2.3 Program Kedua: Perulangan For 2

1. Ulangi pembuatan class seperti program sebelumnya dan beri nama class tersebut “perulanganfor2”. Lalu masukkan sintaks seperti berikut.



2.2.7 sintaks perulangan for 2

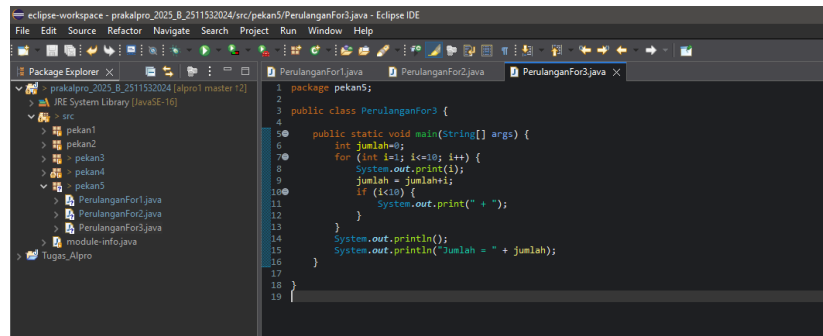
2. klik “run” untuk melihat hasil sintaks yang sudah dibuat



2.2.8 running kode dan output sintaks

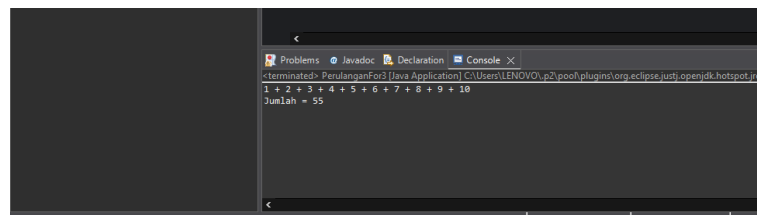
2.2.4 Program Ketiga: Perulangan For 3

1. Ulangi pembuatan class seperti program sebelumnya dan beri nama class tersebut “perulanganfor3”. Lalu masukkan sintaks seperti berikut.



2.2.9 sintaks perulangan for 3

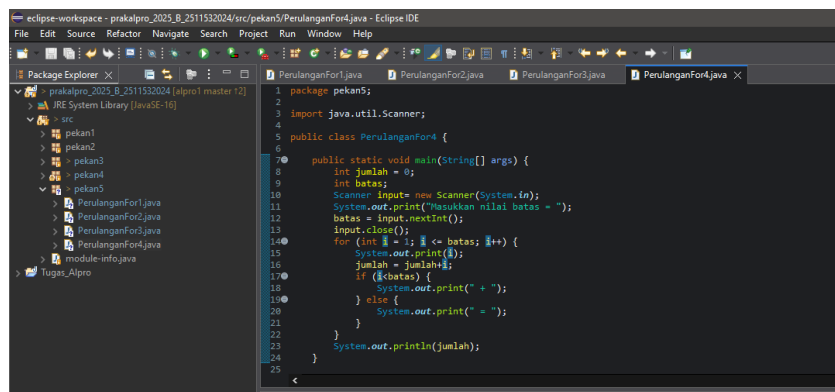
2. Klik “run” untuk melihat hasil sintaks yang sudah dibuat.



2.2.10 running code dan output sintaks

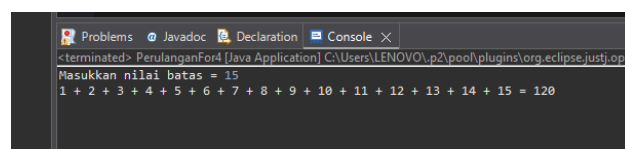
2.2.5 Program Keempat: Perulangan For 4

1. Ulangi pembuatan Class seperti program sebelumnya dan beri nama Class tersebut “perulanganfor4”. Lalu masukkan sintaks seperti berikut.



2.2.11 sintaks perulangan for 4

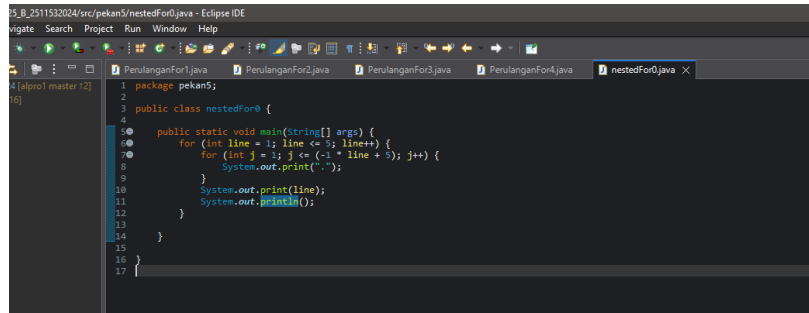
2. Klik “run” untuk melihat hasil sintaks yang sudah dibuat.



2.2.12 running code dan output sintaks

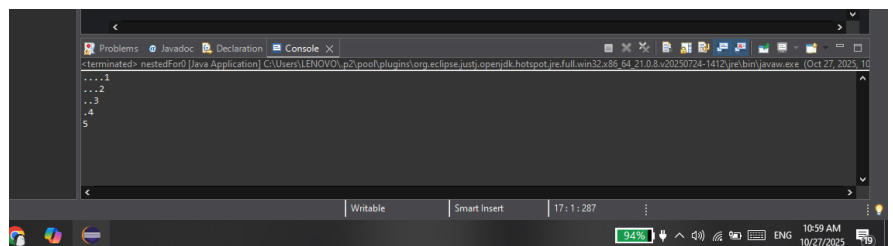
2.2.6 Program Kelima: Nested For 0

1. Ulangi pembuatan Class seperti program sebelumnya dan beri nama Class tersebut “nestedFor0”. Lalu masukkan sintaks seperti berikut.



2.2.13 sintaks nested for 0

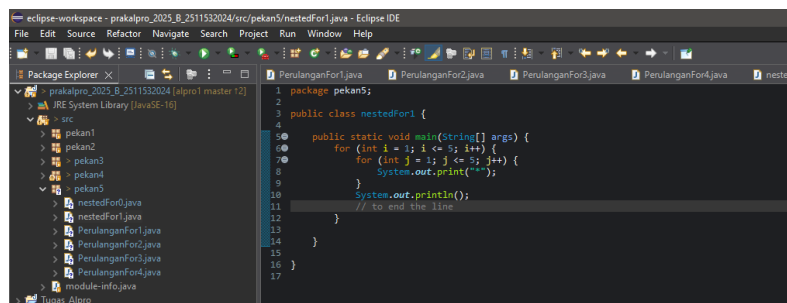
2. Klik “run” untuk melihat hasil sintaks yang sudah dibuat.



2.2.14 running kode dan output sintaks

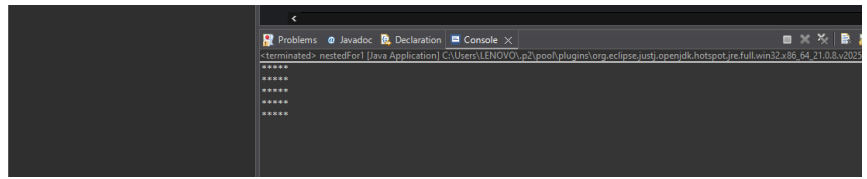
2.2.7 Program Keenam: Nested For 1

1. Ulangi pembuatan Class seperti program sebelumnya dan beri nama Class tersebut “nestedFor1”. Lalu masukkan sintaks seperti berikut.



2.2.15 sintaks nested For 1

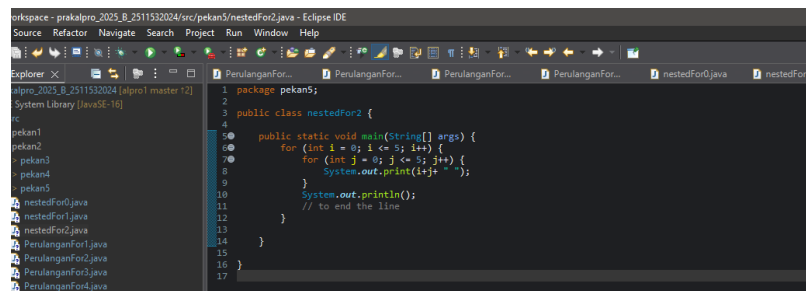
2. Klik “run” untuk melihat hasil sintaks yang sudah dibuat.



2.2.16 running code dan output sintaks

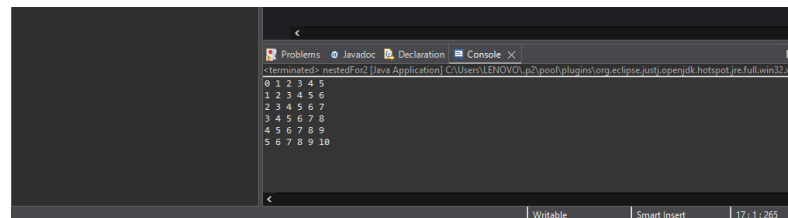
2.2.8 Program Ketujuh: Nested For 2

1. Ulangi pembuatan Class seperti program sebelumnya dan beri nama Class tersebut “nestedFor2”. Lalu masukkan sintaks seperti berikut.



2.2.17 sintaks nested for 2

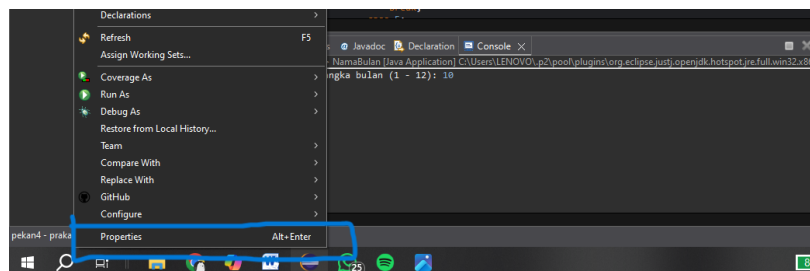
2. Klik “run” untuk melihat hasil sintaks yang sudah dibuat.



2.2.18 running code dan output sintaks

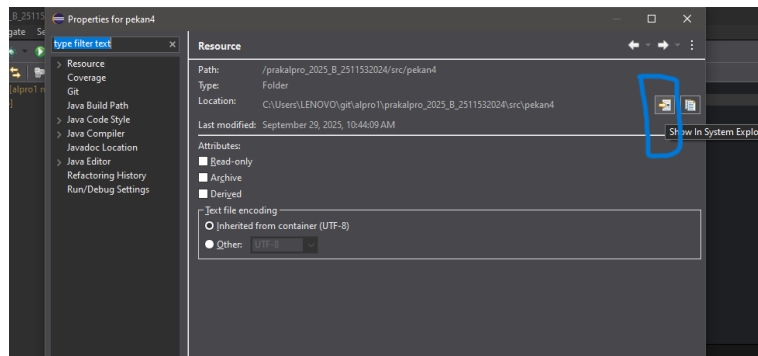
2.2.9 Commit dan Push ke GitHub

1. Klik kanan pada bagian “pekan5”. Kemudian pilih bagian “properties”.



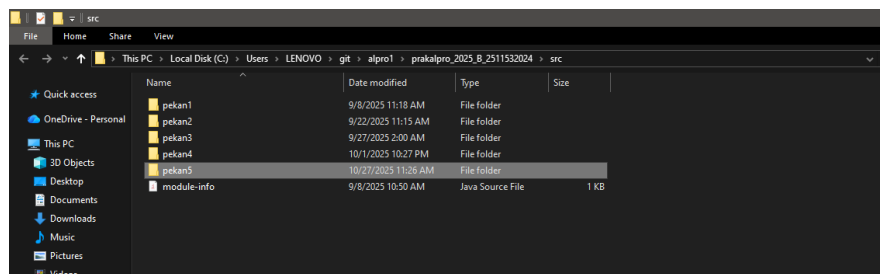
2.2.19 proses commit dan push ke GitHub

- Setelah itu keluar opsi properties. Kemudian klik “Show In System Explorer”.



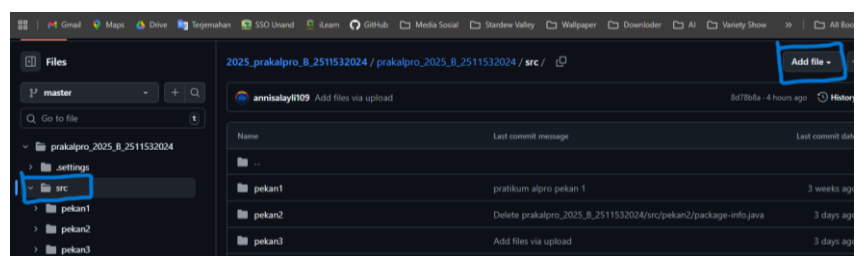
2.2.20 proses commit dan push ke GitHub

- Setelah itu akan terbuka folder file tersebut di file explorer.



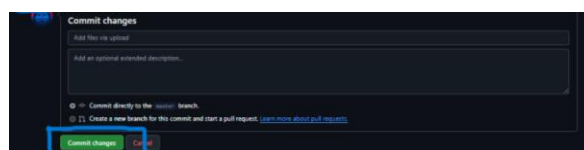
2.2.21 proses commit dan push ke GitHub

- Kemudian buka akun GitHub dan Buka pada bagian src. Dan klik pada bagian “add file”. Dan pilih “upload file”.



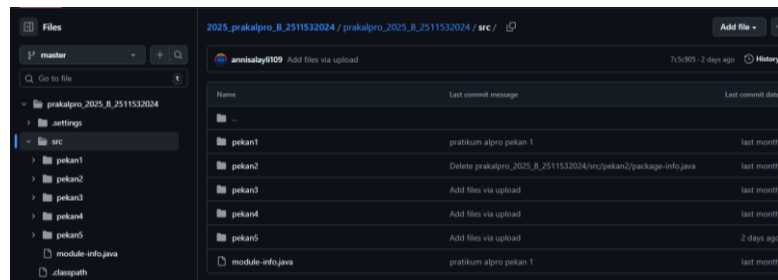
2.2.22 proses commit dan push ke GitHub

- Setelah itu tarik file “pekan5” dari file explorer ke bagian upload file dan kemudian tunggu loading file diunggah ke GitHub. Dan setelah itu klik “Commit changes”.



2.2.23 proses commit dan push ke GitHub

6. Kemudian cek akun GitHub untuk melihat apakah file yang diupload sudah masuk atau belum. Jika sudah maka upload sudah selesai.



2.2.24 proses commit dan push ke GitHub

BAB III

PENUTUP

3.1 Kesimpulan

Perulangan merupakan konsep dasar yang sangat penting dalam pemrograman karena berfungsi untuk menjalankan perintah secara berulang tanpa perlu menulis kode yang sama berkali-kali. Dalam bahasa Java, salah satu bentuk perulangan yang paling umum digunakan adalah perulangan for, yang efektif ketika jumlah pengulangan sudah diketahui dengan pasti. Selain itu, terdapat perulangan nested for yang digunakan untuk menangani proses berulang yang lebih kompleks, seperti pembuatan pola atau pengolahan data dua dimensi. Melalui pemahaman kedua jenis perulangan ini, kita dapat menulis kode yang lebih efisien, sistematis, dan mudah dipahami. Penguasaan konsep for dan nested for juga menjadi dasar penting untuk mempelajari struktur logika yang lebih lanjut dalam pemrograman seperti array, algoritma, dan pengolahan data.

3.2 Saran

Dalam mempelajari konsep perulangan for dan nested for, sebaiknya mahasiswa atau pembelajar pemrograman tidak hanya memahami teori, tetapi juga memperbanyak praktik secara langsung melalui penulisan kode program. Latihan rutin akan membantu dalam memahami alur logika, kondisi, serta cara kerja setiap bagian perulangan agar terhindar dari kesalahan seperti infinite loop atau perulangan tanpa akhir yang dapat menyebabkan program berhenti tidak semestinya. Penyusunan kode hendaknya dilakukan secara terstruktur, disertai komentar yang jelas, sehingga mudah dibaca, diuji, dan dikembangkan di tahap berikutnya.

DAFTAR PUSTAKA

- [1] Oracle, “The for Statement,” *The Java™ Tutorials*, 2024. [Online]. Available: <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/for.html>
- [2] Oracle, “Nested Loops and While Statement,” *The Java™ Tutorials*, 2024. [Online]. Available: <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/while.html>
- [3] Wahana Komputer, *Belajar Pemrograman Java untuk Pemula*. Yogyakarta: Andi Publisher, 2019.