Nama : Angga Saputra Ramadhan

Kelas : TI\_1B

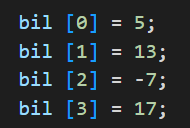
NIM : 244107020065

**2.1 Percobaan 1: Mengisi Elemen Array**

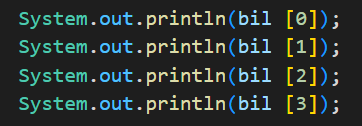
1. Buka text editor, buat class Java baru dengan nama ArrayBilanganXX.java. (XX=nomor absen)
2. Buat array bertipe integer dengan nama bil dengan kapasitas 4 elemen.



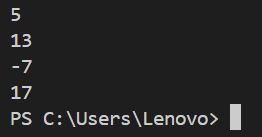
1. Isi masing-masing elemen array bil tadi dengan angka 5, 13, -7, 17.



1. Tampilkan ke layar semua isi elemennya:



1. Cocokkan dan amati hasilnya dengan gambar berikut ini:



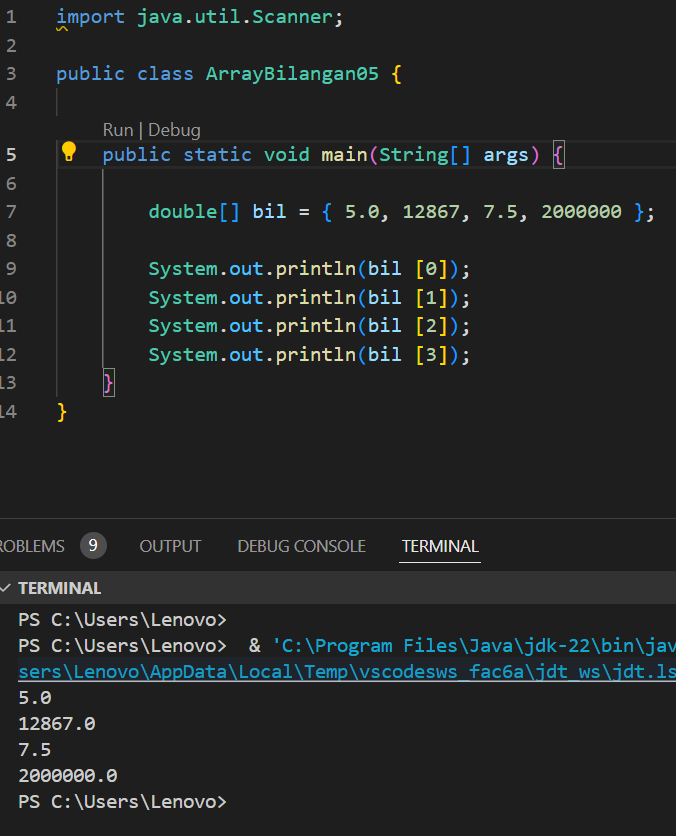
1. Push dan commit kode program ke github.

**Pertanyaan**

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

Jika mengubah isi masing-masing elemen array dengan angka decimal, java tidak akan bisa dijalankan

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 Apa keluaran dari program? Jelaskan maksud dari statement tersebut.

Output sama kode dengan sebelumnya, for digunakan agar bisa menghemat kode program

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

Akan memunculkan kesalahan ketidak dijalankan karena program mencoba mengakses elemen array ke-4 sedangkan elemen array hanya sampai no 3.

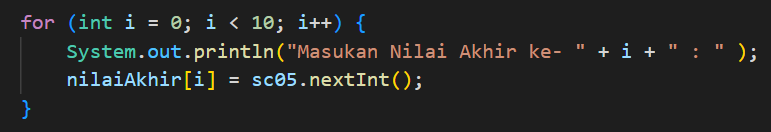
1. Push dan commit kode program ke github.

**Percobaan 2: Studi Kasus Nilai Mahasiswa di SIAKAD - Meminta Inputan Pengguna untuk Mengisi Elemen Array**

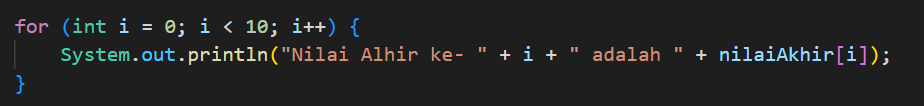
1. Buka text editor, buat file Java kemudian simpan dengan nama ArrayNilaiXX.java. (XX=nomor absen)
2. Buatlah struktur dasar java (membuat class dan method main).
3. Tambahkan import library Scanner.
4. Buat deklarasi array bertipe integer dengan nama nilaiAkhir dan berkapasitas 10 elemen seperti di bawah ini :



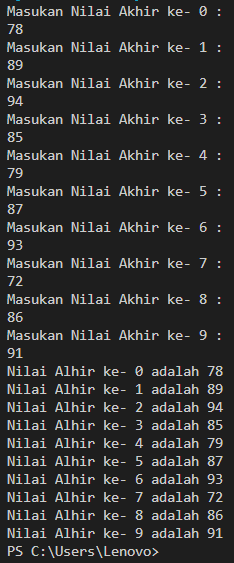
1. Buatlah struktur perulangan untuk menerima input dan mengisi elemen array nilaiAkhir, seperti berikut:



1. Menggunakan struktur perulangan, tampilkan semua isi elemen dari array nilaiAkhir, seperti berikut:



1. Jalankan program. Amati dan cocokkan dengan output berikut:



1. Push dan commit kode program ke github.

**Pertanyaan**

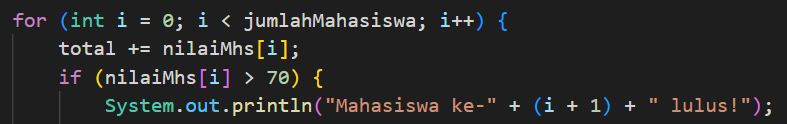
1. Ubah statement pada langkah nomor 5 menjadi seperti berikut ini: Jalankan program. Apakah terjadi perubahan? Mengapa demikian?

Tidak ada yang berubah dengan outputnya tetapi jika menggunakan nilaiAkhir.length jelas menunjukkan bahwa loop tersebut berjalan berdasarkan panjang array bukan angka tetap (10)

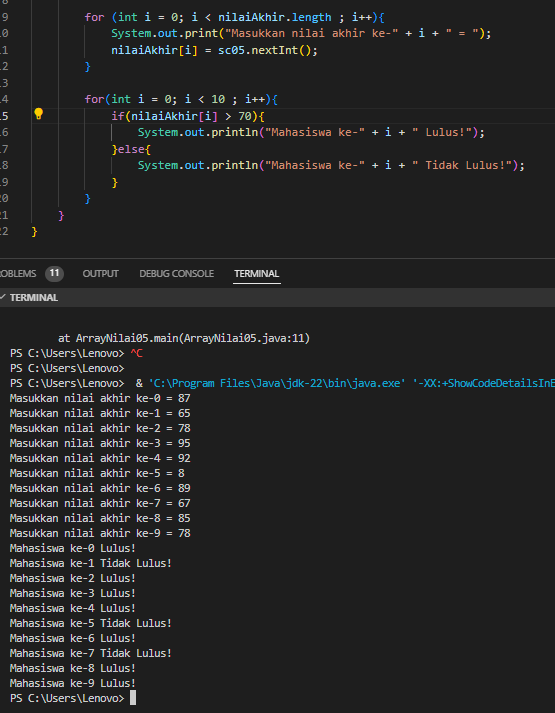
1. Apa yang dimaksud dengan kondisi: i < nilaiAkhir.length ?

Berarti loop akan terus berjalan selama nilai I lebih kecil dari pada panjang array nilaiAkhir.

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): Jalankan program dan jelaskan alur program!



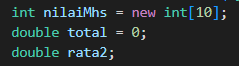
1. Modifikasi program agar menampilkan status kelulusan semua mahasiswa berdasarkan nilai, yaitu dengan menampilkan status mana mahasiswa yang lulus dan tidak lulus, seperti ilustrasi output berikut:



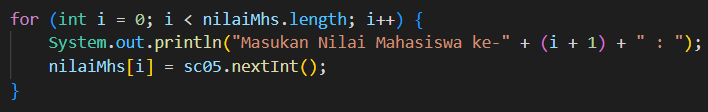
1. Push dan commit kode program ke github.

**2.3 Percobaan 3: Studi Kasus Nilai Mahasiswa di SIAKAD - Melakukan Operasi Aritmatika terhadap Elemen Array Waktu percobaan:**

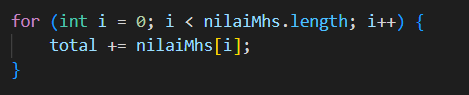
1. Buka text editor, buat file Java, kemudian simpan dengan nama ArrayRataNilaiXX.java. (XX = nomor absen).
2. Buatlah struktur dasar java (membuat class dan method main).
3. Import dan deklarasikan Scanner untuk keperluan input.
4. Buat array nilaiMhs bertipe integer dengan kapasitas 10. Kemudian deklarasikan variable total dan rata2 seperti gambar berikut ini:



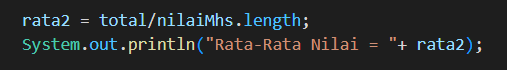
1. Isi array nilaiMhs dengan nilai dari input pengguna, sebagai berikut:



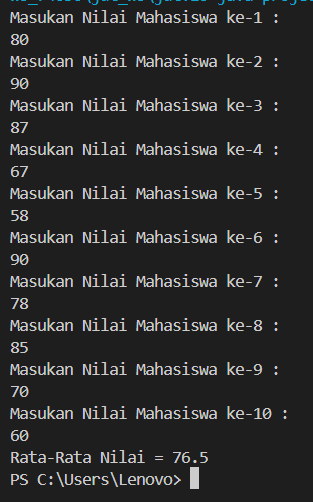
1. Gunakan perulangan untuk menghitung jumlah keseluruhan nilai dalam array nilaiMhs, sebagai berikut:



1. Kemudian hitung nilai rata-rata dengan cara nilai total dibagi jumlah elemen dari array nilaiMhs

:

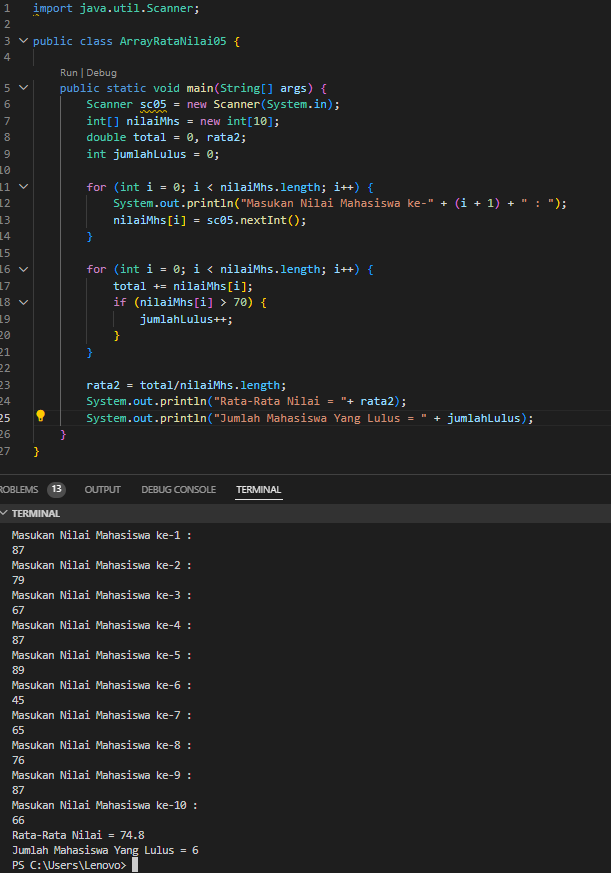
1. Amati hasilnya sebagai berikut:



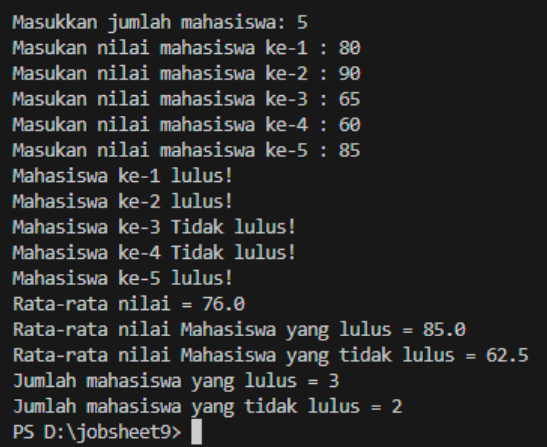
1. Push dan commit kode program ke github.

**Pertanyaan**

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).



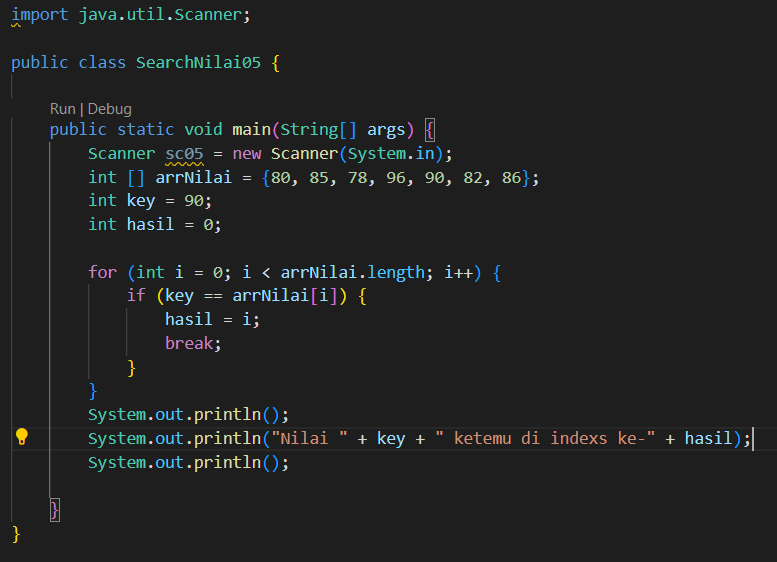
1. Modifikasi program pada praktikum percobaan 3 di atas (ArrayRataNilaiXX.java)sehingga program menerima jumlah elemen berdasarkan input dari pengguna dan mengeluarkan output seperti berikut ini:



1. Push dan commit kode program ke github.

**2.4 Percobaan 4:**

1. Buka text editor, buat file Java, kemudian simpan dengan nama SearchNilaiXX.java. (XX = nomor absen).
2. Tambahkan kode berikut ini:



1. Jalankan program tersebut. Amati apa yang terjadi. Cocokkan dengan output berikut:



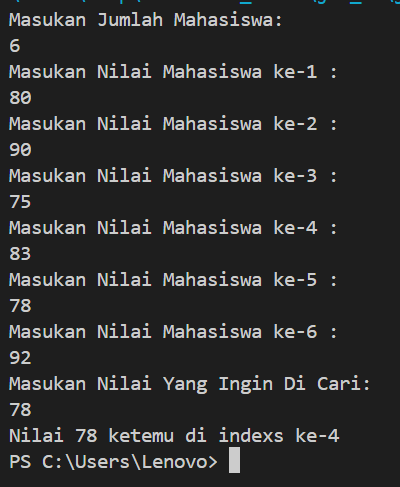
1. Push dan commit kode program ke github.

**Pertanyaan**

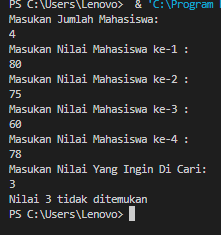
1. Jelaskan maksud dari statement break; pada baris ke-10 kode program percobaan 4 di atas.

Digunakan untuk menghentikan eksekusi dari loop atau switch-case yang sedang berjalan.

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:



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:



1. Push dan commit kode program ke github.

**Tugas**

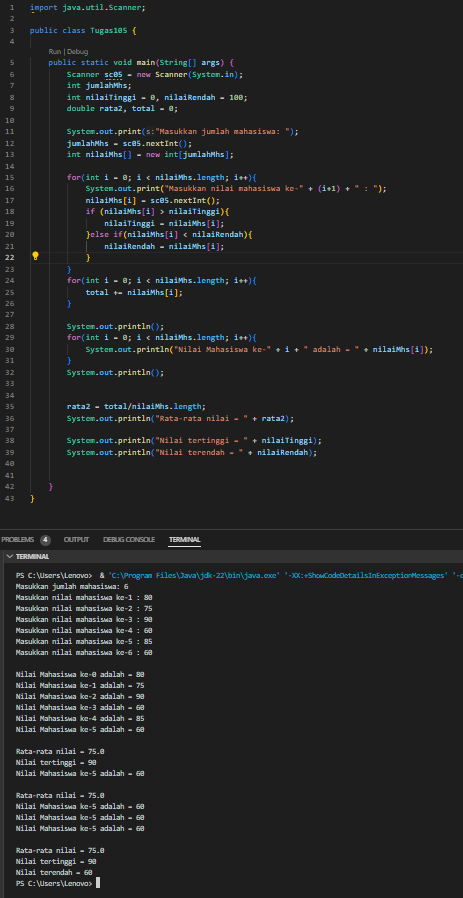
1. Anda diminta untuk membuat program yang dapat menyimpan dan mengelola nilai mahasiswa. Nilai berupa bilangan bulat. Program harus menyediakan fitur untuk:

- 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:

o jumlah pesanan (input dari pengguna).

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

* Proses:

o simpan data pesanan dalam array satu dimensi untuk nama pesanan; dan array satu dimensi

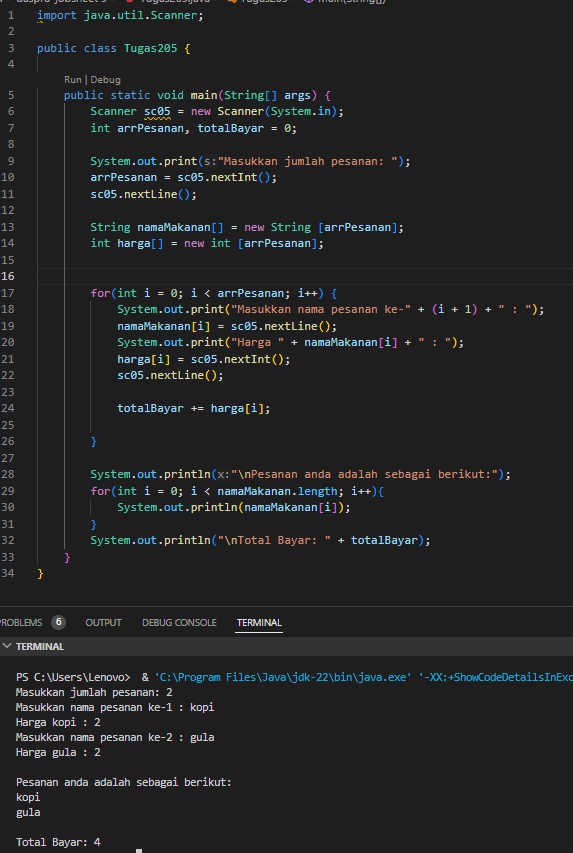
terpisah untuk harga.

o hitung total biaya dari semua pesanan yang dimasukkan.

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

* Output:

o 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:

o 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"};

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

- Proses:

o program mencari nama makanan yang dimasukkan pengguna menggunakan algoritma linear

search.

o 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:

o Tampilkan hasil pencarian kepada pengguna.

