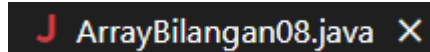


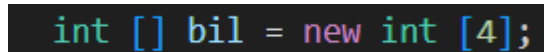
Nama : Dicky Darmawan
Kelas : TI-1B
No. Absen : 08
NIM : 244107020037

Percobaan 1: Mengisi Elemen Array

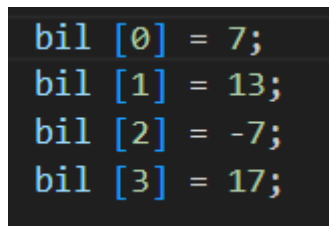
1. Buka text editor, buat class Java baru dengan nama ArrayBilanganXX.java.
(XX=nomorabsen)



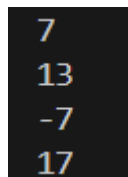
2. Buat array bertipe integer dengan nama bil dengan kapasitas 4 elemen.



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



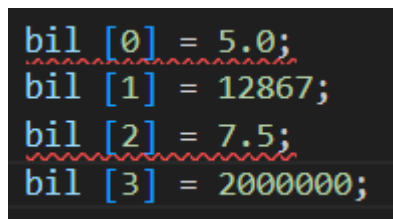
4. Tampilkan ke layar semua isi elemennya:
5. Cocokkan dan amati hasilnya dengan gambar berikut ini:



6. 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?



Terjadi error karena tipe data yang digunakan hanya cocok untuk bilangan bulat, bukan desimal

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

```
int [] bil = {7, 13, -7, 17};
```

- Ubah statement pada langkah No 4 menjadi seperti berikut

Apa keluaran dari program? Jelaskan maksud dari statement tersebut.

```
7  
13  
-7  
17
```

maksud dari statement tersebut adalah selama $i < 4$ maka akan tetap di print elemen dari array

- Jika kondisi pada statement for-loop di atas diubah menjadi: $i \leq 4$, apa keluaran dari program? Mengapa demikian?

```
7  
13  
-7  
17
```

tetap sama, tapi dengan rentang yang berbeda. Tetap sama karena elemen array dihitung dari 0 jadi elemen array ke 4 kosong atau tidak ada karena tidak diinisialisasikan

- 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)

```
J ArrayNilai08.java 1
```

2. Buatlah struktur dasar java (membuat class dan method main).

```
public class ArrayNilai08 {  
    Run main | Debug main | Run | Debug  
    public static void main(String[] args) {
```

3. Tambahkan import library Scanner.

```
import java.util.Scanner;
```

4. Buat deklarasi array bertipe integer dengan nama nilaiAkhir dan berkapasitas 10 elemen seperti di bawah ini :

```
int nilaiAkhir [] = new int [10];
```

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

```
int nilaiAkhir [] = new int [10];  
  
for (int i = 0 ; i < 10 ; i++) {  
    System.out.print("Masukkan nilai akhir ke-" + i + " : ");  
    nilaiAkhir [i] = sc.nextInt();  
}
```

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

```
}  
for (int i = 0; i < 10; i++) {  
    System.out.println("Nilai akhir ke-" + i + " adalah " + nilaiAkhir [i]);  
}
```

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

```

Masukkan nilai akhir ke-0 : 89
Masukkan nilai akhir ke-1 : 98
Masukkan nilai akhir ke-2 : 88
Masukkan nilai akhir ke-3 : 78
Masukkan nilai akhir ke-4 : 65
Masukkan nilai akhir ke-5 : 89
Masukkan nilai akhir ke-6 : 88
Masukkan nilai akhir ke-7 : 85
Masukkan nilai akhir ke-8 : 84
Masukkan nilai akhir ke-9 : 89
Nilai akhir ke-0 adalah 89
Nilai akhir ke-1 adalah 98
Nilai akhir ke-2 adalah 88
Nilai akhir ke-3 adalah 78
Nilai akhir ke-4 adalah 65
Nilai akhir ke-5 adalah 89
Nilai akhir ke-6 adalah 88
Nilai akhir ke-7 adalah 85
Nilai akhir ke-8 adalah 84
Nilai akhir ke-9 adalah 89

```

8. 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?

```

Nilai akhir ke-0 adalah 89
Nilai akhir ke-1 adalah 89
Nilai akhir ke-2 adalah 88
Nilai akhir ke-3 adalah 78
Nilai akhir ke-4 adalah 85
Nilai akhir ke-5 adalah 86
Nilai akhir ke-6 adalah 89
Nilai akhir ke-7 adalah 92
Nilai akhir ke-8 adalah 95
Nilai akhir ke-9 adalah 99

```

Tidak terjadi perubahan, karena

perubahan dalam rentang sama saja, hanya menyesuaikan panjang atau banyak dari array

2. Apa yang dimaksud dengan kondisi: $i < \text{nilaiAkhir.length}$?

Adalah batasan yang biasanya digunakan dalam konteks perulangan untuk memastikan bahwa indeks i berada dalam batas yang valid dari array.

- 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.println("Mahasiswa ke-" + i + " lulus!");  
    }  
}
```

Jalankan program dan jelaskan alur program!

```
Mahasiswa ke-0 lulus!  
Mahasiswa ke-2 lulus!  
Mahasiswa ke-4 lulus!  
Mahasiswa ke-5 lulus!  
Mahasiswa ke-6 lulus!  
Mahasiswa ke-7 lulus!  
Mahasiswa ke-8 lulus!  
Mahasiswa ke-9 lulus!
```

Program hanya akan mencetak nilai mahasiswa yang > 70 dengan keterangan lulus, jika < 70 maka tidak akan mencetak apapun

- 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:

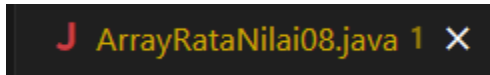
```
Masukkan nilai akhir ke-0 : 70  
Masukkan nilai akhir ke-1 : 69  
Masukkan nilai akhir ke-2 : 77  
Masukkan nilai akhir ke-3 : 85  
Masukkan nilai akhir ke-4 : 86  
Masukkan nilai akhir ke-5 : 89  
Masukkan nilai akhir ke-6 : 55  
Masukkan nilai akhir ke-7 : 62  
Masukkan nilai akhir ke-8 : 89  
Masukkan nilai akhir ke-9 : 92  
Mahasiswa ke-0 tidak lulus!  
Mahasiswa ke-1 tidak lulus!  
Mahasiswa ke-2 lulus!  
Mahasiswa ke-3 lulus!  
Mahasiswa ke-4 lulus!  
Mahasiswa ke-5 lulus!  
Mahasiswa ke-6 tidak lulus!  
Mahasiswa ke-7 tidak lulus!  
Mahasiswa ke-8 lulus!  
Mahasiswa ke-9 lulus!
```

- Push dan commit kode program ke github.

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

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 10 mahasiswa.

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).

```
public class ArrayRataNilai08 {  
    Run main | Debug main | Run | Debug  
    public static void main(String[] args) {
```

3. Import dan deklarasikan Scanner untuk keperluan input.

```
Scanner sc = new Scanner(System.in);
```

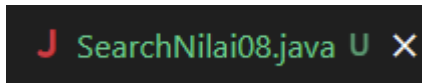
4. Buat array nilaiMhs bertipe integer dengan kapasitas 10. Kemudian deklarasikan variable total dan rata2 seperti gambar berikut ini
5. Isi array nilaiMhs dengan nilai dari input pengguna, sebagai berikut:
6. Gunakan perulangan untuk menghitung jumlah keseluruhan nilai dalam array nilaiMhs, sebagai berikut:
7. Kemudian hitung nilai rata-rata dengan cara nilai total dibagi jumlah elemen dari array nilaiMhs:
8. Amati hasilnya sebagai berikut:

```
Masukkan nilai mahasiswa ke-1 : 80  
Masukkan nilai mahasiswa ke-2 : 56  
Masukkan nilai mahasiswa ke-3 : 89  
Masukkan nilai mahasiswa ke-4 : 88  
Masukkan nilai mahasiswa ke-5 : 92  
Masukkan nilai mahasiswa ke-6 : 93  
Masukkan nilai mahasiswa ke-7 : 78  
Masukkan nilai mahasiswa ke-8 : 65  
Masukkan nilai mahasiswa ke-9 : 55  
Masukkan nilai mahasiswa ke-10 : 78  
Rata-rata nilai = 77.4
```

9. Push dan commit kode program ke github.

Percobaan 4: Searching

1. Buka text editor, buat file Java, kemudian simpan dengan nama SearchNilaiXX.java. (XX=nomor absen).



2. Tambahkan kode berikut ini:

```
public class SearchNilai08 {  
    Run main | Debug main | Run | Debug  
    public static void main(String[] args) {  
        int [] arrNilai = {80, 85, 78, 96, 90, 82, 86};  
        int key = 90;  
        int hasil = 0;  
  
        for (int i = 0; i < arrNilai.length; i++){  
            if (key == arrNilai[i]) {  
                hasil = i;  
                break;  
            }  
        }  
        System.out.println();  
        System.out.println("Nilai " + key + " ketemu di index ke-" + hasil);  
        System.out.println();  
    }  
}
```

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

```
Nilai 90 ketemu di index ke-4
```

4. Push dan commit kode program ke github.

Pertanyaan

1. Jelaskan maksud dari statement break; pada baris ke-10 kode program percobaan 4 di atas.
2. 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:

3. 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:

```
Masukkan banyaknya nilai yang akan diinput: 5
Masukkan nilai mahasiswa ke-1: 45
Masukkan nilai mahasiswa ke-2: 85
Masukkan nilai mahasiswa ke-3: 65
Masukkan nilai mahasiswa ke-4: 45
Masukkan nilai mahasiswa ke-5: 85
Masukkan nilai yang ingin dicari: 85
Nilai 85 ketemu di index ke-1
```

- 4.
5. 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.

```
Masukkan nilai mahasiswa ke-1: 89
Masukkan nilai mahasiswa ke-2: 88
Nilai ke-0: 89
Nilai ke-1: 88
Rata-rata nilai: 88.5
Nilai tertinggi: 89.0
Nilai terendah: 88.0
```

2. 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.

```
Masukkan jumlah pesanan: 2
Masukkan nama pesanan ke-1: Ayam
Masukkan harga pesanan ke-1: 12000
Masukkan nama pesanan ke-2: Burger
Masukkan harga pesanan ke-2: 9000
Daftar Pesanan:
1. Ayam= Rp12000.0
2. Burger= Rp9000.0
Total Biaya: Rp21000.0
```

3. 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.

Daftar Menu Makanan:

Nasi Goreng

Mie ayam

Roti Bakar

Kentang Goreng

Teh Tarik

Cappucino

Chocolate Ice

Masukkan nama makanan yang dicari: teh manis

teh manis tidak ada di menu