**ALGORITMA STRUKTUR DATA**

**Praktikum – Stack**

**Lavina 2341760062**

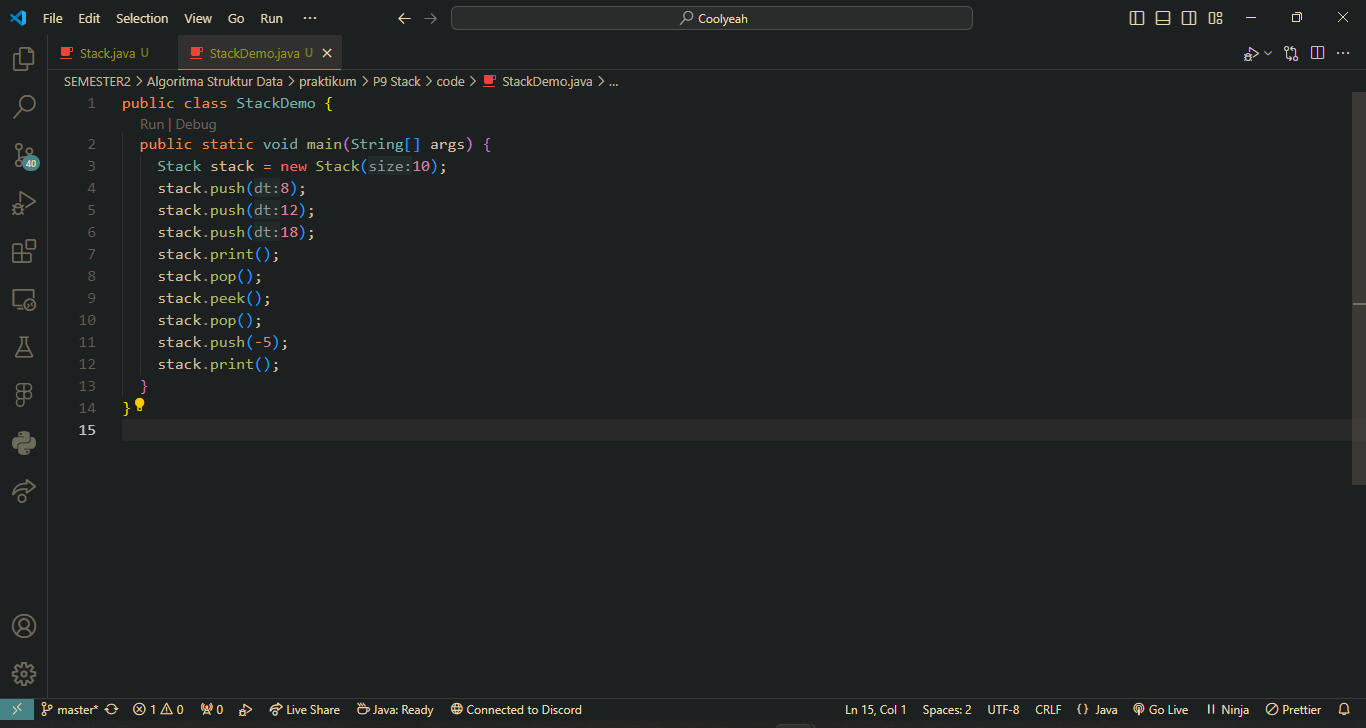
**Praktikum 1**

Stack.java

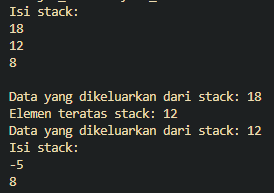




StackDemo.java



Output:



**Pertanyaan**

1. Pada method pop(), mengapa diperlukan pemanggilan method isEmpty()? Apa yang terjadi jika tidak ada pemanggilan isEmpty()?

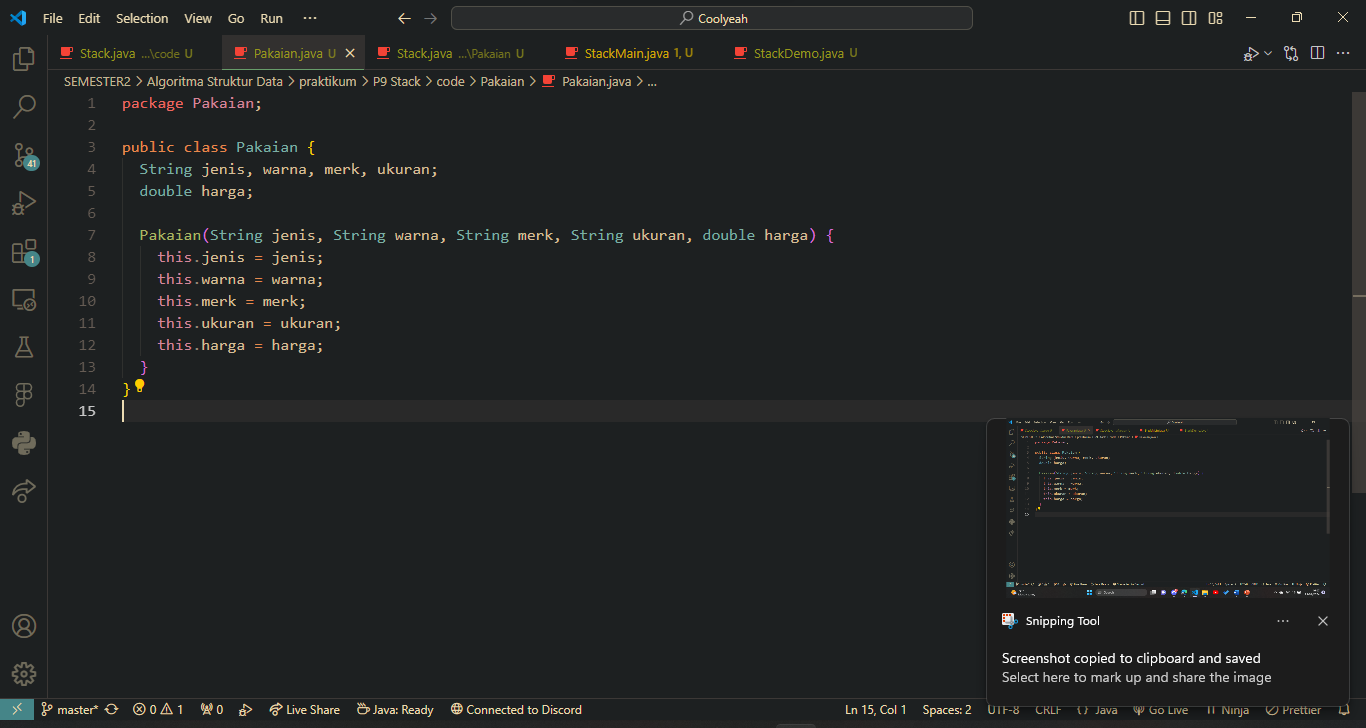
**Jawab :** Method tersebut dipanggil untuk mengecek apakah stacknya kosong atau tidak, karena jika stacknya kosong maka proses pop atau penghapusan data tidak bisa dilakukan.

1. Jelaskan perbedaan antara method peek() dengan method pop() pada class Stack.

**Jawab :** Perbedaannya adalah pada method peek() hanya menampilkan data yang ada dipaling atas stack tanpa menghapusnya sedangkan method pop() mengambil data teratas dari stack sehingga menghilangkan data tersebut juga.

**Percobaan 2**

Pakaian.java



Stack.java



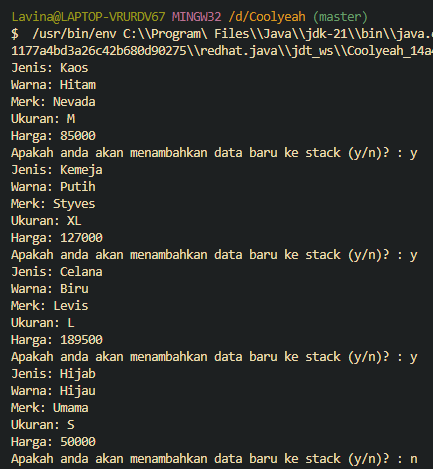


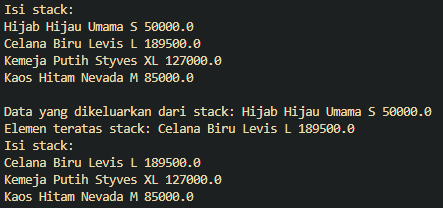
StackMain.java





Output :





**Pertanyaan**

1. Berapa banyak data pakaian yang dapat ditampung di dalam stack? Tunjukkan potongan kode program untuk mendukung jawaban Anda tersebut!

**Jawab :** Data yang bisa ditampung adalah 5, ditunjukkan pada sintaks **Stack stk = new Stack(5);**

1. Perhatikan class StackMain, pada saat memanggil fungsi push, parameter yang dikirimkan adalah p. Data apa yang tersimpan pada variabel p tersebut?

**Jawab :** Variabel p berisi data jenis, warna, merk, ukuran, harga yang diinputkan melalui keyboard.

1. Apakah fungsi penggunaan do-while yang terdapat pada class StackMain?

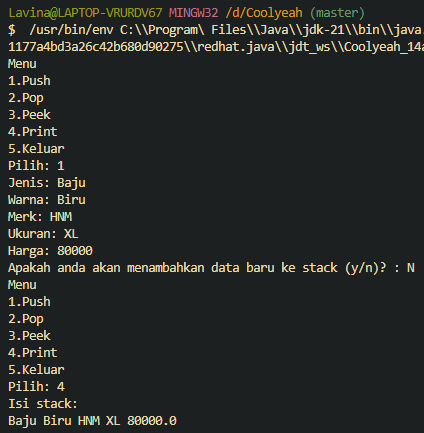
**Jawab :** Fungsinya agar bisa mengisi lebih dari 1 jenis pakaian, sehingga jika memilih “y” maka formnya akan diulang.

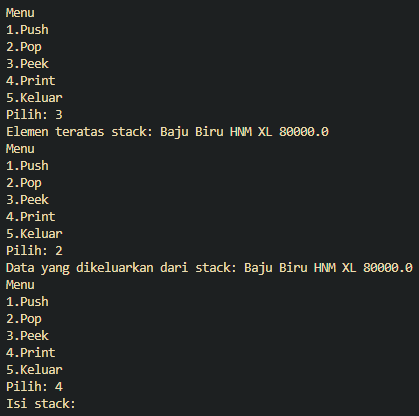
1. Modifikasi kode program pada class StackMain sehingga pengguna dapat memilih operasi-operasi pada stack (push, pop, peek, atau print) melalui pilihan menu program dengan memanfaatkan kondisi IF-ELSE atau SWITCH-CASE!





Output :





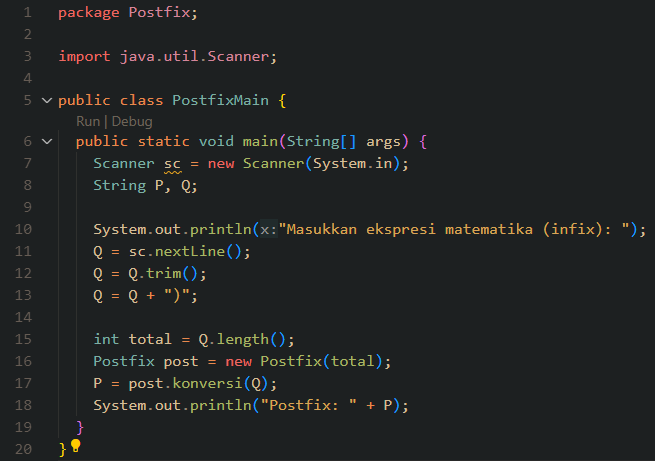
Percobaan 3

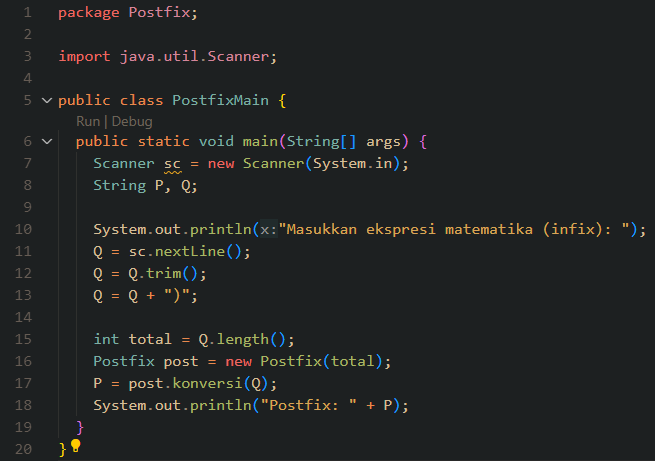
Postfix.java



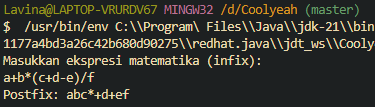


PostfixMain.java





Output :



**Pertanyaan**

1. Perhatikan class **Postfix**, jelaskan alur kerja method **derajat**!

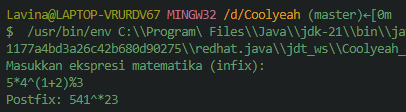
**Jawab :** Method ini berfungsi untuk menentukan tingkat atau operasi mana yang lebih tinggi untuk nantinya akan dioperasikan terlebih dahulu.

1. Apa fungsi kode program berikut?



**Jawab :** Untuk mengambil setiap karakter dari inputan atau memisahkan inputan dari pengguna menjadi array yang berisi karakter-karakter.

1. Jalankan kembali program tersebut, masukkan ekspresi **5\*4^(1+2)%3**. Tampilkan hasilnya!



1. Pada soal nomor 3, mengapa tanda kurung tidak ditampilkan pada hasil konversi? Jelaskan !

**Jawab :** Karena tanda kurung hanya di-pop, tidak perlu dimasukkan ke postfix