# JOBSHEET 9

**Array 1**

# Tujuan

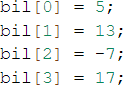
* Mahasiswa mampu memahami pembuatan Array 1 dimensi dan pengaksesan elemennya di Java
* Mahasiswa mampu membuat program dengan penggunaan Array satu dimensi

# Praktikum

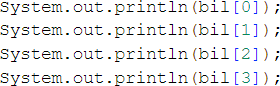
* 1. **Percobaan 1: Mengisi Elemen Array Waktu percobaan: 20 menit**
     1. Buka text editor, buat class Java baru dengan nama ArrayBilangan**XX**.java. (XX=nomor absen)
     2. Buat array bertipe integer dengan nama bil dengan kapasitas 4 elemen.



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



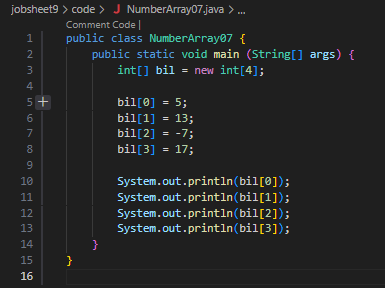
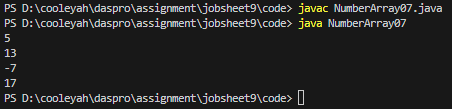
* + 1. Tampilkan ke layar semua isi elemennya:



* + 1. Cocokkan dan amati hasilnya dengan gambar berikut ini:



* + 1. Push dan commit kode program ke github.

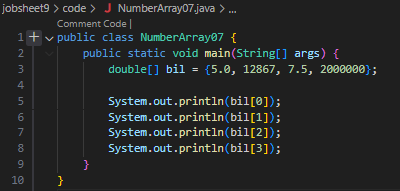
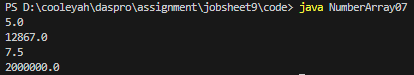


# Pertanyaan

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

* Jika bil diubah dengan angka 5.0, 12867, 7.5, 2000000, maka tipe datanya juga harus diubah dari int menjadi double. Karena, tipe data int digunakan untuk bilangan bulat (angka tanpa koma), sedangkan tipe data double digunakan untuk bilangan decimal (angka dengan koma).

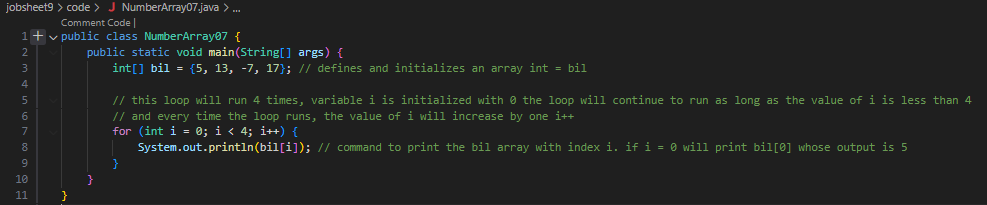
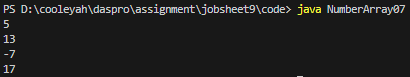
1. Modifikasi kode program di atas dengan melakukan inisialisasi elemen array sekaligus pada saat deklarasi array.

1. Ubah statement pada langkah No 4 menjadi seperti berikut

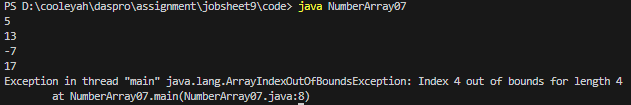


Apa keluaran dari program? Jelaskan maksud dari statement tersebut.



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

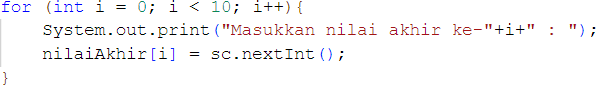
* then the program will generate an error because the valid indices in the array "bil" with length 4 are 0, 1, 2, and 3. When i reaches the value 4 (as required by the condition i <= 4), the program will try to access bil[4], which is outside the valid indices for the array "bil."



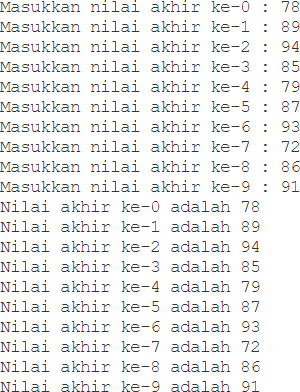
1. Push dan commit kode program ke github.

# Percobaan 2: Meminta Inputan Pengguna untuk Mengisi Elemen Array Waktu percobaan: 40 menit

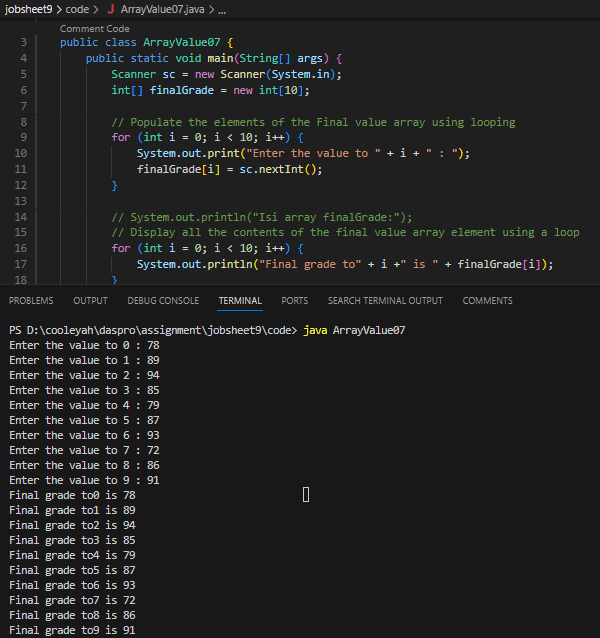
* + 1. Buka text editor, buat file Java kemudian simpan dengan nama ArrayNilai**XX**.java. (XX=nomor absen)
    2. Buatlah struktur dasar java (membuat class dan method main).
    3. Tambahkan import library Scanner.
    4. Buat deklarasi array bertipe integer dengan nama nilaiAkhir dan berkapasitas 10 elemen seperti di bawah ini :
    5. 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:
    2. Jalankan program. Amati dan cocokkan dengan output berikut:

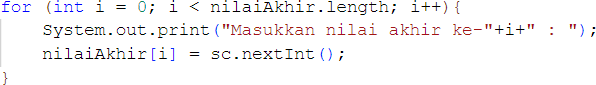


* + 1. Push dan commit kode program ke github



# Pertanyaan

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



Jalankan program. Apakah terjadi perubahan? Mengapa demikian?

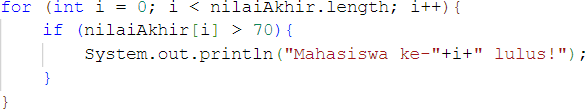
* 'finalGrade.length' this returns the length (number of elements) of the 'finalGrade' array or at 'int[] finalGrade = new int[10]'

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

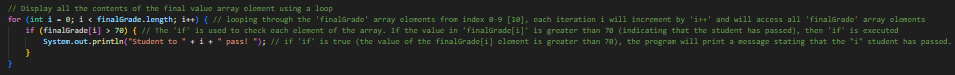
* 'i < finalGrade.length' means that the loop will continue as long as i is smaller than the length of the Endvalue array. This ensures that the loop will stop when i reaches or exceeds the length of the array, so all elements in the array will be accessed and processed. Thus, 'i' will loop 10 times, corresponding to the length of the End value array, and on each iteration, the value of i will increment by one to access a different array element.

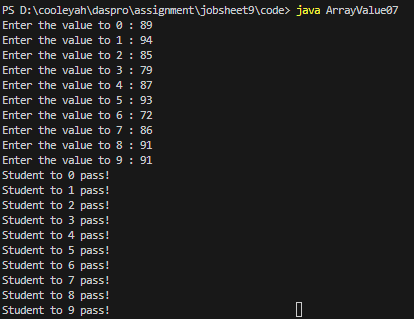
1. Ubah statement pada langkah nomor 6 menjadi seperti berikut ini, sehingga program hanya menampilkan nilai Mahasiswa yang lulus saja (yaitu mahasiswa yang memiliki nilai

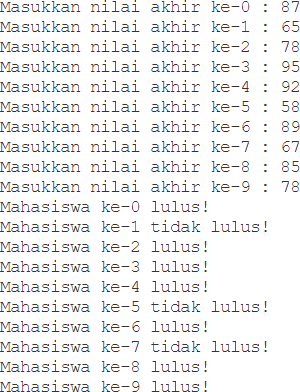
> 70):

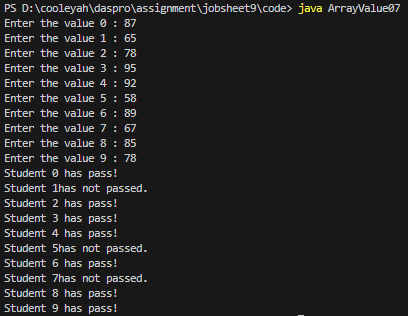
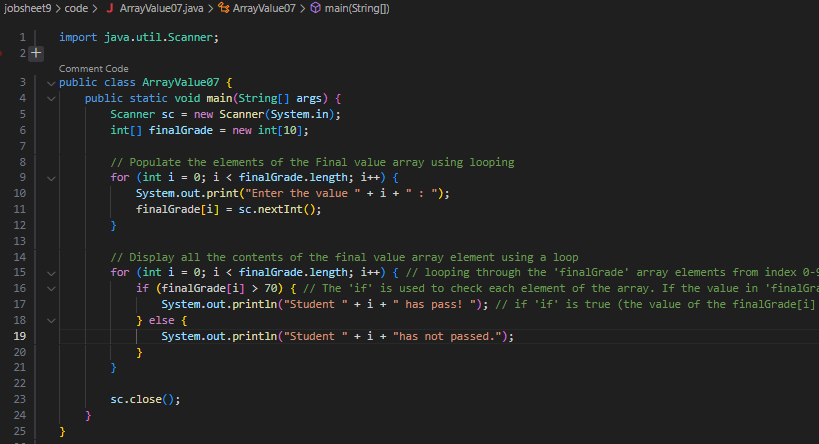


Jalankan program dan jelaskan alur program!





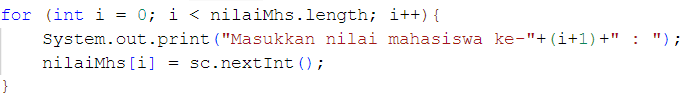
1. Modifikasi program agar menampilkan status kelulusan semua mahasiswa berdasarkan nilai, yaitu dengan menampilkan status mana mahasiswa yang lulus dan tidak lulus, seperti ilustrasi output berikut:
2. Push dan commit kode program ke github.



# Percobaan 3: Melakukan Operasi Aritmatika terhadap Elemen Array Waktu percobaan: 75 menit

Pada percobaan ini, akan dilakukan percobaan untuk menjumlahkan array. Program akan menerima input sebanyak 10 nilai mahasiswa. Kemudian program akan menampilkan nilai rata-rata dari dari 10 mahasiswa.

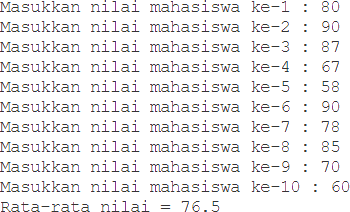
1. Buka text editor, buat file Java, kemudian simpan dengan nama ArrayRataNilai**XX**.java. (XX = nomor absen).
2. Buatlah struktur dasar java (membuat class dan method main).
3. Import dan deklarasikan Scanner untuk keperluan input.
4. Buat array nilaiMhs bertipe integer dengan kapasitas 10. Kemudian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



1. Gunakan perulangan untuk menghitung jumlah keseluruhan nilai dalam array nilaiMhs, sebagai berikut:
2. Kemudian hitung nilai rata-rata dengan cara nilai total dibagi jumlah elemen dari array nilaiMhs:

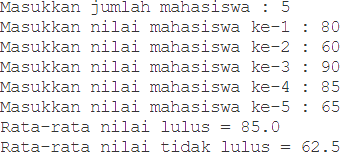


1. Amati hasilnya sebagai berikut:



1. Push dan commit kode program ke github.

# Pertanyaan

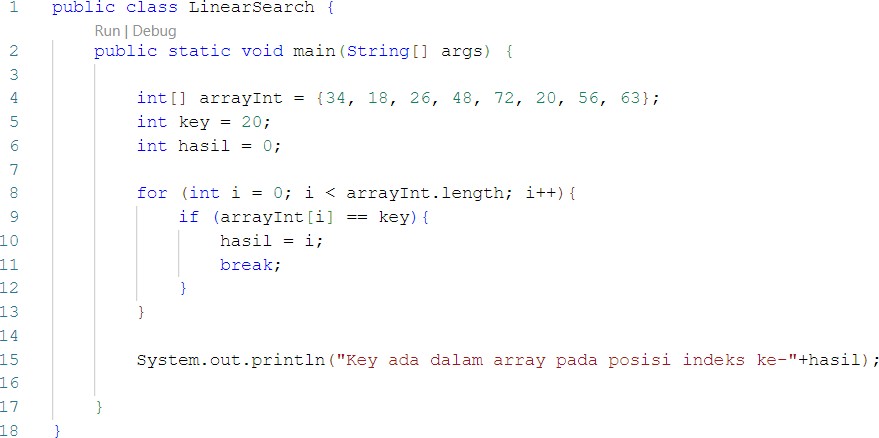
1. Modifikasi kode program pada praktikum percobaan 3 di atas (ArrayRataNilai**XX**.java) agar program dapat menampilkan banyaknya mahasiswa yang lulus, yaitu mahasiswa yang memiliki lebih besar dari 70 (>70).
2. Modifikasi program pada praktikum percobaan 3 di atas (ArrayRataNilai**XX**.java) sehingga program menerima jumlah elemen berdasarkan input dari pengguna dan mengeluarkan output seperti berikut ini:
3. Push dan commit kode program ke github.

# Percobaan 4: Searching Waktu percobaan: 45 menit

1. Buka text editor, buat file Java, kemudian simpan dengan nama LinearSearch**XX**.java. (XX

= nomor absen).

1. Tambahkan kode berikut ini:



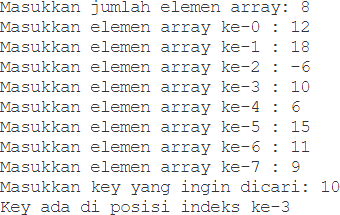
1. Jalankan program tersebut. Amati apa yang terjadi. Cocokkan dengan output berikut:



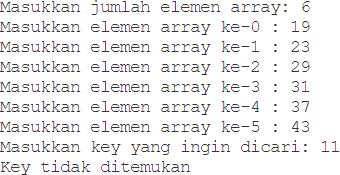
1. Push dan commit kode program ke github.

# Pertanyaan

1. Jelaskan maksud dari statement **break;** pada baris ke-11 kode program percobaan 4 di atas.
2. Modifikasi kode program pada percobaan 4 di atas sehingga program dapat menerima input berupa banyaknya elemen array, isi array, dan key yang ingin dicari. Lalu cetak ke layar indeks posisi elemen dari key yang dicari. Contoh hasil program:



1. Modifikasi program pada percobaan 4 di atas, sehingga program akan memberikan pesan "key tidak ditemukan" jika key tidak ada di dalam array. Contoh tampilan program sebagai berikut:



1. Push dan commit kode program ke github.

# 3. Tugas

**Waktu pengerjaan: 150 menit**

1. Buat program untuk menghasilkan nilai tertinggi, nilai terendah, dan rata-rata dari suatu array berisi bilangan bertipe integer.

Ketentuan:

* + Input: Banyaknya elemen, nilai tiap elemen
  + Output: Nilai tertinggi, nilai terendah, nilai rata-rata

1. Implementasikan *flowchart* yang telah dibuat pada tugas pertemuan 9 mata kuliah Dasar Pemrograman terkait project kelompok ke dalam kode program Java.

Push dan commit hasil kode program Anda ke repository github project Anda.

Catatan: tugas hanya boleh menerapkan materi dari pertemuan 1 hingga pertemuan 9.