NAMA : SUTOKO(23)

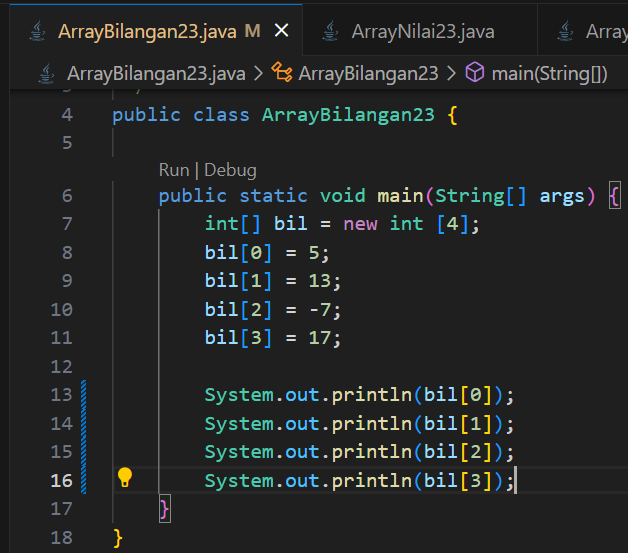
KELAS : SIB-1B

**ARRAY 1**

**LINK GITHUB SUTOKO**

https://github.com/sutoko395/daspro-jobsheet9.git

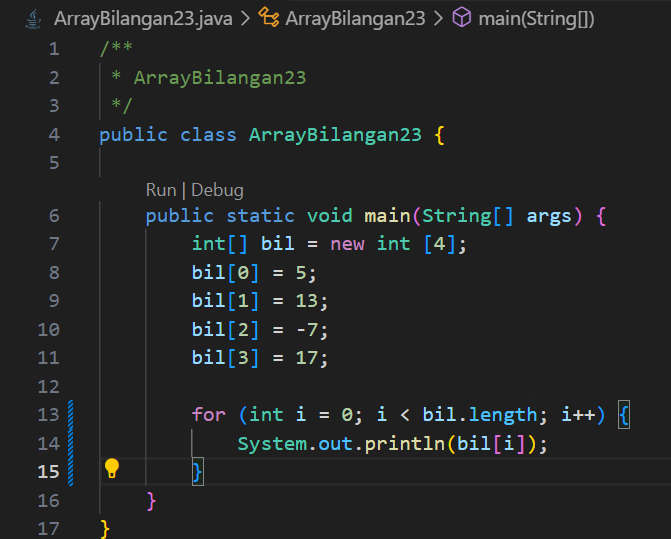
1. **Percobaan 1: Mengisi Elemen Array**

****

1. Jika isi masing-masing elemen array bil diubah dengan angka 5.0, 12867, 7.5, 2000000. Apa yang terjadi? Mengapa bisa demikian?

JAWABAN: Array bil dideklarasikan sebagai array dengan tipe data int. Dalam Java, tipe data int hanya dapat menyimpan nilai bilangan bulat. Sedangkan 5.0 atau 7.5 adalah tipe data double, maka akan terjadi kesalahan kompilasi (compilation error) karena double bukanlah int.

1. Modifikasi kode program di atas dengan melakukan inisialisasi elemen array sekaligus pada saat deklarasi array.



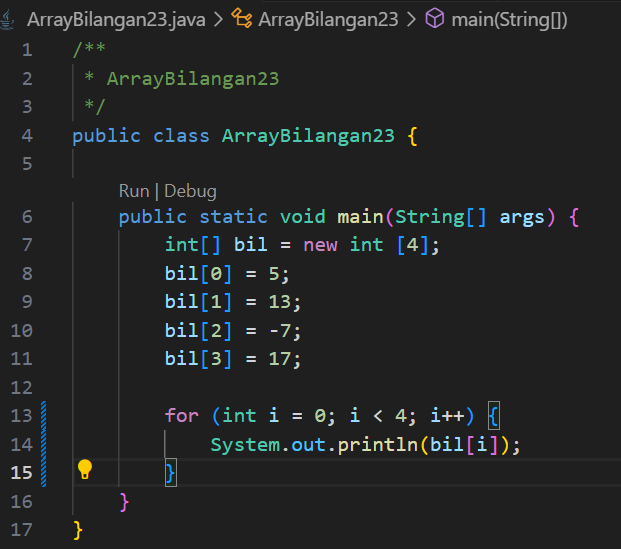
1. Ubah statement pada langkah No 4 menjadi seperti berikut

For(int i=0; i<4; i++) {

System.out.printIn(bil[i]);

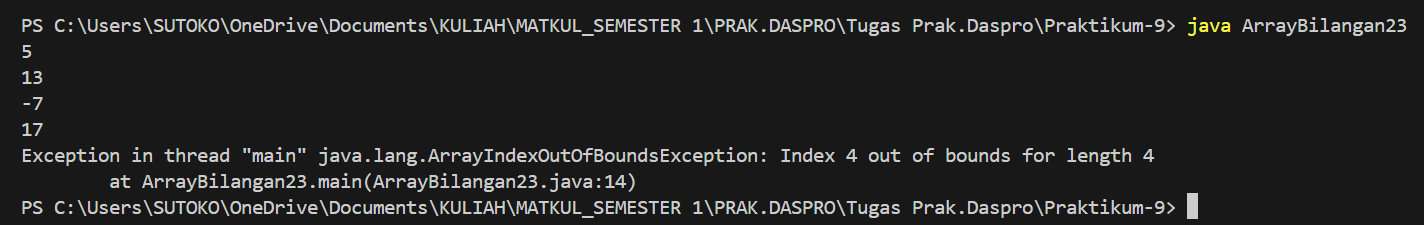
}

Apa keluaran dari program? Jelaskan maksud dari statement tersebut

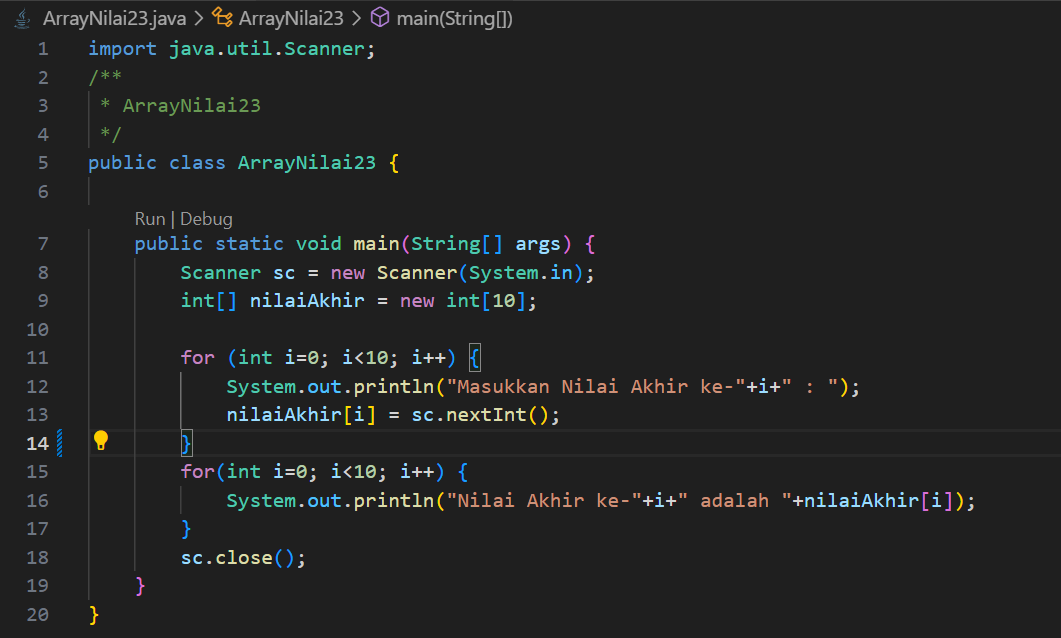
Outut dari program tetapsama seperti diawal yaitu mencetak angka 5, 13, -7, 17.

for loop ini akan mengiterasi dari i = 0 hingga i = 3 (karena i < 4), mencetak setiap elemen dari array bil. Loop ini mencetak nilai 5, 13, -7, dan 17.

1. Jika kondisi pada statement for-loop di atas diubah menjadi: i <= 4, apa keluaran dari program? Mengapa demikian?

Jika kondisinya diubah menjadi i <= 4, loop akan mencoba mengakses bil[4] pada iterasi terakhir.Ini akan menyebabkan ArrayIndexOutOfBoundsException karena array bil hanya memiliki indeks 0 hingga 3. Mengakses indeks 4 berada di luar batas (out of bounds).

1. Push dan commit kode program ke github.
2. **Percobaan 2: Studi Kasus Nilai Mahasiswa di SIAKAD**

****

1. Ubah statement pada langkah nomor 5 menjadi seperti berikut ini:

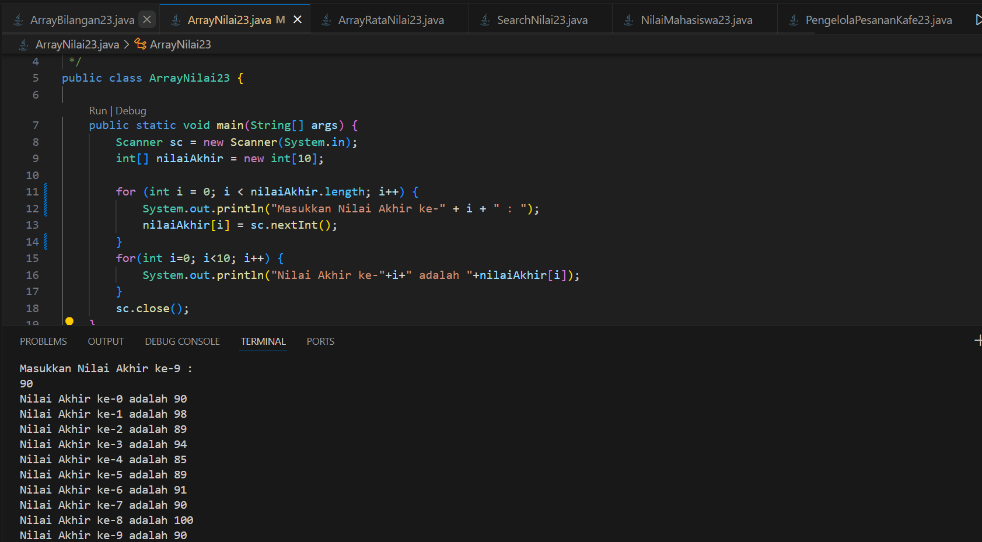
For(int i=0; i<nilaiAkhir.length; i++) {

System.out.printIn(“Masukkan Nilai Akhir ke-“+i+” : ”);

nilaiAkhir[i] = sc.nextInt();

}

Jalankan program. Apakah terjadi perubahan? Mengapa demikian?



JAWABAN: Tidak akan terjadi perubahan signifikan. Program akan tetap meminta pengguna memasukkan 10 nilai akhir seperti sebelumnya. Bedanya adalah nilaiAkhir.length digunakan untuk menentukan batas loop, yang merupakan praktik yang lebih baik karena jika ukuran array berubah, loop akan tetap berjalan dengan benar tanpa perlu mengubah batasan loop secara manual.

1. Apa yang dimaksud dengan kondisi: i < nilaiAkhir.length ?

JAWABAN: Kondisi ini memastikan loop berjalan selama nilai i lebih kecil dari panjang array nilaiAkhir. NilaiAkhir.length memberikan panjang array nilaiAkhir, jadi loop akan berjalan dari indeks 0 sampai indeks terakhir array. Ini adalah cara yang aman untuk menghindari kesalahan akses di luar batas array (out of bounds).

1. Ubah statement pada langkah nomor 6 menjadi seperti berikut ini, sehingga program hanya menampilkan nilai Mahasiswa yang lulus saja (yaitu mahasiswa yang memiliki nilai > 70):

For(int i=0; i<nilaiAkhir.length; i++) {

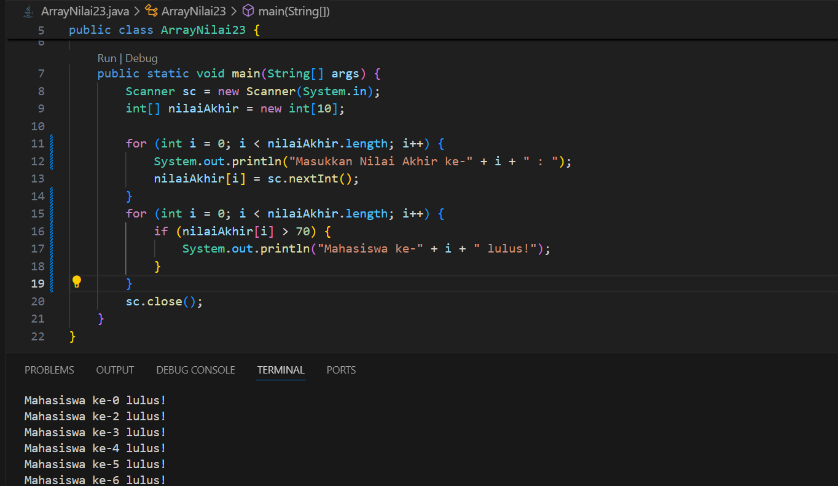
If(nilaiAkhir[i] > 70) {

System.out.printIn(“Mahasiswa ke-“+i+” lulus ! ”);

}

}

Jalankan program dan jelaskan alur program!

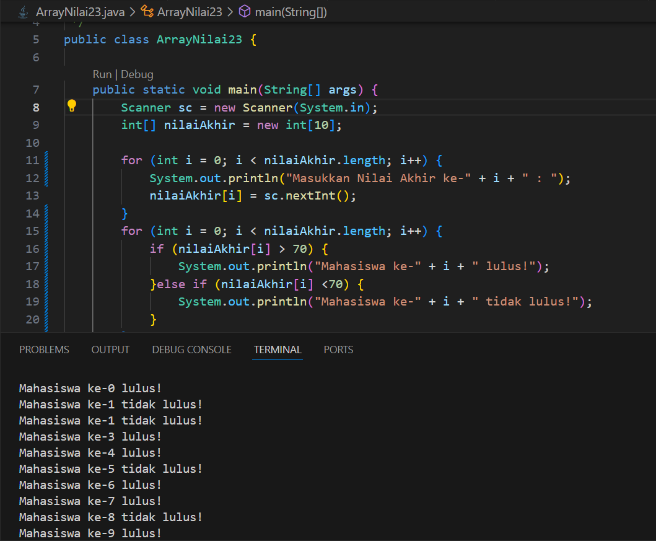


Program akan meminta pengguna memasukkan nilai akhir untuk 10 mahasiswa. Setelah semua nilai dimasukkan, program akan memeriksa setiap nilai dalam array. Jika nilai mahasiswa lebih besar dari 70, maka program akan mencetak pesan bahwa mahasiswa tersebut lulus. Dan jika nilai kurang dari 70 maka sistemtidak akan mencetak atau menampilkan output.

Alur program:

* Program meminta pengguna untuk memasukkan 10 nilai akhir.
* Program menyimpan nilai-nilai tersebut dalam array nilaiAkhir.
* Program memeriksa setiap nilai dalam array.
* Jika nilai lebih besar dari 70, program mencetak bahwa mahasiswa lulus.

1. Modifikasi program agar menampilkan status kelulusan semua mahasiswa berdasarkan nilai, yaitu dengan menampilkan status mana mahasiswa yang lulus dan tidak lulus, seperti ilustrasi output disoal:



Untuk menjadikan output seperti gambar disoal maka cukup tambahkan:

*}else if (nilaiAkhir[i] <70) {*

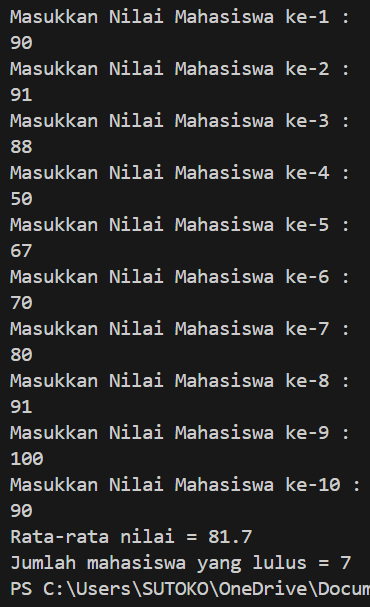
*System.out.println("Mahasiswa ke-" + i + " tidak lulus!");*

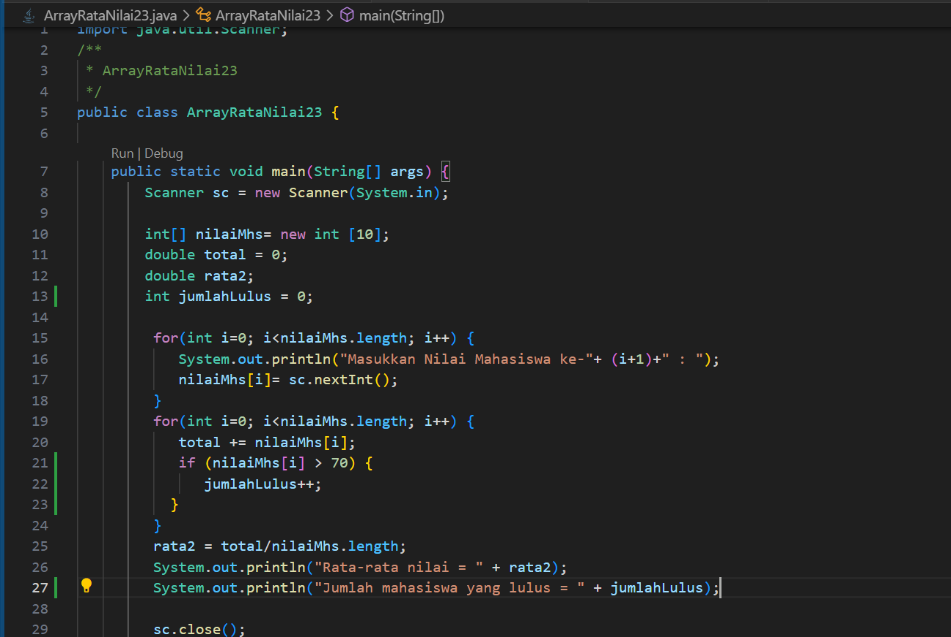
*}*

Tujuannya adalah agar nilai dibawah 70 dapat ditampilkan dan terdapat keterangan tidak lulus!

1. Push dan commit kode program ke github.
2. **Percobaan 3: Studi Kasus Nilai Mahasiswa di SIAKAD**

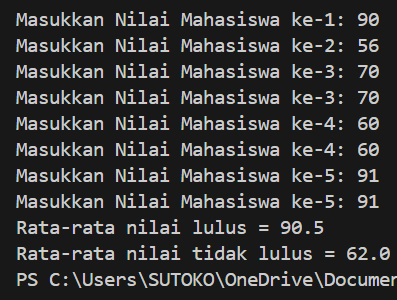
****

1. Modifikasi kode program pada praktikum percobaan 3 di atas (ArrayRataNilaiXX.java) agar program dapat menampilkan banyaknya mahasiswa yang lulus, yaitu mahasiswa yang memiliki lebih besar dari 70 (>70).

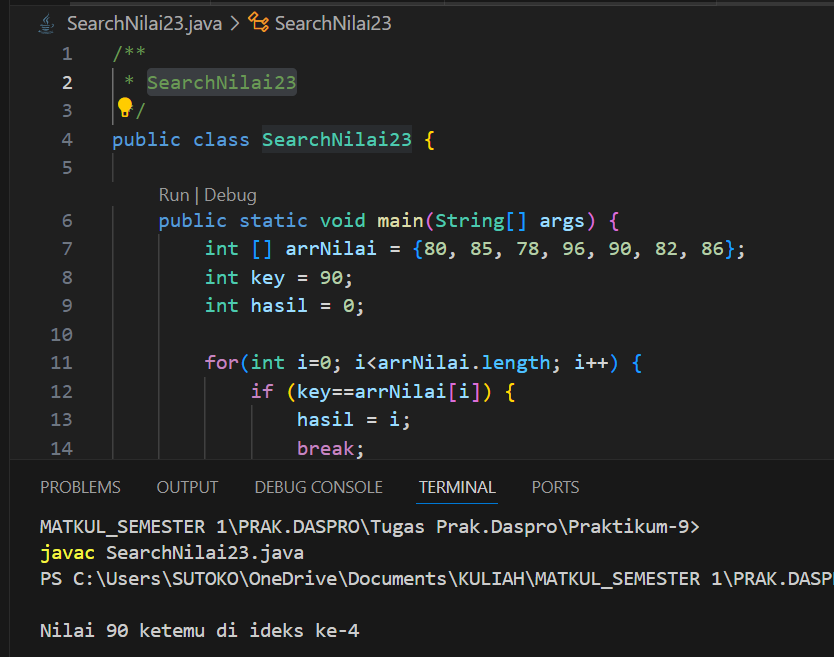


1. Modifikasi program pada praktikum percobaan 3 di atas (ArrayRataNilaiXX.java) sehingga program menerima jumlah elemen berdasarkan input dari pengguna dan mengeluarkan output seperti berikut ini:





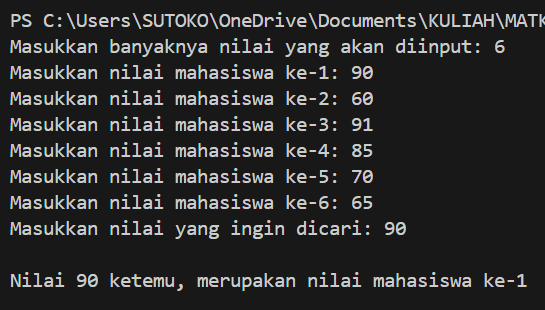
1. Push dan commit kode program ke github.
2. **Percobaan 4: Searching**

****

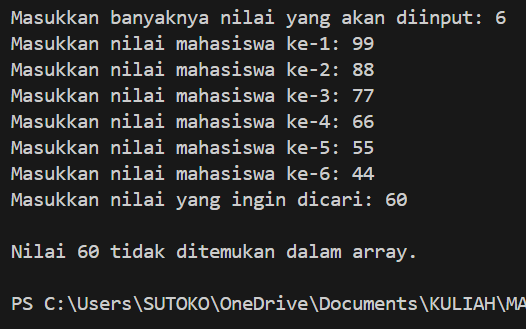
1. Jelaskan maksud dari statement break; pada baris ke-10 kode program percobaan 4 di atas.

JAWABAN: Perintah break; menghentikan loop segera setelah kondisi key == arrNilai[i] terpenuhi. Ini mencegah pemeriksaan yang tidak perlu setelah nilai yang dicari ditemukan, membuat program lebih efisien.

1. Modifikasi kode program pada percobaan 4 di atas sehingga program dapat menerima input berupa banyaknya elemen array nilai, isi array, dan sebuah nilai (key) yang ingin dicari. Lalu cetak ke layar indeks posisi elemen dari nilai (key) yang dicari. Contoh hasil program seperti disoal:



1. Modifikasi program pada percobaan 4 di atas, sehingga program akan memberikan pesan "Nilai yang dicari tidak ditemukan" jika nilai yang dicari (key) tidak ada di dalam array. Contoh tampilan program sebagai berikut(seperti disaol):



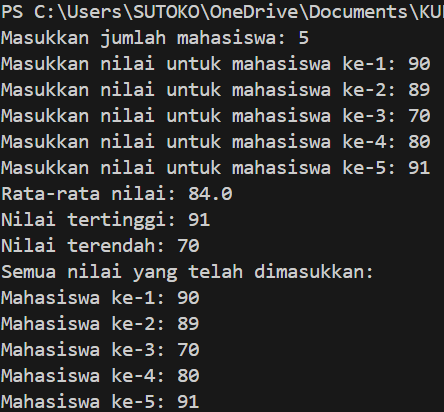
1. Push dan commit kode program ke github.
2. **Tugas**
3. Anda diminta untuk membuat program yang dapat menyimpan dan mengelola nilai mahasiswa. Nilai berupa bilangan bulat. Program harus menyediakan fitur untuk:

- memasukkan banyaknya nilai mahasiswa yang akan diinput,

- memasukkan setiap nilai mahasiswa, menghitung nilai rata-rata,

- menampilkan nilai tertinggi dan nilai terendah, serta

- menampilkan semua nilai yang telah dimasukkan.



1. Buat program yang dapat mengelola pemesanan makanan dan minuman di sebuah kafe. Program akan memungkinkan pengguna untuk memasukkan pesanan, menghitung total biaya pesanan, dan menampilkan daftar pesanan yang telah dibuat.

* Input:

-jumlah pesanan (input dari pengguna).

-nama makanan/minuman dan harga untuk masing-masing pesanan (input dari pengguna)

* Proses:

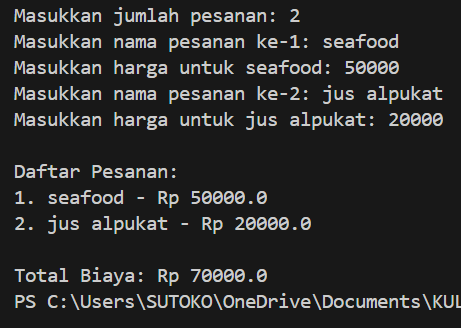
-simpan data pesanan dalam array satu dimensi untuk nama pesanan; dan array satu dimensi terpisah untuk harga.

-hitung total biaya dari semua pesanan yang dimasukkan.

-tampilkan daftar pesanan yang telah dimasukkan bersama dengan total biaya.

* Output:

-daftar pesanan dan total biaya dari semua pesanan.



1. Masih menggunakan kasus pada pemesanan makanan di kafe, buatlah program yang memungkinkan pengguna untuk memesan makanan dari menu yang tersedia di kafe. Program harus menyimpan daftar nama makanan dalam sebuah array dan memberikan opsi untuk mencari makanan yang diinginkan menggunakan metode linear search.

* Input:

- daftar menu makanan yang telah ditentukan sebelumnya dalam bentuk array. Nama-nama makanan telah di-inisialisasi saat deklarasi array. Misal: String[] menu = {"Nasi Goreng", "Mie Goreng", "Roti Bakar", "Kentang Goreng", "Teh Tarik", "Cappucino", "Chocolate Ice"};

-nama makanan yang ingin dicari (input dari pengguna).

* Proses:

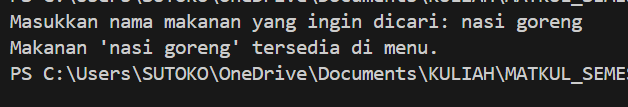
-program mencari nama makanan yang dimasukkan pengguna menggunakan algoritma linear search.

-jika makanan ditemukan, program akan menginformasikan pengguna bahwa makanan tersebut tersedia. Jika tidak ditemukan, program akan memberi tahu pengguna bahwa makanan yang dicari tidak ada di menu.

* Output:

-Tampilkan hasil pencarian kepada pengguna.





**LINK GITHUB SUTOKO**

https://github.com/sutoko395/daspro-jobsheet9.git