

Nama : Akbar Bagus Wicaksana

Nim : 254107020067

Kelas : TI – 1H

Matkul : Praktikum Daspro

Pertemuan ke 11

## Percobaan 1



```
1 public static void Menu() {
2     System.out.println("==== MENU RESTO KAFE =====");
3     System.out.println("1. Kopi Hitam - Rp 15,000");
4     System.out.println("2. Cappuccino - Rp 20,000");
5     System.out.println("3. Latte - Rp 22,000");
6     System.out.println("4. Teh Tarik - Rp 12,000");
7     System.out.println("5. Roti Bakar - Rp 10,000");
8     System.out.println("6. Mie Goreng - Rp 18,000");
9     System.out.println("=====");
10    System.out.println("Silahkan pilih menu yang Anda inginkan.");
11 }
12
13 public static void main(String[] args) {
14     Menu();
15 }
```

## Pertanyaan!

1. Apakah fungsi tanpa parameter selalu harus bertipe void?
  - Jawaban: Tidak, Karena parameter dan tipe kembalian adalah dua konsep yang berbeda dan tidak saling mempengaruhi.
2. Apakah daftar menu pada program kafe dapat ditampilkan tanpa menggunakan fungsi Menu()? Modifikasi kode program tersebut untuk dapat menampilkan daftar menu tanpa menggunakan fungsi!
  - Jawaban: Ya, daftar menu bisa ditampilkan tanpa menggunakan fungsi Menu().

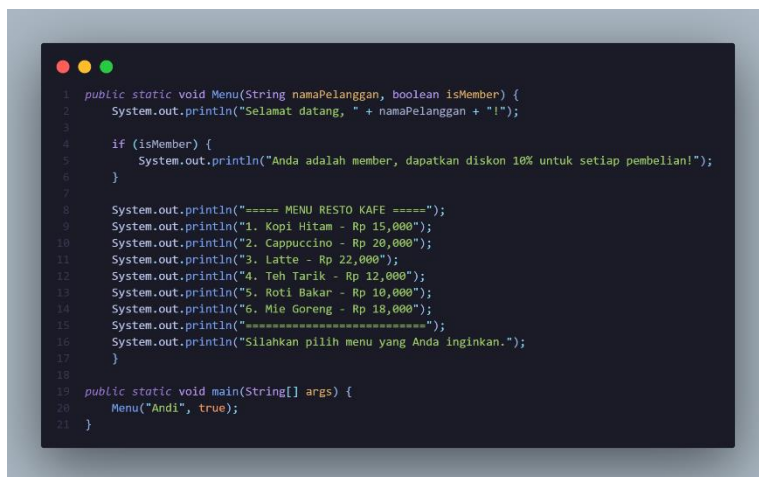


```
1 public static void main(String[] args) {
2     System.out.println("==== MENU RESTO KAFE =====");
3     System.out.println("1. Kopi Hitam - Rp 15,000");
4     System.out.println("2. Cappuccino - Rp 20,000");
5     System.out.println("3. Latte - Rp 22,000");
6     System.out.println("4. Teh Tarik - Rp 12,000");
7     System.out.println("5. Roti Bakar - Rp 10,000");
8     System.out.println("6. Mie Goreng - Rp 18,000");
9     System.out.println("=====");
10    System.out.println("Silahkan pilih menu yang Anda inginkan.");
11 }
```

3. Jelaskan keuntungan menggunakan fungsi Menu() dibandingkan menulis semua perintah penampilan menu langsung di dalam fungsi main.

- Jawaban: Menggunakan fungsi Menu() membuat program lebih rapi, mencegah penulisan ulang kode, memudahkan perubahan menu, dan membuat program lebih terstruktur sehingga mudah dipahami dan diperbaiki.
4. Uraikan secara singkat alur eksekusi program ketika fungsi Menu() dipanggil dari main (mulai dari program dijalankan sampai daftar menu tampil di layar).
- Jawaban: Saat program dijalankan, eksekusi dimulai dari fungsi main(). Di dalam main, program menemukan pemanggilan Menu(), lalu alur eksekusi berpindah ke fungsi Menu(). Di dalam fungsi Menu(), program menjalankan semua perintah System.out.println() untuk menampilkan daftar menu ke layar. Setelah semua perintah selesai dijalankan, eksekusi kembali ke fungsi main dan program berlanjut atau selesai.

## Percobaan 2



```

1 public static void Menu(String namaPelanggan, boolean isMember) {
2     System.out.println("Selamat datang, " + namaPelanggan + "!");
3
4     if (isMember) {
5         System.out.println("Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!");
6     }
7
8     System.out.println("==== MENU RESTO KAFE =====");
9     System.out.println("1. Kopi Hitam - Rp 15,000");
10    System.out.println("2. Cappuccino - Rp 20,000");
11    System.out.println("3. Latte - Rp 22,000");
12    System.out.println("4. Teh Tarik - Rp 12,000");
13    System.out.println("5. Roti Bakar - Rp 10,000");
14    System.out.println("6. Mie Goreng - Rp 18,000");
15    System.out.println("=====");
16    System.out.println("Silahkan pilih menu yang Anda inginkan.");
17 }
18
19 public static void main(String[] args) {
20     Menu("Andi", true);
21 }

```

## Pertanyaan!

1. Apakah kegunaan parameter di dalam fungsi?
  - Jawaban: Parameter dalam fungsi digunakan sebagai tempat menerima data dari luar fungsi sehingga fungsi dapat bekerja berdasarkan nilai yang diberikan. Parameter membuat fungsi menjadi lebih fleksibel, karena satu fungsi bisa menghasilkan output atau perilaku berbeda tergantung data yang dikirim saat dipanggil.
2. Jelaskan mengapa pada percobaan ini fungsi Menu() menggunakan parameter namaPelanggan dan isMember?
  - Jawaban: karena fungsi tersebut perlu menampilkan menu berdasarkan informasi pelanggan yang berbeda-beda.
3. Apakah parameter sama dengan variabel? Jelaskan.
  - Jawaban: Parameter bukan sama persis dengan variabel, tetapi parameter adalah variabel khusus yang digunakan untuk menerima nilai saat fungsi dipanggil. semua parameter adalah variabel, tetapi tidak semua variabel adalah parameter. Parameter hanya ada pada fungsi, sedangkan variabel bisa ada di seluruh bagian program.
4. Jelaskan bagaimana cara kerja parameter isMember pada fungsi Menu(). Apa perbedaan output ketika isMember bernilai true dan ketika false?

- Jawaban: Parameter **isMember** bekerja sebagai penentu apakah pelanggan adalah member atau bukan. Nilai ini dikirim saat fungsi **Menu()** dipanggil. Jika bernilai **true**, muncul pesan diskon; jika **false**, pesan tersebut tidak muncul.
5. Apa yang akan terjadi jika memanggil fungsi Menu() tanpa menyertakan parameter namaPelanggan dan isMember?
    - Jawaban: Jika memanggil fungsi Menu() tanpa menyertakan parameter namaPelanggan dan isMember, maka program akan error.
  6. Modifikasi kode di atas dengan menambahkan parameter baru kodePromo (String). Jika kodePromo adalah "DISKON50", tampilkan berikan diskon 50%. Jika kodePromo adalah "DISKON30", tampilkan berikan diskon 30%. Jika tidak ada kode promo yang berlaku, tampilkan kode invalid.

```

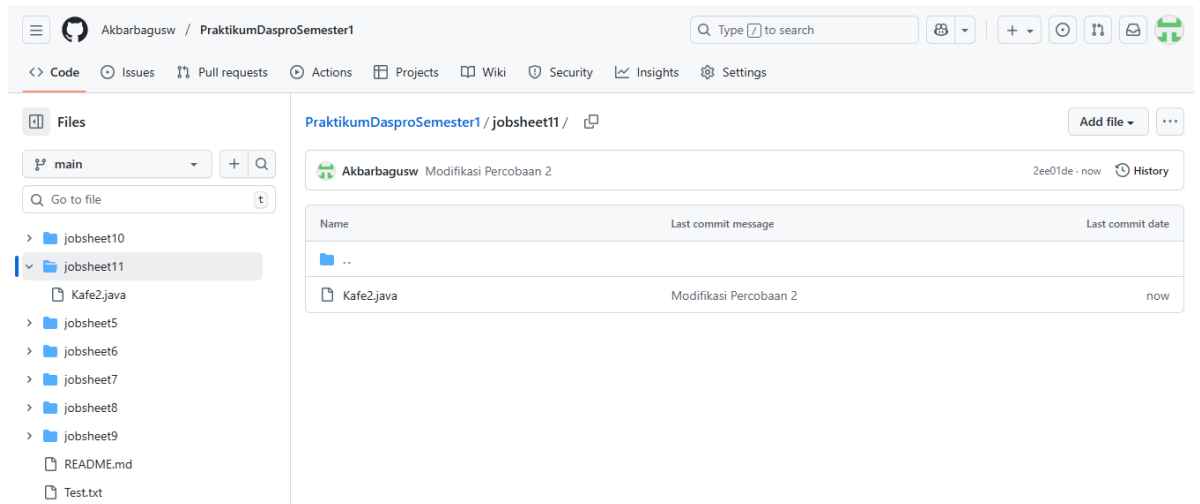
1  public static void Menu(String namaPelanggan, boolean isMember, String kodePromo) {
2      System.out.println("Selamat datang, " + namaPelanggan + "!");
3
4      if (isMember) {
5          System.out.println("Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!");
6      }
7
8      if (kodePromo.equals("DISKON50")){
9          System.out.println("Kode promo valid! Anda mendapatkan diskon 50%.");
10     } else if (kodePromo.equals("DISKON30")) {
11         System.out.println("Kode promo valid! Anda mendapatkan diskon 30%.");
12     } else {
13         System.out.println("Kode promo invalid.");
14     }
15
16     System.out.println("==== MENU RESTO KAFE =====");
17     System.out.println("1. Kopi Hitam - Rp 15,000");
18     System.out.println("2. Cappuccino - Rp 20,000");
19     System.out.println("3. Latte - Rp 22,000");
20     System.out.println("4. Teh Tarik - Rp 12,000");
21     System.out.println("5. Roti Bakar - Rp 10,000");
22     System.out.println("6. Mie Goreng - Rp 18,000");
23     System.out.println("=====");
24     System.out.println("Silahkan pilih menu yang Anda inginkan.");
25 }
26
27 public static void main(String[] args) {
28     Menu("Andi", true, "DISKON50");
29 }

```

7. Berdasarkan fungsi Menu() di atas, jika nama pelanggan adalah "Budi", pelanggan tersebut member, dan menggunakan kode promo "DISKON30", tuliskan satu baris perintah pemanggilan fungsi menu yang benar.
  - Jawaban: Menu("Budi", true, "DISKON30");
8. Menurut Anda, apakah penggunaan parameter namaPelanggan dan isMember pada fungsi Menu() membuat program lebih mudah dibaca dan dikembangkan dibandingkan jika nilai-nilai tersebut ditulis langsung di dalam fungsi tanpa parameter? Jelaskan alasan Anda.
  - Jawaban: Ya, penggunaan parameter namaPelanggan dan isMember membuat program lebih mudah dibaca dan dikembangkan. Dengan parameter, fungsi dapat menerima nilai dari luar sehingga tidak terikat pada data tertentu. Akibatnya,

fungsi bisa digunakan untuk banyak pelanggan dan berbagai kondisi tanpa perlu mengubah isi fungsi. Hal ini membuat kode lebih fleksibel, lebih rapi, dan memudahkan pengembangan program di masa depan.

## 9. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 2”



## Percobaan 3

```
1 public static void Menu(String namaPelanggan, boolean isMember, String kodePromo) {
2     System.out.println("Selamat datang, " + namaPelanggan + "!");
3
4     if (isMember) {
5         System.out.println("Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!");
6     }
7
8     if (kodePromo.equals("DISKON50")){
9         System.out.println("Kode promo valid! Anda mendapatkan diskon 50%.");
10    } else if (kodePromo.equals("DISKON30")) {
11        System.out.println("Kode promo valid! Anda mendapatkan diskon 30%.");
12    } else {
13        System.out.println("Kode promo invalid.");
14    }
15
16    System.out.println("==== MENU RESTO KAFE =====");
17    System.out.println("1. Kopi Hitam - Rp 15,000");
18    System.out.println("2. Cappuccino - Rp 20,000");
19    System.out.println("3. Latte - Rp 22,000");
20    System.out.println("4. Teh Tarik - Rp 12,000");
21    System.out.println("5. Roti Bakar - Rp 10,000");
22    System.out.println("6. Mie Goreng - Rp 18,000");
23    System.out.println("=====");
24    System.out.println("Silahkan pilih menu yang Anda inginkan.");
25 }
26
27 public static int hitungTotalHarga(int pilihanMenu, int banyakItem){
28     int[] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};
29
30     int hargaTotal = hargaItems[pilihanMenu - 1] * banyakItem;
31     return hargaTotal;
32 }
33
34 public static void main(String[] args) {
35     Scanner sc = new Scanner(System.in);
36     Menu("Budi", true, "DISKON30");
37     System.out.println("\nMasukkan nomor menu yang ingin Anda pesan: ");
38     int pilihanMenu = sc.nextInt();
39     System.out.println("Masukkan jumlah item yang ingin dipesan: ");
40     int banyakItem = sc.nextInt();
41
42     int totalHarga = hitungTotalHarga(pilihanMenu, banyakItem);
43
44     System.out.println("Total harga untuk pesanan Anda: Rp"+totalHarga);
45 }
```

## Pertanyaan!

1. Jelaskan secara singkat kapan suatu fungsi membutuhkan nilai kembalian (return value) dan kapan fungsi tidak perlu mengembalikan nilai. Berikan minimal satu contoh dari program kafe pada Percobaan 3 untuk masing-masing kasus.
  - Jawaban: Suatu fungsi membutuhkan nilai kembalian (return value) ketika fungsi tersebut harus menghasilkan suatu data atau hasil perhitungan yang masih akan digunakan oleh bagian program lainnya. Sebaliknya, fungsi tidak perlu mengembalikan nilai jika tugasnya hanya menampilkan informasi atau melakukan aksi tanpa harus memberikan hasil kembali ke pemanggil fungsi.
2. Fungsi `hitungTotalHargaNoPresensi` saat ini mengembalikan total harga berdasarkan `pilihanMenu` dan `jumlahPesanan`. Sebutkan tipe data nilai kembalian dan dua buah parameter yang digunakan fungsi tersebut. Jelaskan arti masing-masing parameter dalam konteks program kafe.
  - Jawaban: Fungsi `hitungTotalHarga` mengembalikan nilai bertipe `int`, yaitu total harga pesanan. Fungsi ini menggunakan dua parameter: `pilihanMenu`, yang menunjukkan nomor menu yang dipilih pelanggan, dan `banyakItem`, yang menunjukkan jumlah pesanan. Kedua parameter tersebut digunakan untuk menghitung total harga berdasarkan menu yang dipilih dan jumlah item yang dipesan.
3. Modifikasi kode di atas sehingga fungsi `hitungTotalHargaNoPresensi` dapat menerima `kodePromo`. Jika `kodePromo` adalah "DISKON50", maka mendapat diskon 50% dari `totalHarga` dan tampilkan diskon. Jika `kodePromo` adalah "DISKON30", maka mendapat diskon 30% dari `totalHarga` dan tampilkan diskon. Jika tidak ada kode promo yang berlaku, tampilkan kode invalid dan tidak ada pengurangan total harga `totalHarga`.
  - Jawaban:

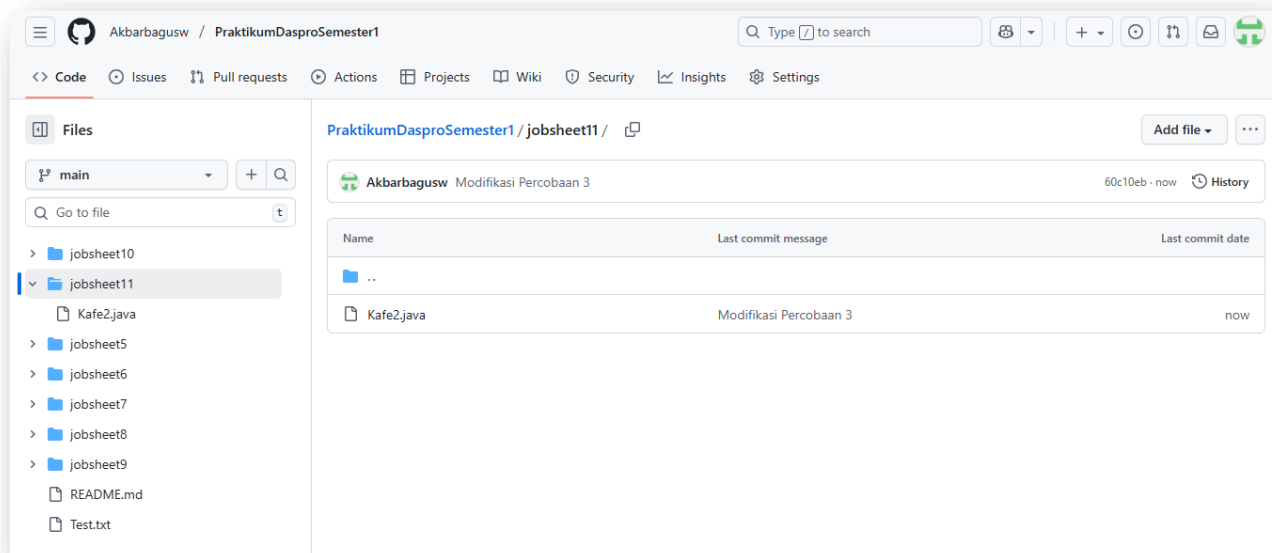
```
1  public static int hitungTotalHarga2(int pilihanMenu, int banyakItem, String kodePromo) {
2      int[] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};
3
4      int hargaTotal = hargaItems[pilihanMenu - 1] * banyakItem;
5      int diskon = 0;
6
7      if (kodePromo.equals("DISKON50")) {
8          diskon = hargaTotal * 50 / 100;
9          System.out.println("Kode promo valid! Diskon 50%: Rp" + diskon);
10     } else if (kodePromo.equals("DISKON30")) {
11         diskon = hargaTotal * 30 / 100;
12         System.out.println("Kode promo valid! Diskon 30%: Rp" + diskon);
13     } else {
14         System.out.println("Kode promo invalid. Tidak ada diskon.");
15     }
16
17     return hargaTotal - diskon;
18 }
19
20 public static void main(String[] args) {
21     Scanner sc = new Scanner(System.in);
22     Menu("Budi", true, "DISKON50");
23     System.out.println("\nMasukkan nomor menu yang ingin Anda pesan: ");
24     int pilihanMenu = sc.nextInt();
25     System.out.println("Masukkan jumlah item yang ingin dipesan: ");
26     int banyakItem = sc.nextInt();
27
28     int totalHarga = hitungTotalHarga2(pilihanMenu, banyakItem, "DISKON50");
29
30     System.out.println("Total harga untuk pesanan Anda: Rp"+totalHarga);
31 }
32 }
```

4. Modifikasi kode di atas sehingga bisa memilih beberapa jenis menu berbeda serta menampilkan total keseluruhan pesanan. Bagaimana memodifikasi program sehingga pengguna dapat: memesan lebih dari satu jenis menu (misalnya menu 1 dan 3 sekaligus), dan menampilkan total keseluruhan pesanan (gabungan dari semua jenis menu)?

- Jawaban:

```
1 public static void main(String[] args) {
2     Scanner sc = new Scanner(System.in);
3     Menu("Budi", true, "DISKON50");
4
5     int totalKeseleruhan = 0;
6     int pilihanMenu;
7
8     while (true) {
9         System.out.println("\nMasukkan nomor menu yang ingin Anda pesan (0 untuk keluar): ");
10        pilihanMenu = sc.nextInt();
11
12        if (pilihanMenu == 0) break;
13
14        System.out.println("Masukkan jumlah item yang ingin dipesan: ");
15        int banyakItem = sc.nextInt();
16
17        int totalPerMenu = hitungTotalHarga2(pilihanMenu, banyakItem, "DISKON50");
18        totalKeseleruhan += totalPerMenu;
19        System.out.println("Total harga untuk menu ini: Rp" + totalPerMenu);
20    }
21    System.out.println("Total harga untuk pesanan Anda: Rp"+totalKeseleruhan);
22 }
```

5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 3”



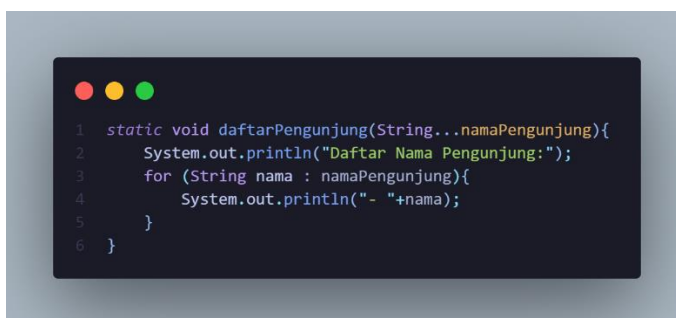


## Percobaan 4



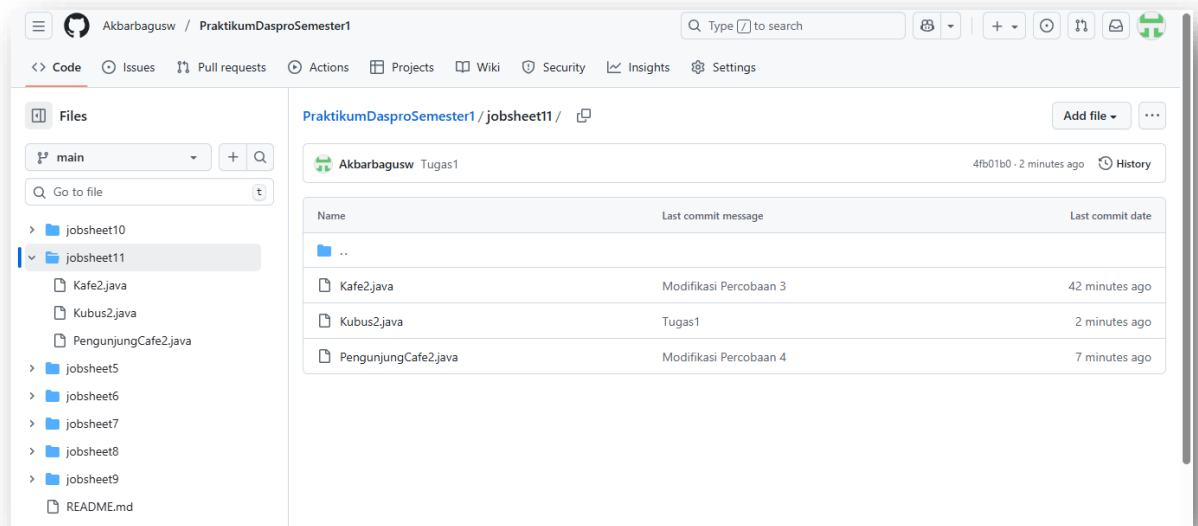
### Pertanyaan!

1. Jelaskan mengapa penulisan parameter di praktikum 4 ditulis dengan `String...namaPengunjung`!
  - Jawaban: karena fungsi tersebut harus dapat menerima jumlah argumen yang tidak tetap (variadic parameter). `String...` memungkinkan pengguna memberikan bebas berapa pun jumlah nama, tanpa perlu membuat array secara manual.
2. Modifikasi method `daftarPengunjung` menggunakan `for-each` loop.
  - Jawaban:



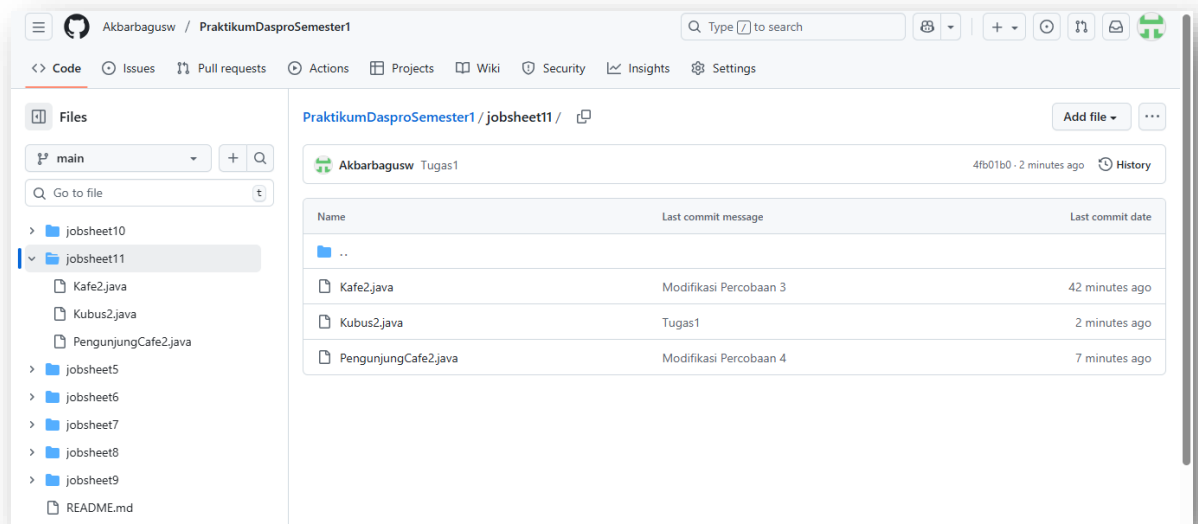
3. Bisakah menggunakan dua tipe data varargs dalam satu fungsi? Jelaskan jawaban Anda berdasarkan aturan varargs di Java, dan berikan contohnya!
  - Jawaban: Tidak bisa, dalam aturan varargs menyatakan bahwa Hanya boleh ada satu parameter varargs dalam satu method, dan Hanya boleh ada satu parameter varargs dalam satu method. Contoh yang benar adalah **`void contoh(String... nama)`**
4. Jelaskan apa yang terjadi jika fungsi `daftarPengunjung` dipanggil tanpa argumen. Apakah program akan error saat kompilasi, error saat dijalankan, atau tetap berjalan? Jika tetap berjalan, bagaimana output yang dihasilkan?
  - Jawaban: Jika fungsi `daftarPengunjung` dipanggil tanpa argumen, program tetap berjalan dan tidak akan error, baik saat kompilasi maupun saat dijalankan. Hasil outputnya adalah **Daftar Nama Pengunjung:** (tidak ada nama yang muncul)

5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 4”



## Tugas!

1. Buatlah sebuah class KubusNoPresensi yang di dalamnya terdapat fungsi untuk menghitung volume kubus dan luar permukaan kubus! Commit dan push program Anda ke Github dengan pesan “Tugas1”

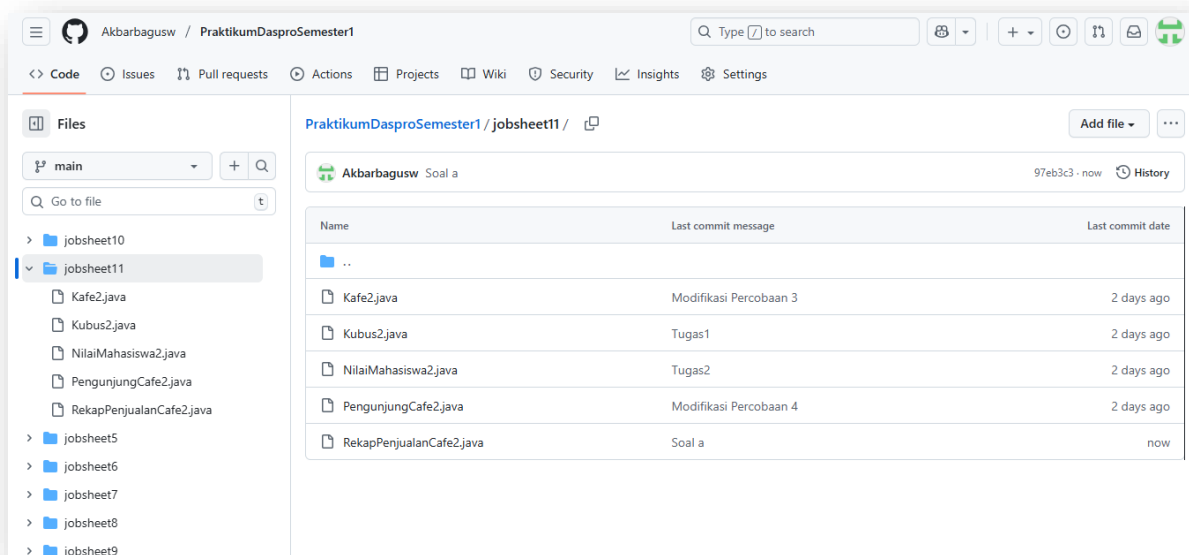


2. Di suatu kelas, terdapat N mahasiswa yang mengikuti satu mata kuliah. Nilai tiap mahasiswa untuk mata kuliah tersebut akan diinputkan oleh pengguna dan disimpan dalam sebuah array 1 dimensi berukuran N. Buatlah sebuah class NilaiMahasiswaNoPresensi yang di dalamnya terdapat beberapa fungsi berikut:
  - a) Fungsi isianArray untuk:
    - Mengisi elemen-elemen array dengan nilai yang diinputkan oleh user (jumlah elemen = N, dibaca dari input)



- b) Fungsi tampilArray untuk:
  - Menerima parameter berupa array nilai mahasiswa
  - Menampilkan seluruh nilai yang telah diinputkan
- c) Fungsi hitTot untuk:
  - Menerima parameter berupa array nilai mahasiswa
  - Menghitung dan mengembalikan total nilai seluruh mahasiswa (return tipe int atau double)
- d) Fungsi main untuk:
  - Membaca nilai N (jumlah mahasiswa) dari input pengguna
  - Membuat array dengan ukuran N
  - Memanggil fungsi isianArray, tampilArray, dan hitTot
  - Menampilkan total nilai yang dikembalikan oleh fungsi hitTot

Commit dan push program Anda ke Github dengan pesan “Tugas2”



3. Ibu Mariana mengelola sebuah kafe. Berikut adalah rekap penjualan 5 menu dari hari pertama hingga ketujuh:
  - a. Buatlah sebuah class RekapPenjualanCafeNoPresensi yang digunakan untuk menyimpan dan mengolah data penjualan 5 menu kafe dari hari pertama hingga hari ketujuh. Di dalam class tersebut, buat dan gunakan beberapa fungsi (method) berikut:
    - Fungsi untuk menginputkan data penjualan setiap menu untuk setiap hari (petunjuk: menggunakan array 2 dimensi).
    - Fungsi untuk menampilkan seluruh data penjualan dari hari pertama hingga hari terakhir dalam bentuk tabel
    - Fungsi untuk menampilkan menu yang memiliki total penjualan tertinggi selama tujuh hari (tampilkan nama menu dan total penjualannya).
    - Fungsi untuk menampilkan rata-rata penjualan untuk setiap menu selama tujuh hari (tampilkan nama menu dan nilai rata-ratanya).

Gunakan fungsi main untuk:

- Memanggil fungsi input data penjualan,
  - Menampilkan seluruh data penjualan,
  - Menampilkan menu dengan penjualan tertinggi, dan
  - Menampilkan rata-rata penjualan untuk setiap menu.
- b. Modifikasi program pada 3 (a) sehingga jumlah hari penjualan dan jumlah menu dibaca dari input pengguna, lalu sesuaikan ukuran array dan proses perhitungan agar tetap berjalan dengan benar.

Commit dan push program Anda ke Github dengan pesan “Tugas3”

