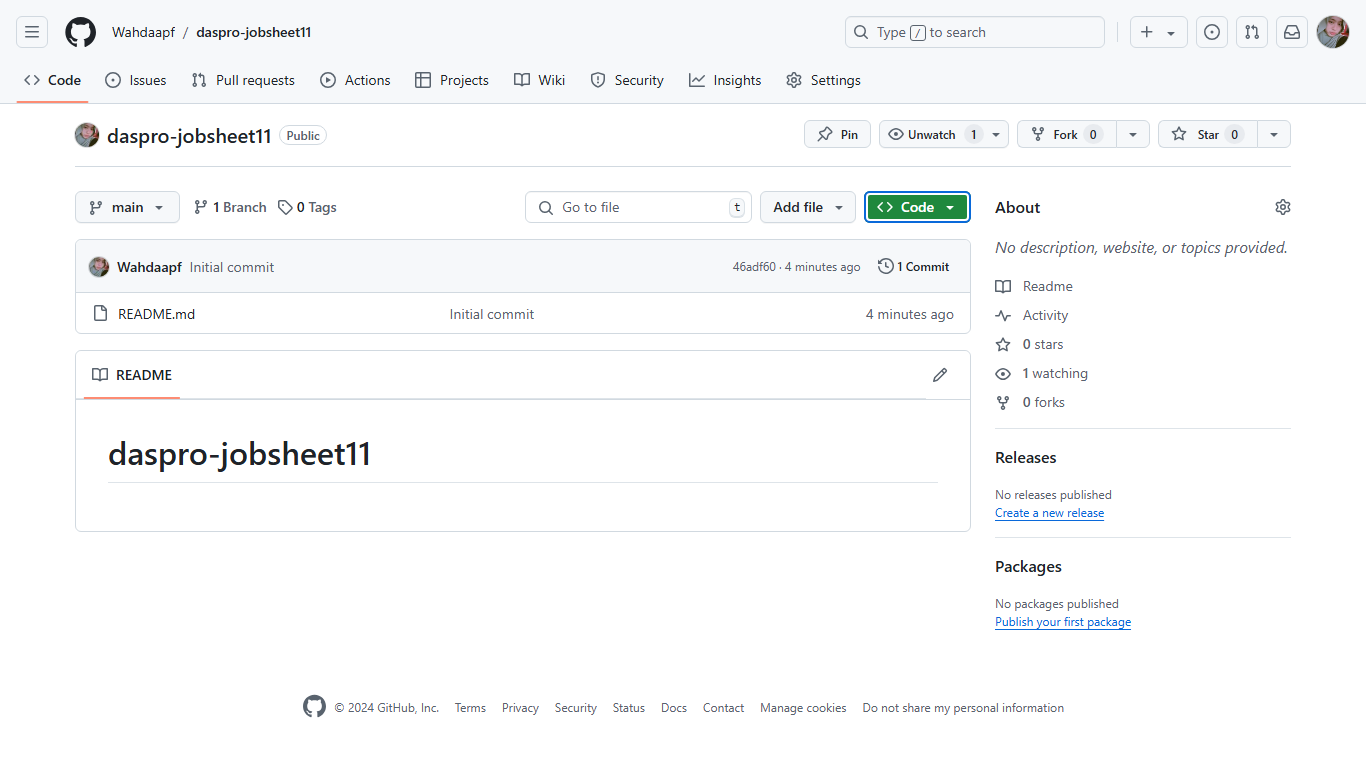
Nama: Wahda Adella Putri Febriana

NIM : 244107020156

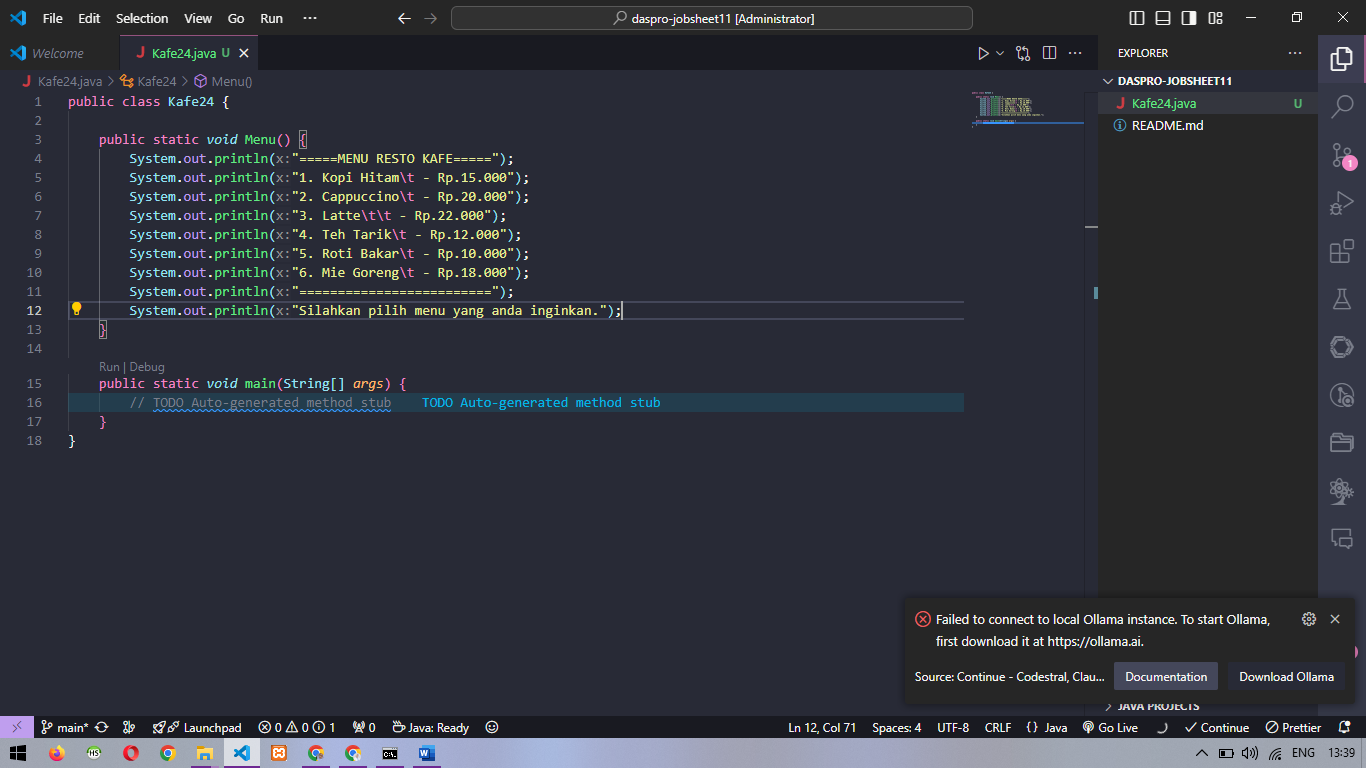
Kelas: 1B

**Percobaan 1**

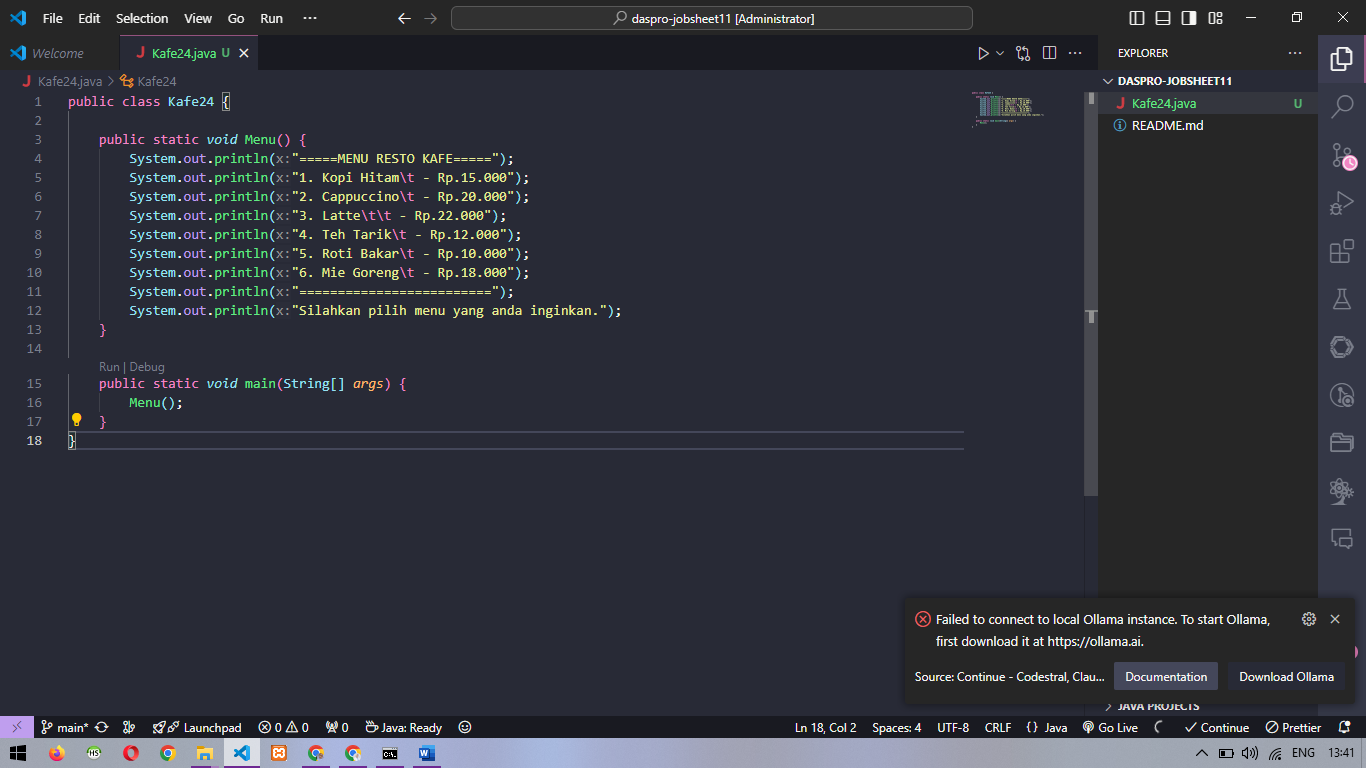
1. Buat Repository “daspro-jobsheet11”



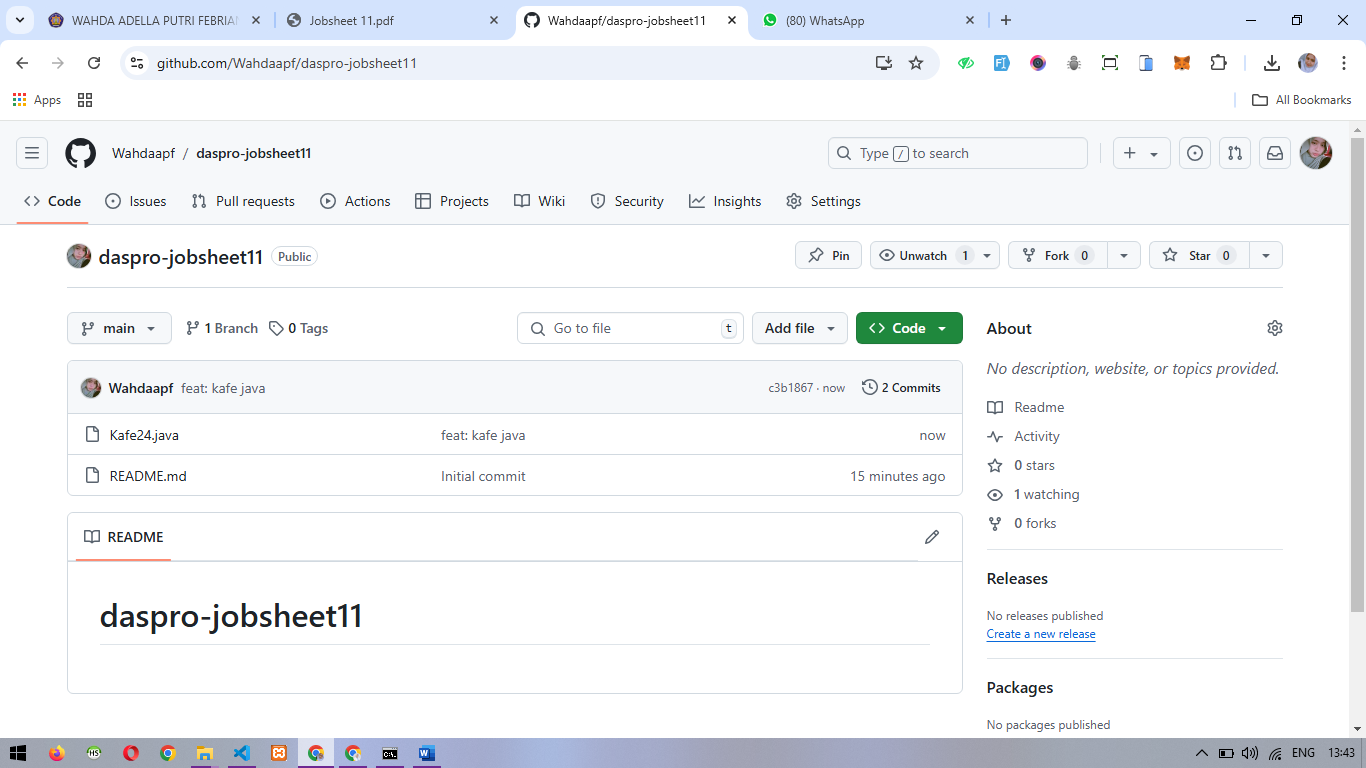
1. Clone dan Open di VSC, buat file Kafe24.java, dan buat fungsi menu



1. Panggil menu di main



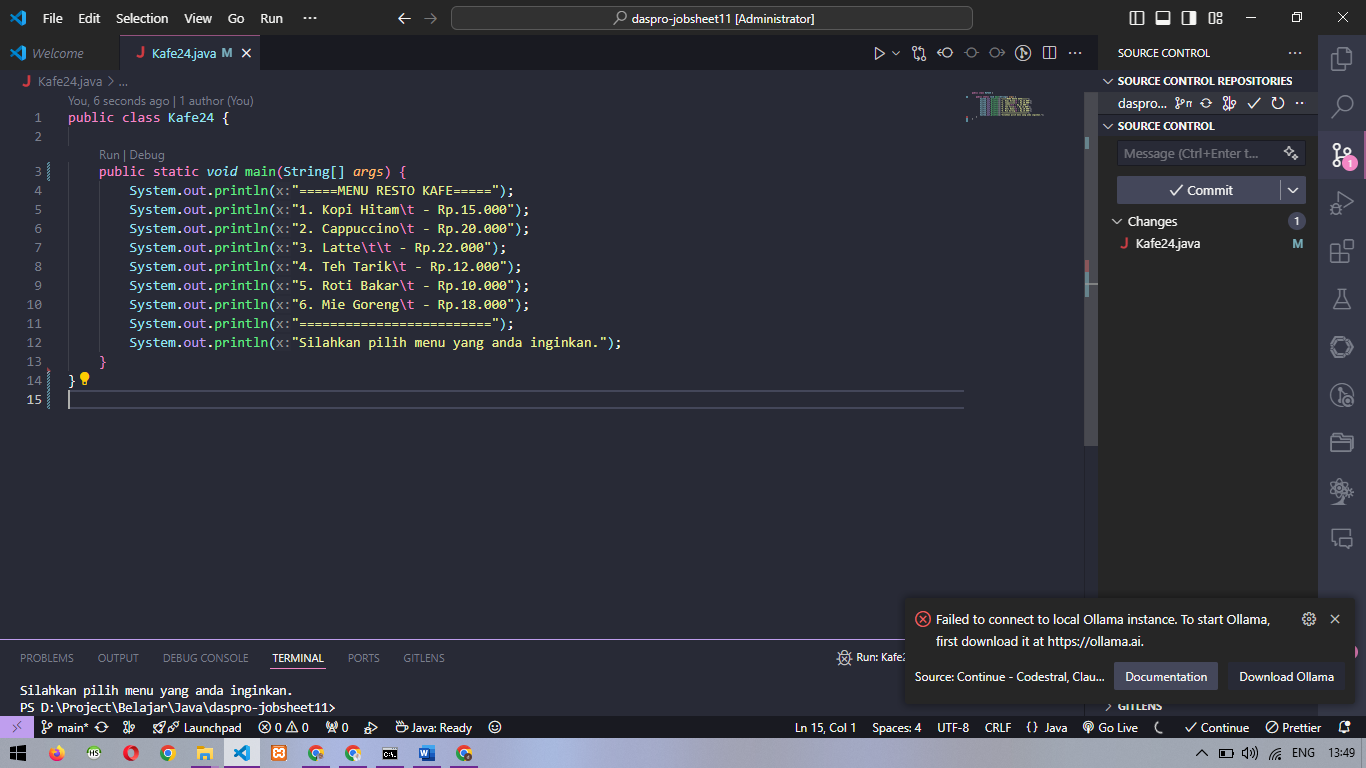
1. Push dan commit di github



**Pertanyaan 1**

1. Tidak, fungsi tanpa parameter tidak selalu digunakan hanya untuk fungsi “void” saja. Sebuah fungsi tanpa parameter dapat mengembalikan nilai apa pun, tergantung pada deklarasi tipenya
2. Tidak, karena kodingan print berada pada fungsi menu, jika menu tidak di panggil maka, print tidak akan nampil.

Modifikasi agar tidak perlu memanggil fungsi



1. a. Membuat kode lebih rapi, modular, dan mudah dipahami.

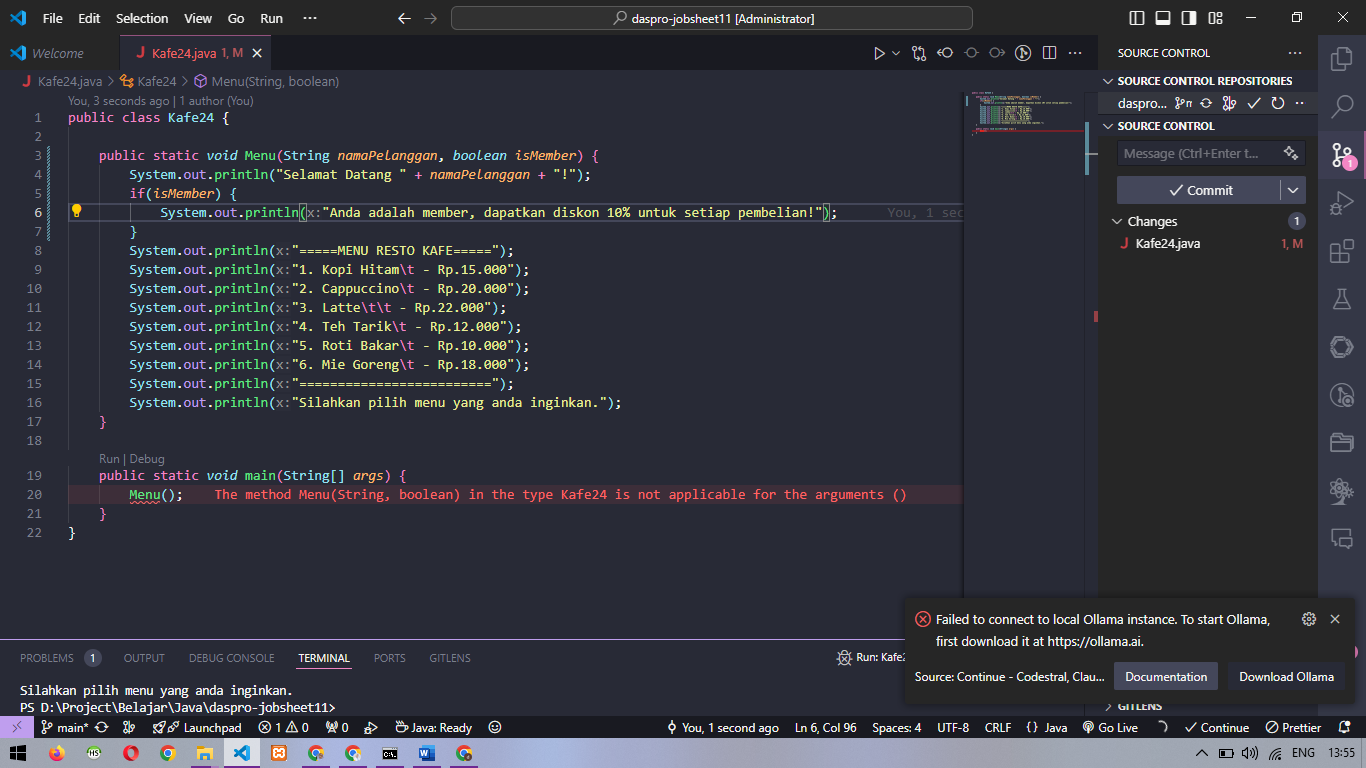
b. Mengurangi duplikasi kode dan meningkatkan efisiensi.

c. Mempermudah debugging, pemeliharaan, dan pengujian.

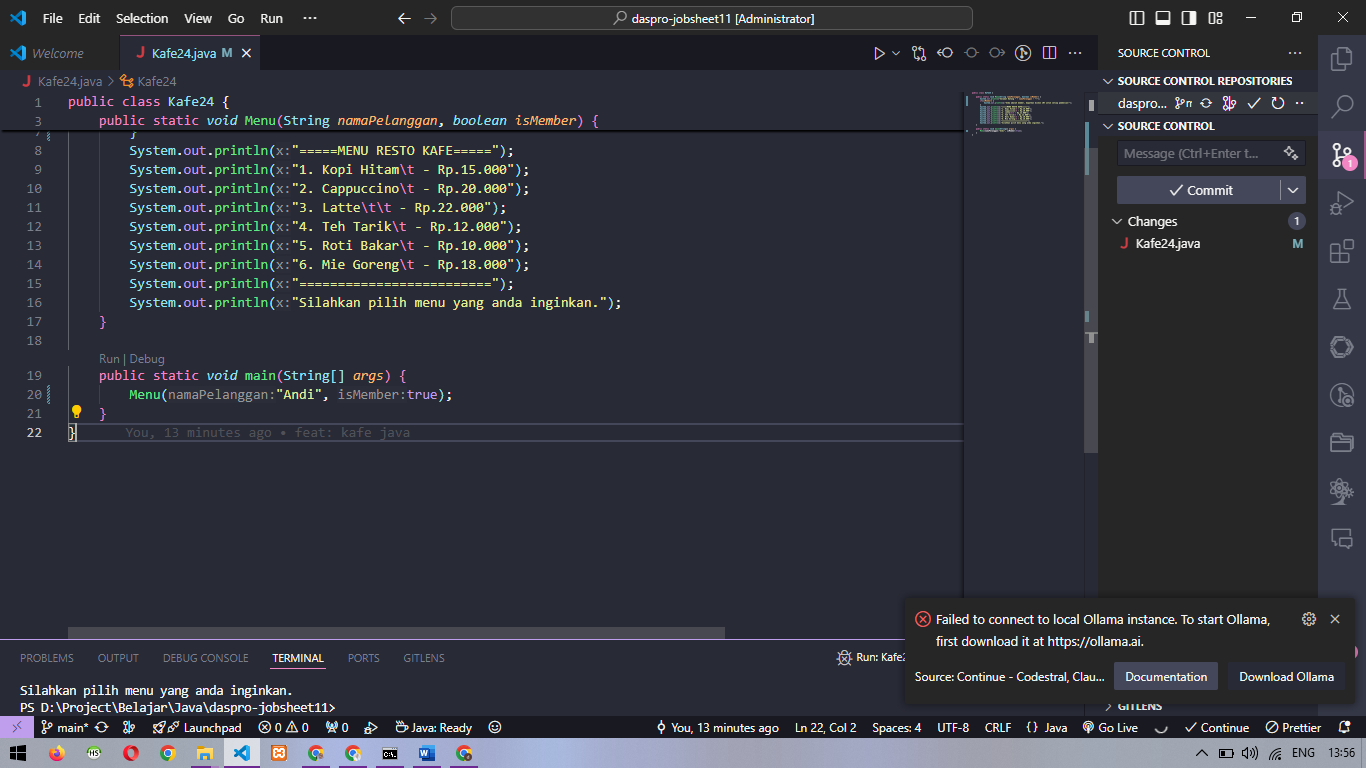
d. Memungkinkan penggunaan kembali kode dan fleksibilitas yang lebih besar.

**Percobaan 2**

1. Update code fungsi menu

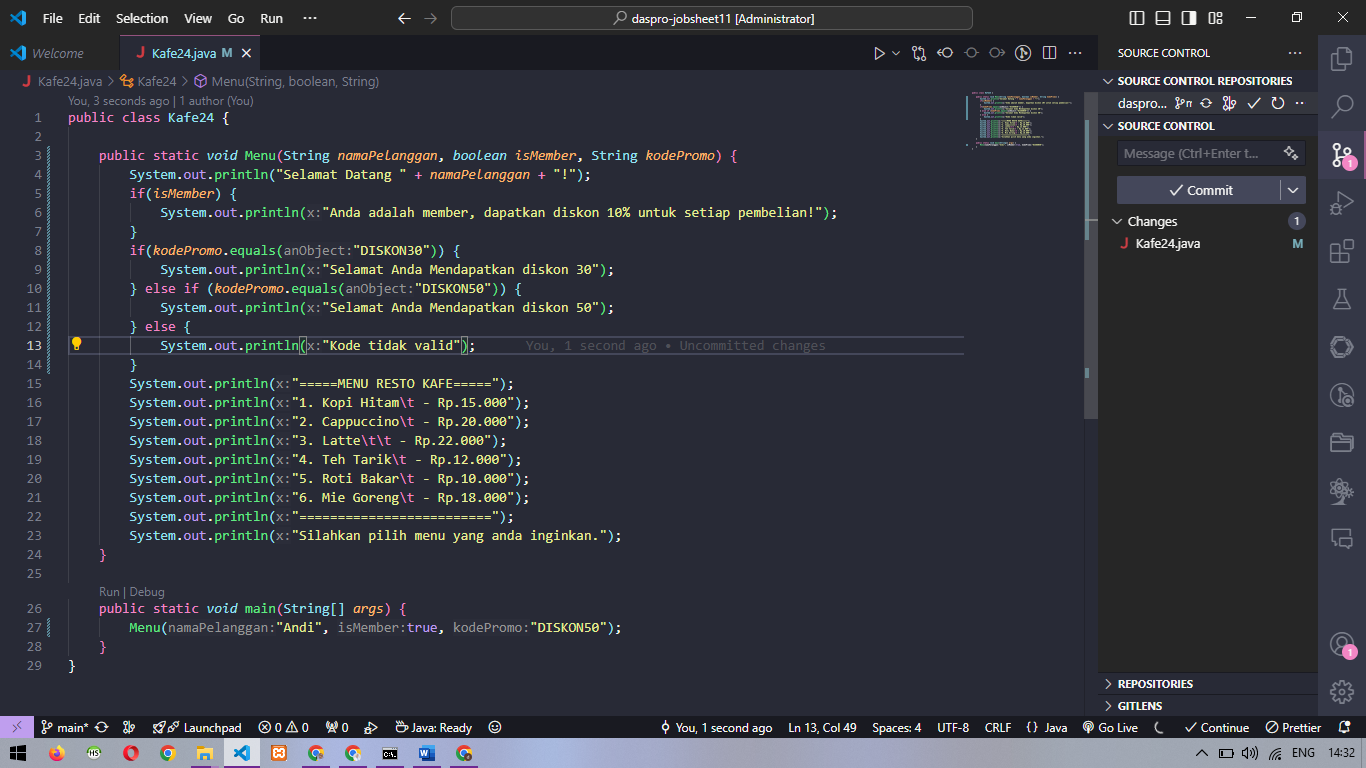


1. Update code pemanggilan fungsi menu

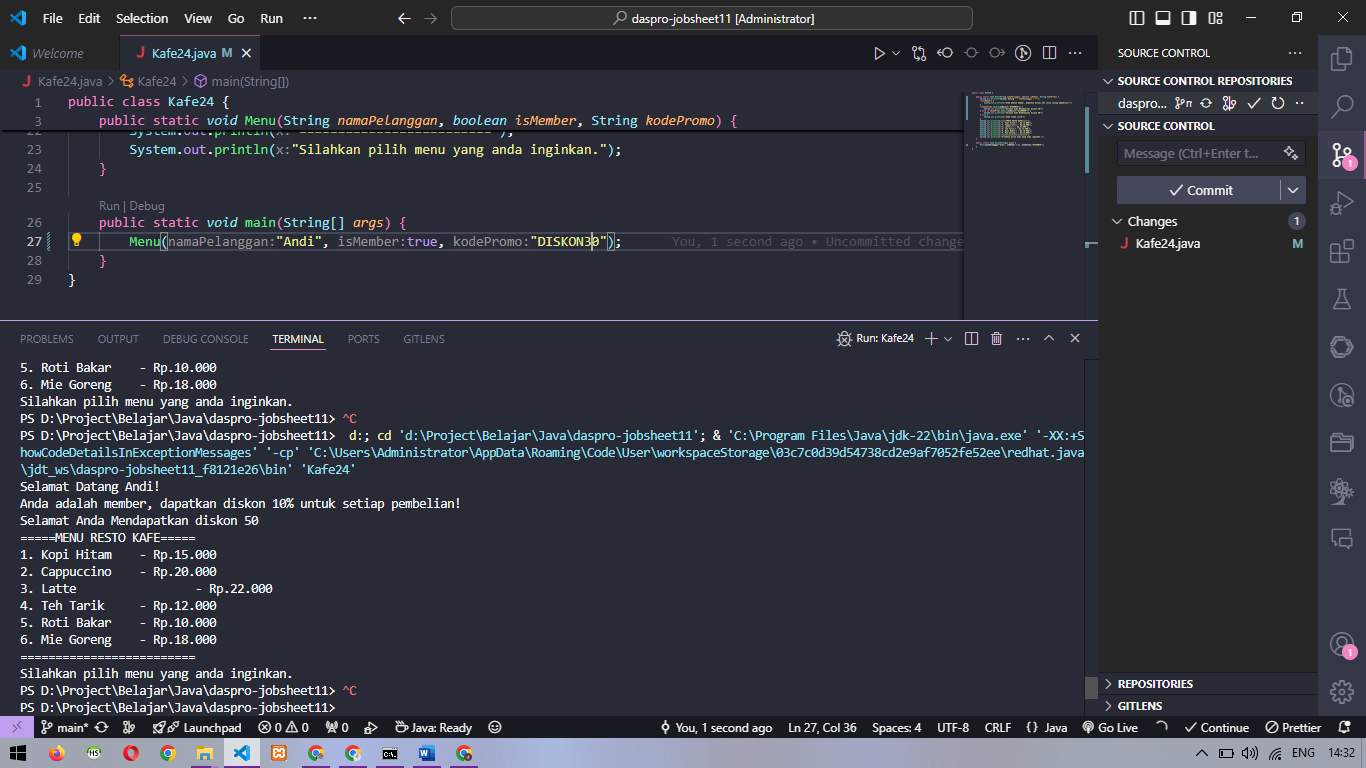


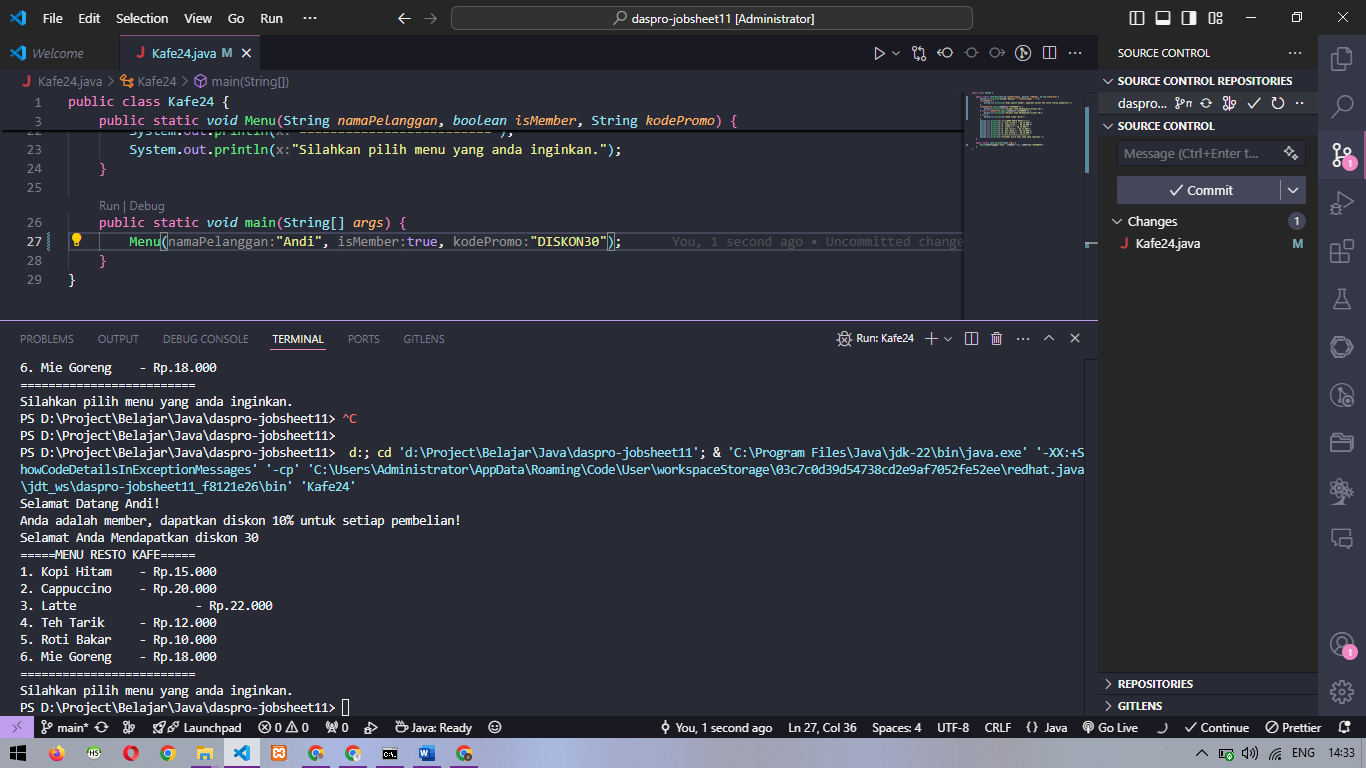
**Pertanyaan 2**

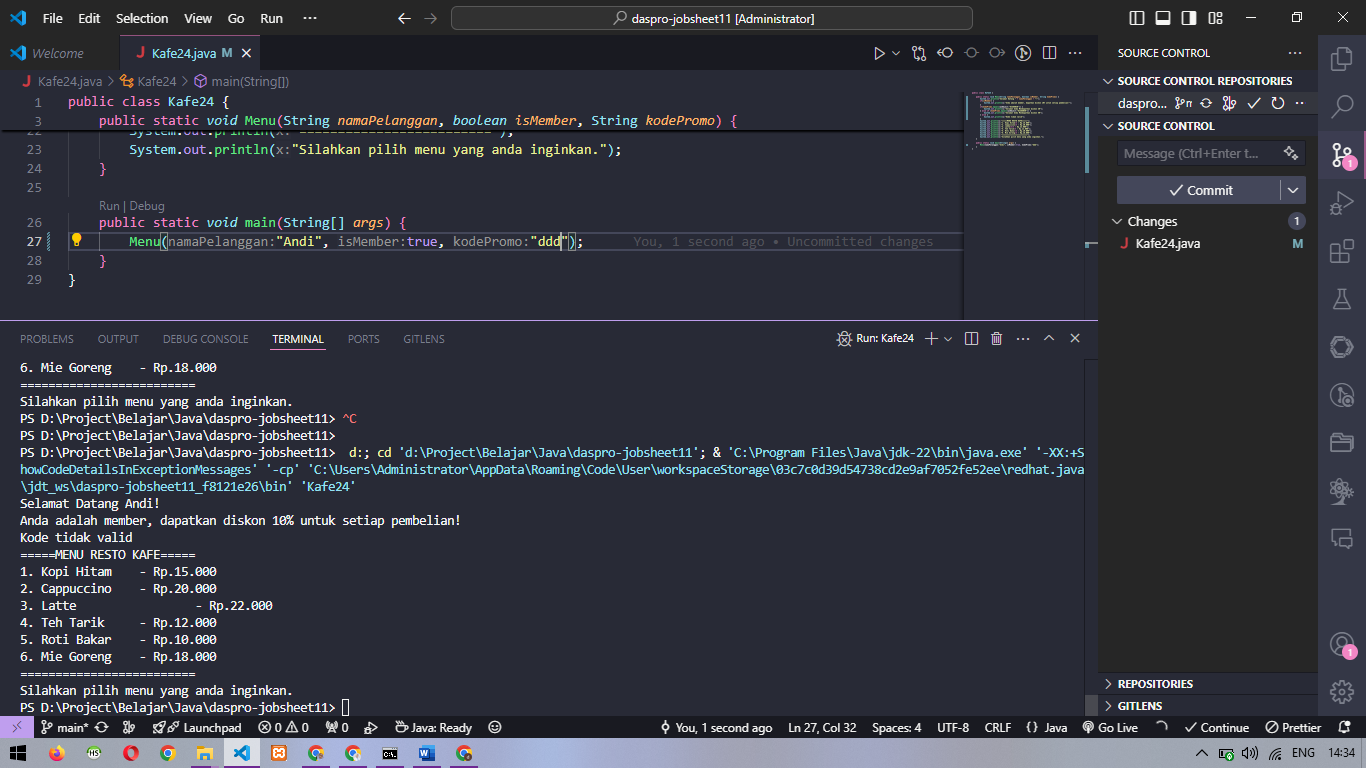
1. Membuat fungsi menjadi flexible, dinamis, dan dapat digunakan Kembali dalam berbagai konteks
2. Tidak, parameter digunakan tempat menampung sementara untuk diteruskan ke fungsi, variable yaitu tempat penampung data umum untuk menyimpan nilai Dimana saja
3. Parameter isMember digunakan untuk mengecheck apakah pelanggan merupakan member atau bukan, tipe data nya Boolean, saat pemanggilan fungsi menu() isMember akan ditaruh sebagai parameter kedua dengan opsi true atau false, lalu melakukan pengechekan di fungsi menu(), jika true maka akan memunculkan print “anda adalah member dan mendapatkan diskon sebesar 10%”
4. Error, dikarenakan fungsi nya membutuhkan parameter
5. Modifikasi kode program



Output

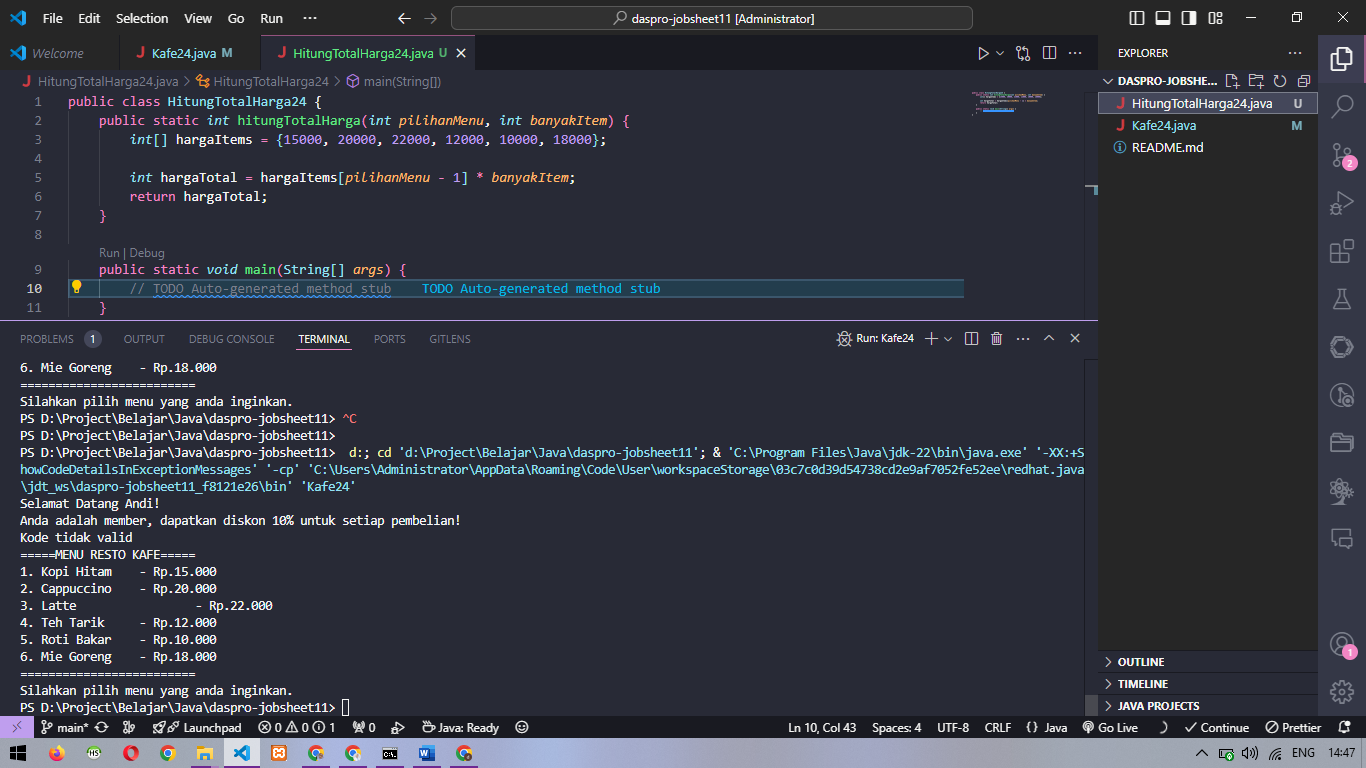




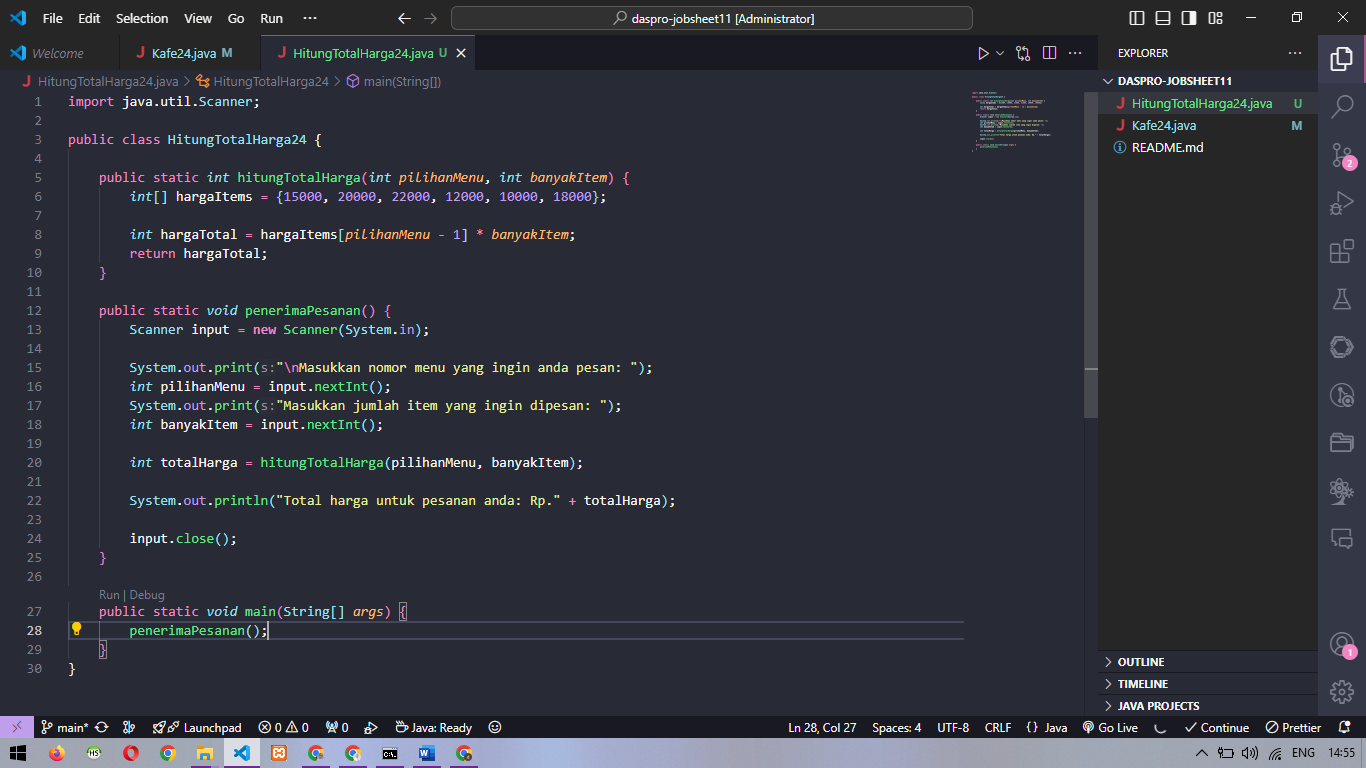


**Percobaan 3**

1. Buat file HitungTotalHarga24.java dan fungsi hitungTotalHarga

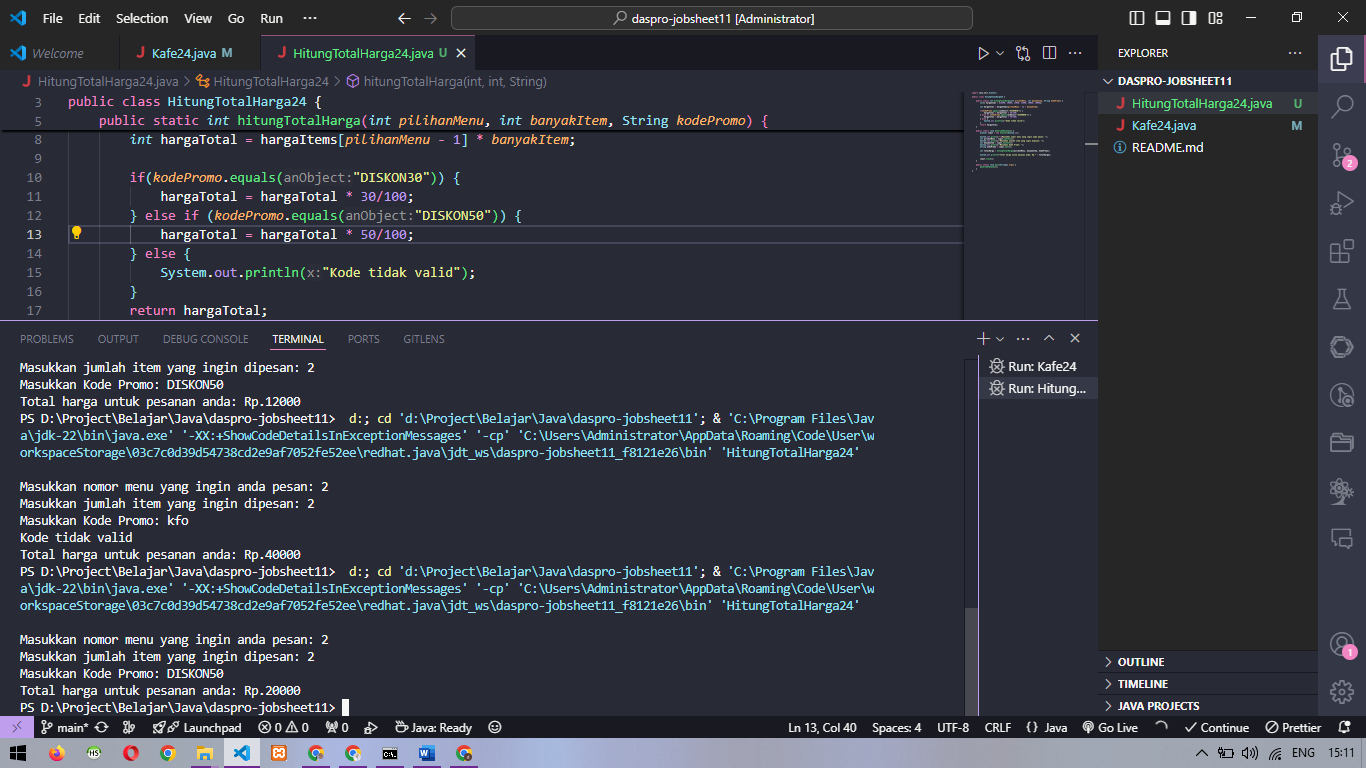


1. Buat fungsi penerimaPesanan, dan panggil di main

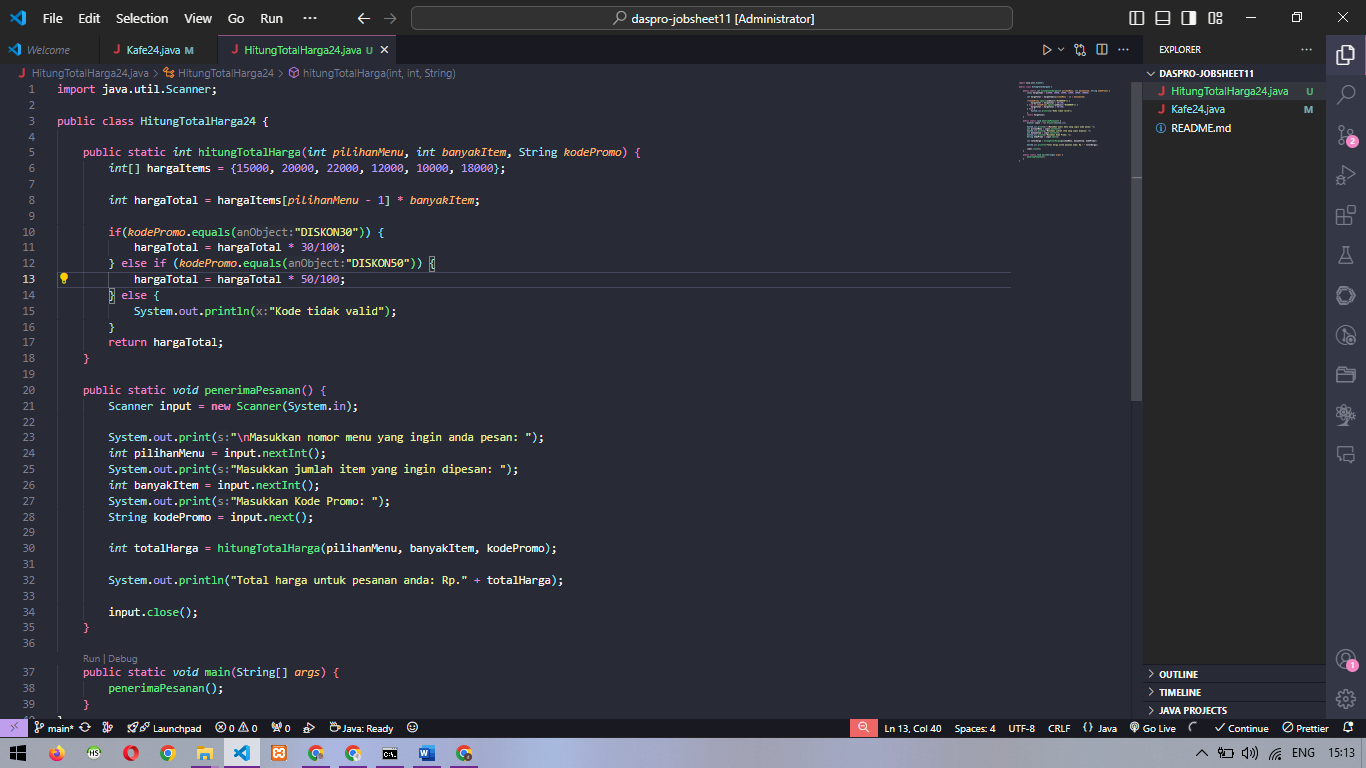


**Pertanyaan**

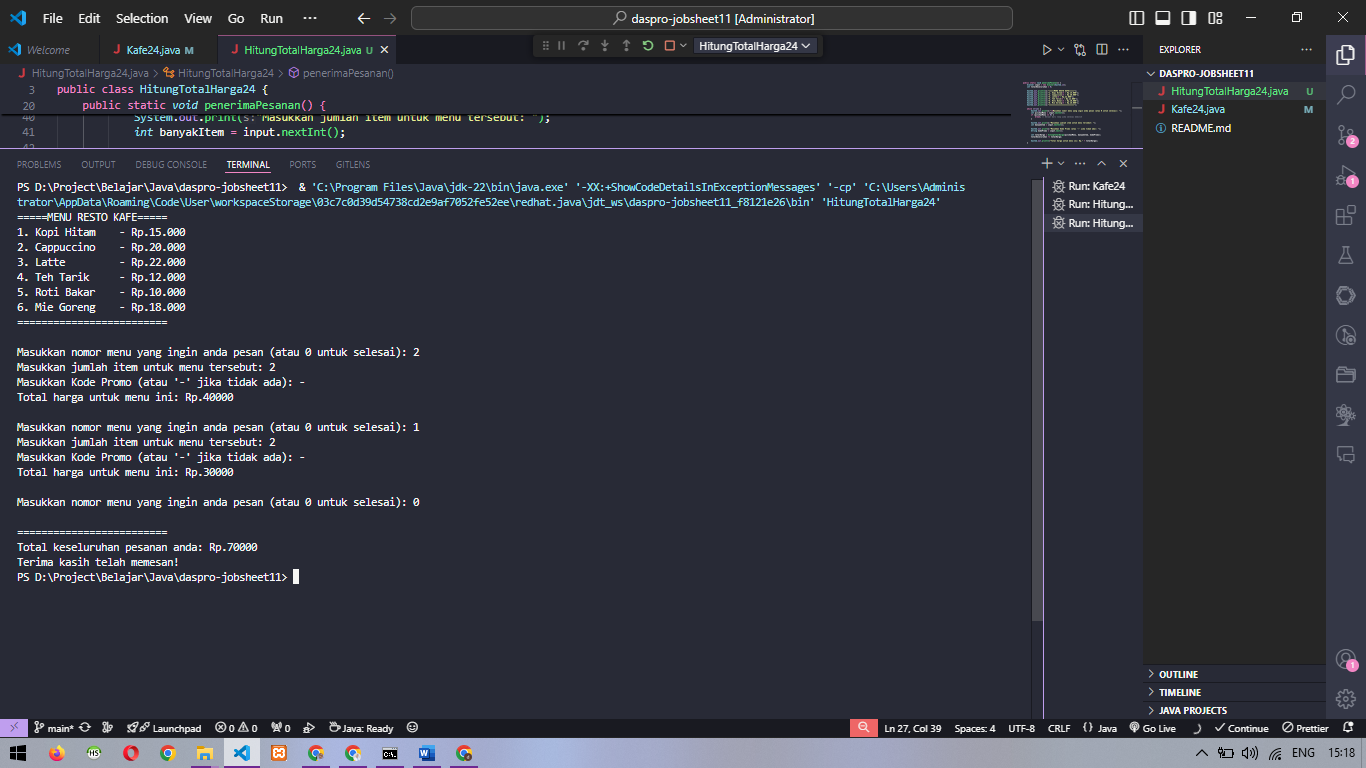
1. Suatu fungsi membutuhkan nilai kembalian (**return**) ketika fungsi tersebut perlu **mengembalikan hasil perhitungan, data, atau nilai tertentu** ke pemanggilnya.
2. Output



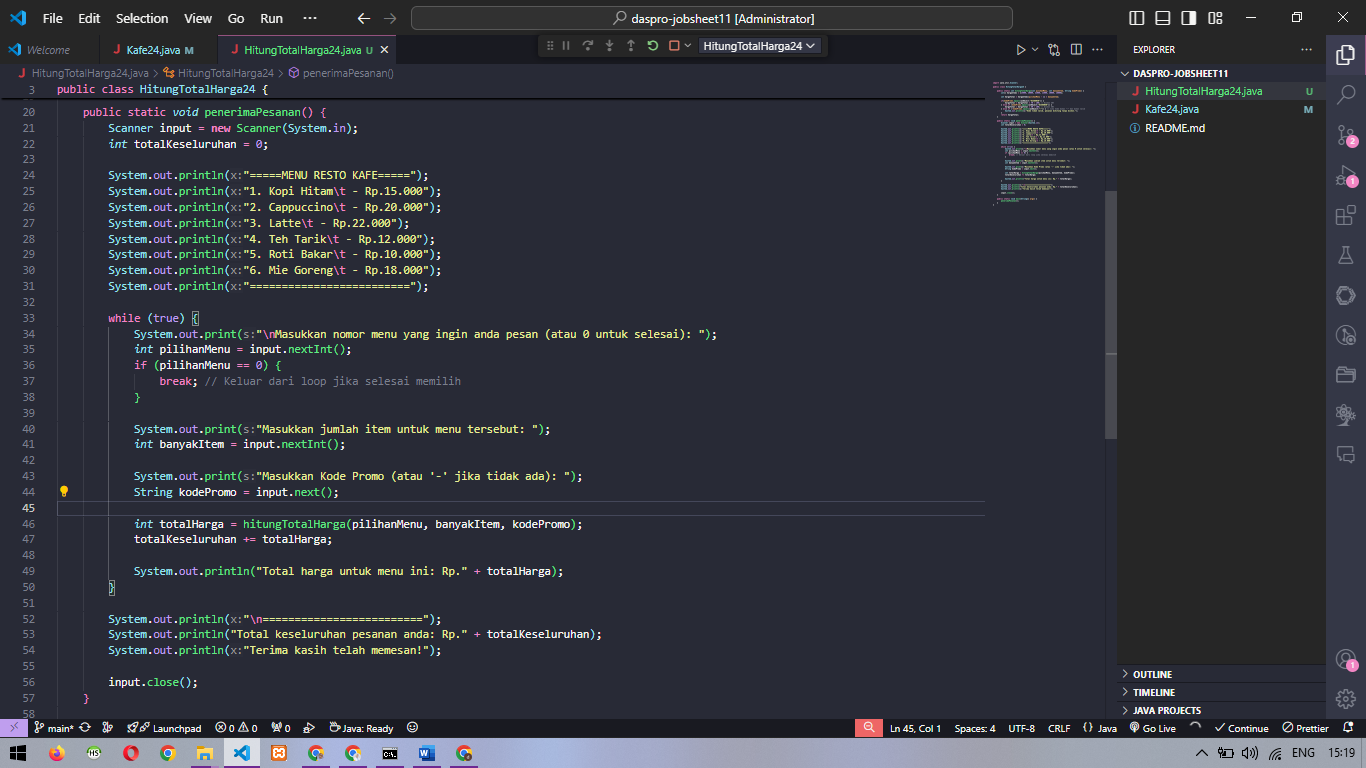
Kode



1. output

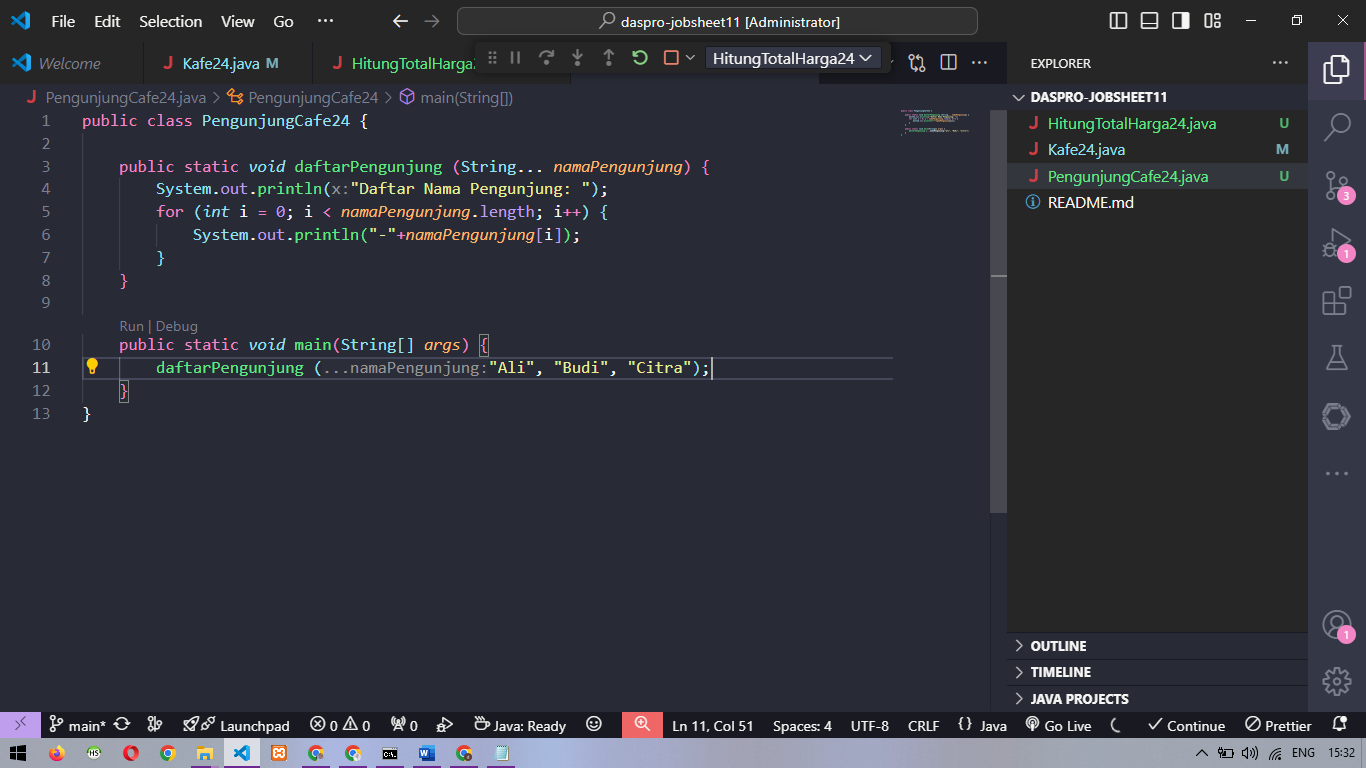


Kodingan



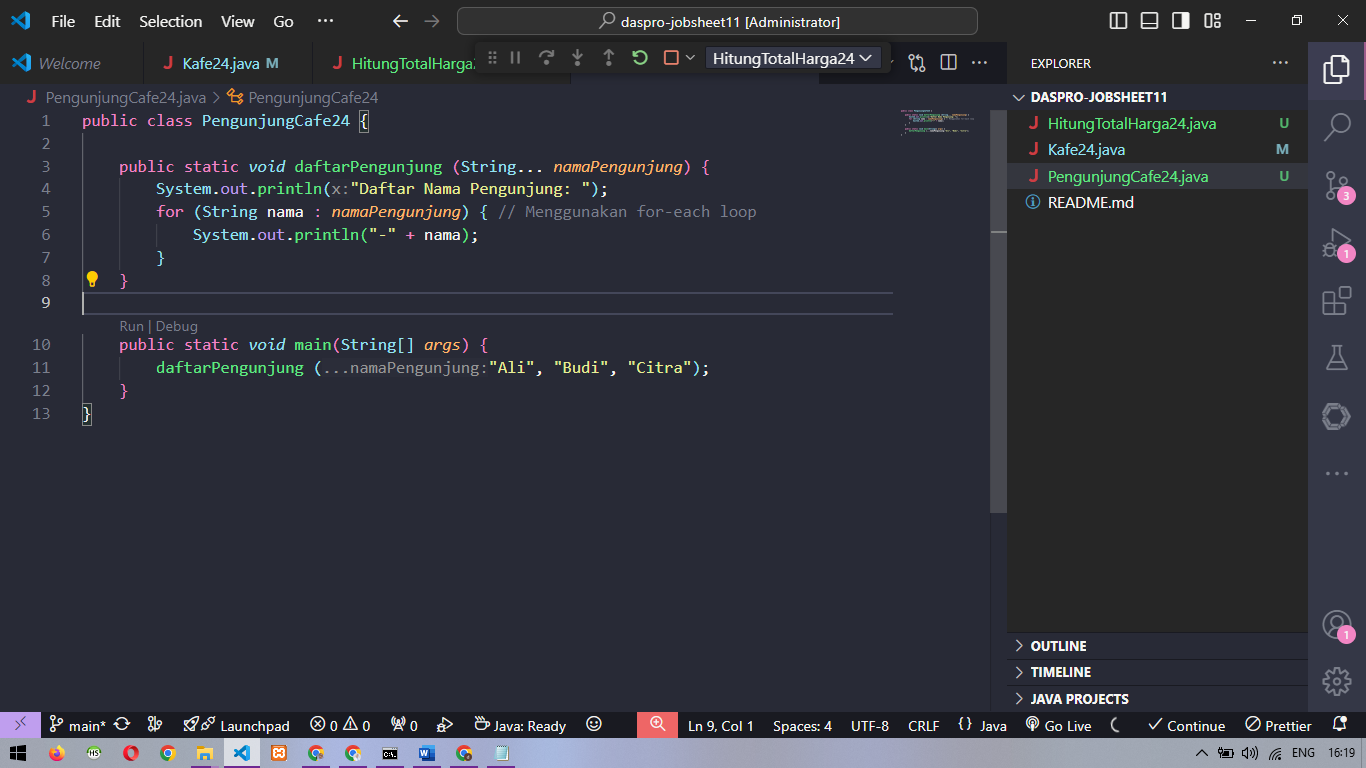
**Percobaan 4**

1. Buat file pengunjungKafeNo24.java, dan tambahkan kodingan berikut



**Pertanyaan**

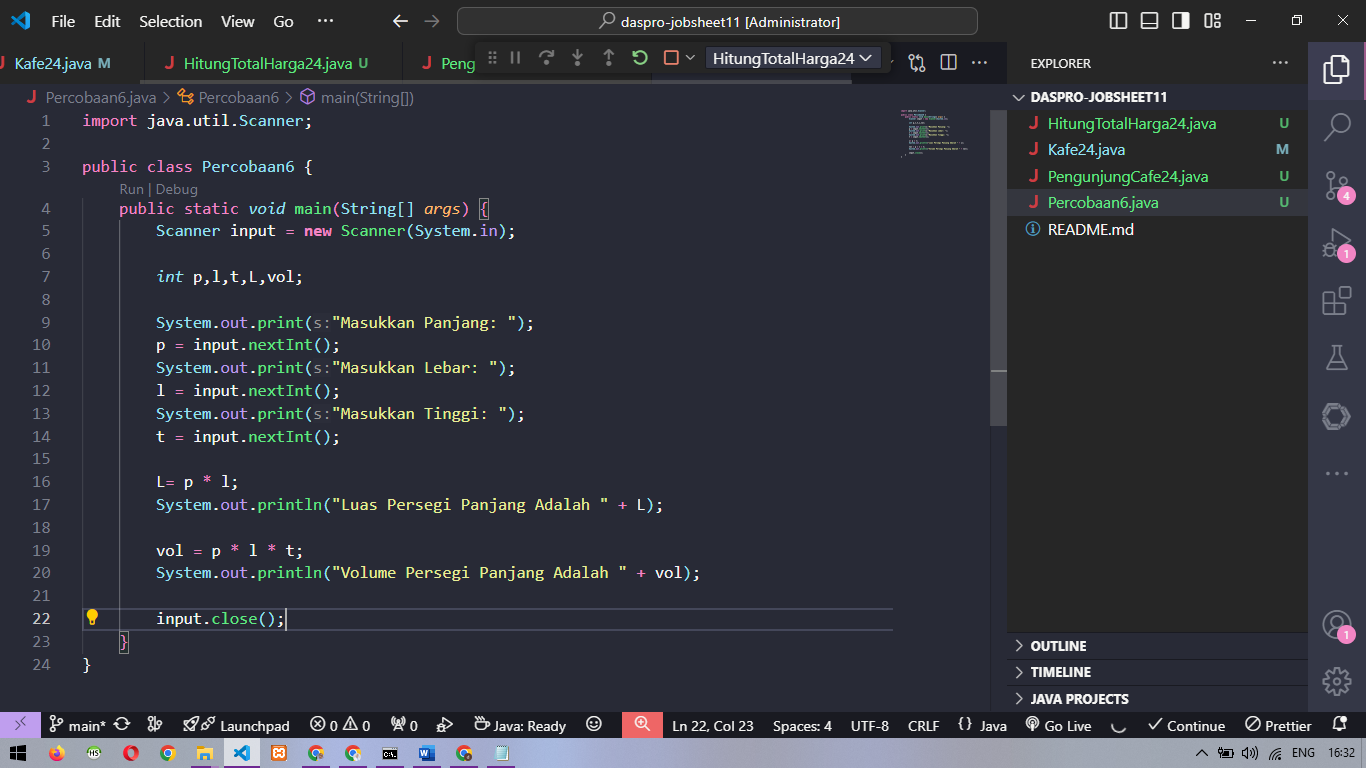
1. bertujuan untuk memungkinkan metode menerima **jumlah argumen yang tidak ditentukan sebelumnya**.
2. Ubah menjadi for each



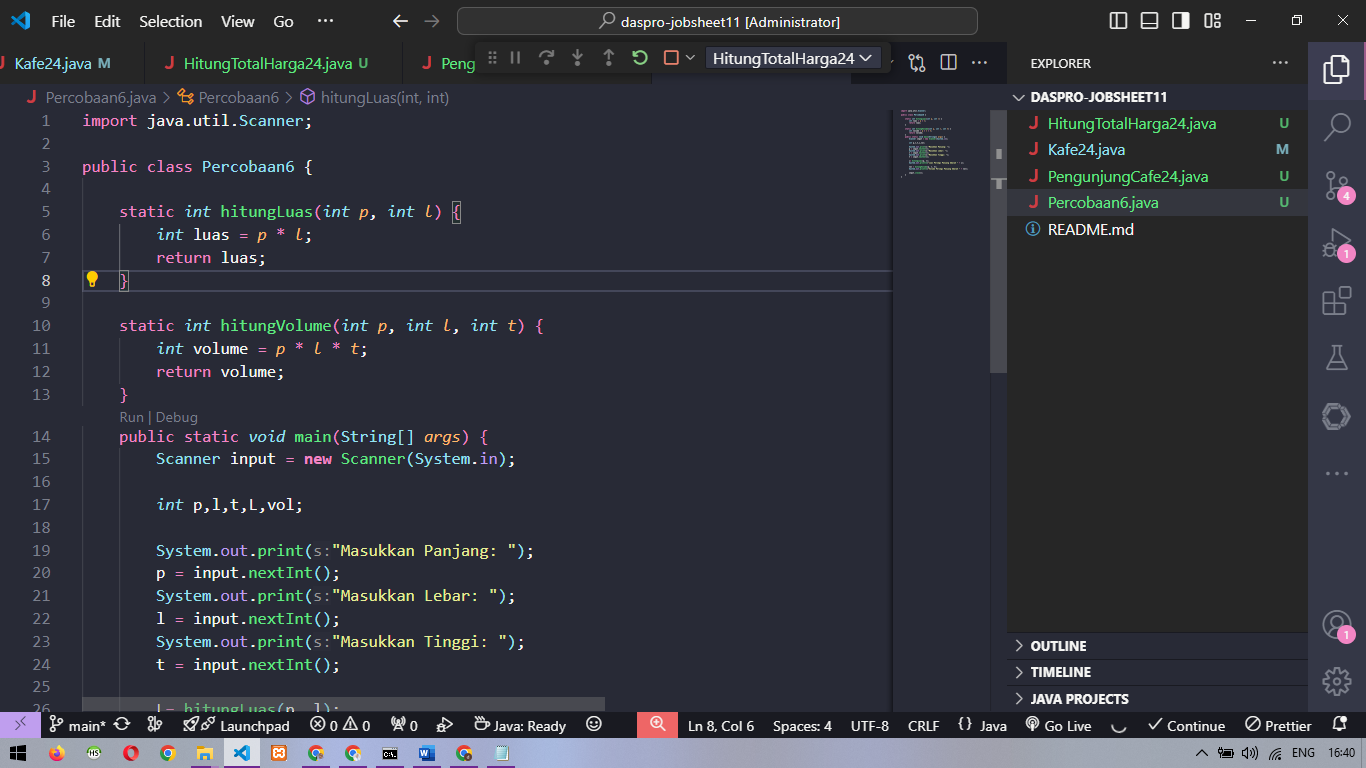
1. Tidak, tidak dapat menggunakan dua parameter varargs (**...**) dalam satu fungsi. Karena parameter varargs akan menangkap semua argumen yang sesuai, sehingga akan membingungkan compiler mengenai argumen mana yang harus masuk ke parameter varargs tertentu.

**Percobaan 5**

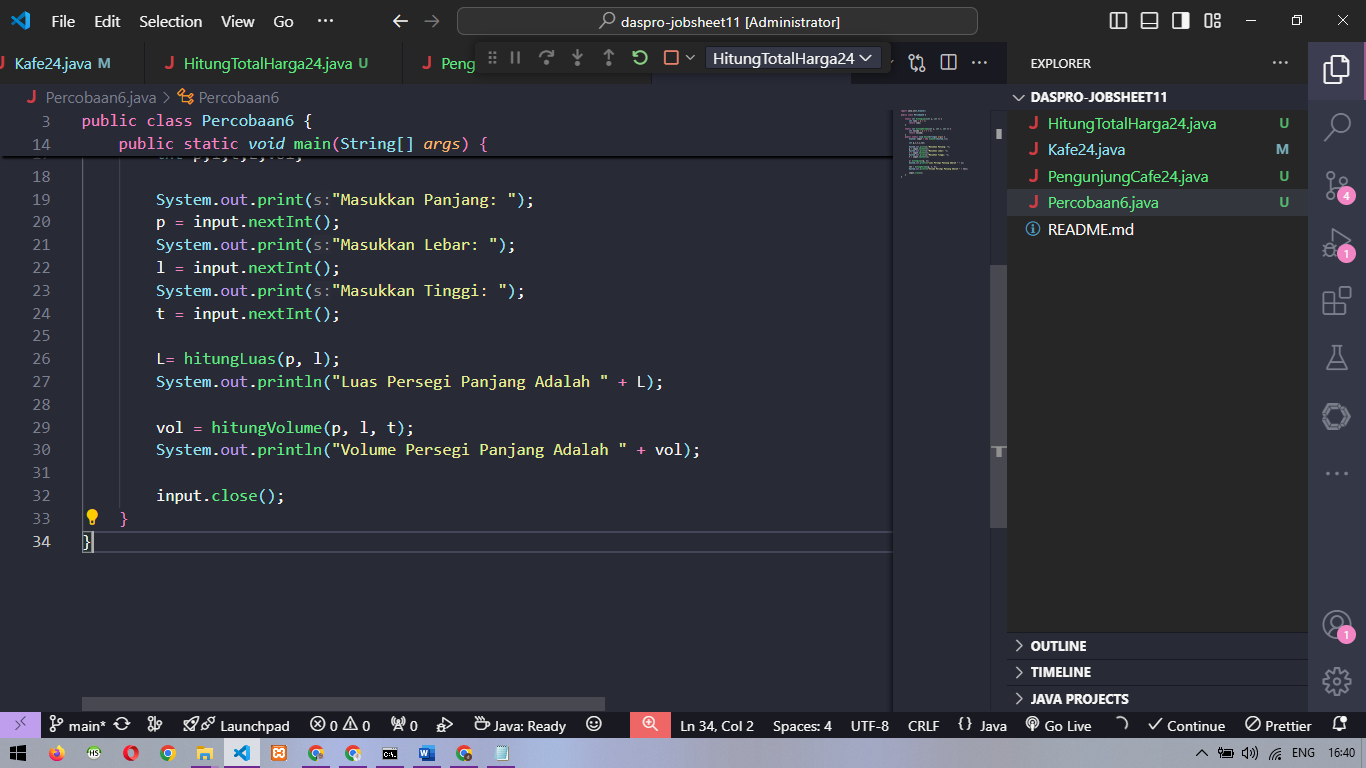
1. Buat file percobaan6.java, dan kode berikut



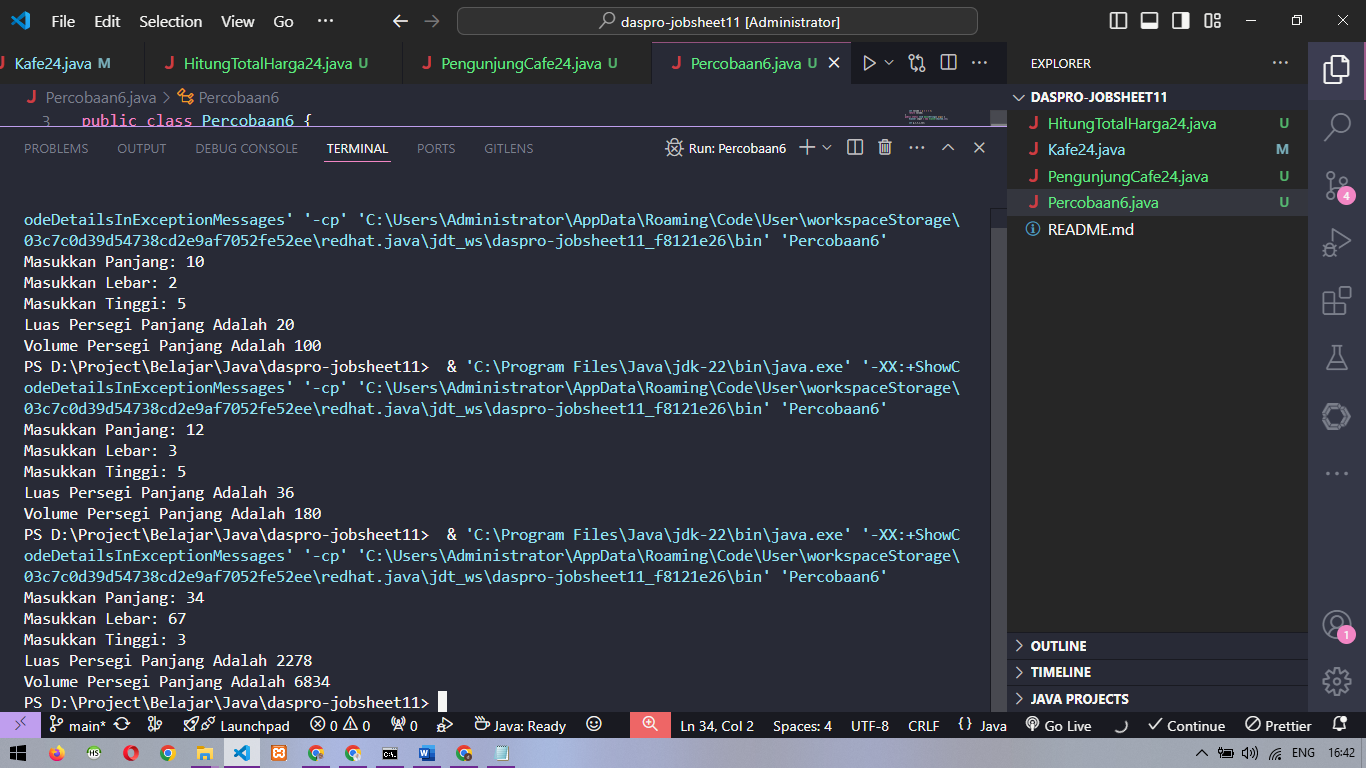
1. Buat fungsi untuk menghitung luas dan volume



1. Perubahan kode main



1. Hasil run



**Pertanyaan**

1. Menginputkan Panjang, lebar dan tinggi lalu memanggil fungsi hitungLuas dan hitungVolume dengan parameter yang diinputkan user lalu di print hasilnya
2. Outputnya 1 2 3 4 5 6 7

Urutan system nya yaitu

1. int temp = Jumlah(1, 1) -> hasil nya 2
2. TampilJumlah(temp, 5)
3. Di dalam tampilJumlah memanggil fungsi tampilHinggaKeI(Jumlah(2, 5)
4. Berarti tampilHinggaKeI(7)
5. Di dalam sana melakukan perulangan sebanyak parameter I, lalu di print i nya
6. A. Fungsi harus menggunakan parameter jika Fungsi membutuhkan data dari luar untuk bekerja dan Fungsi perlu beradaptasi untuk berbagai scenario

B. Fungsi tidak perlu parameter jika Data yang digunakan sudah tersedia di dalam fungsi dan Fungsi tidak memerlukan variasi input.

C. Fungsi harus memiliki nilai kembalian (return) jika Fungsi menghasilkan data atau hasil yang perlu digunakan kembali. Dan Fungsi harus memberikan status, jawaban, atau validasi.

D. Fungsi tidak perlu nilai kembalian jika Fungsi hanya melakukan tindakan tanpa menghasilkan hasil yang perlu digunakan lebih lanjut.