

## Laporan praktikum dasar pemrograman jobsheet 9

Nama : Muhammad Shabran

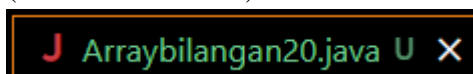
Nim : 244107020112

Kelas :1 D TI

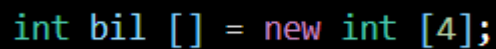
Absen : 20

### Percobaan 1

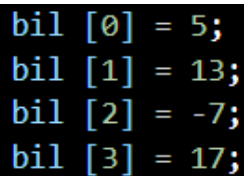
1. Buka text editor, buat class Java baru dengan nama ArrayBilanganXX.java.  
(XX=nomor absen)



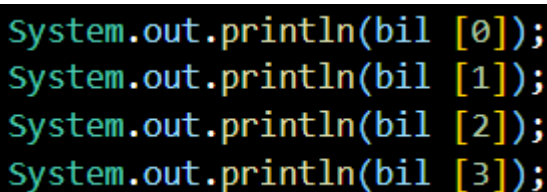
2. Buat array bertipe integer dengan nama bil dengan kapasitas 4 elemen.



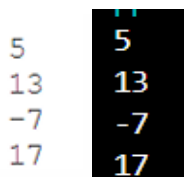
3. Isi masing-masing elemen array bil tadi dengan angka 5, 13, -7, 17.



4. Tampilkan ke layar semua isi elemennya:



5. Cocokkan dan amati hasilnya dengan gambar berikut ini:



### Pertanyaan 1:

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

Jawab :

jika masing masing elemen array diubah dengan angka diatas maka program akan error,karena variabel array merupakan int,sedangkan 5.0, dan 7.5 merupakan tipe data double (bukan int)

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

```
int[] bil = {5, 13, -7, 17};
```

3. Ubah statement pada langkah No 4 menjadi seperti berikut

```
for (int i = 0; i < 4; i++){  
    System.out.println(bil[i]);  
}
```

Apa keluaran dari program? Jelaskan maksud dari statement tersebut.

Jawab:

Keluaran dari program akan tetap sama, fungsi dari statemen diatas yaitu untuk melakukan sout dengan perulangan ,tanpa perlu melakukan sout satu per satu

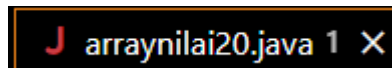
4. Jika kondisi pada statement for-loop di atas diubah menjadi:  $i \leq 4$ , apa keluaran dari program? Mengapa demikian?

Jawab : keluaran akan tetap sama ,karna index bil hanya sampai index 3

5. Push dan commit kode program ke github.

## Percobaan 2

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).

```
public class arraynilai20 {  
    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:

```
for (int i = 0; i <10; i++){  
    System.out.print("masukan 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 : 78	masukan nilai akhir ke 0 = 78
Masukkan nilai akhir ke-1 : 89	masukan nilai akhir ke 1 = 89
Masukkan nilai akhir ke-2 : 94	masukan nilai akhir ke 2 = 94
Masukkan nilai akhir ke-3 : 85	masukan nilai akhir ke 3 = 85
Masukkan nilai akhir ke-4 : 79	masukan nilai akhir ke 4 = 79
Masukkan nilai akhir ke-5 : 87	masukan nilai akhir ke 5 = 87
Masukkan nilai akhir ke-6 : 93	masukan nilai akhir ke 6 = 93
Masukkan nilai akhir ke-7 : 72	masukan nilai akhir ke 7 = 72
Masukkan nilai akhir ke-8 : 86	masukan nilai akhir ke 8 = 86
Masukkan nilai akhir ke-9 : 91	masukan nilai akhir ke 9 = 91
Nilai akhir ke-0 adalah 78	nilai akhir ke 0 adalah 78
Nilai akhir ke-1 adalah 89	nilai akhir ke 1 adalah 89
Nilai akhir ke-2 adalah 94	nilai akhir ke 2 adalah 94
Nilai akhir ke-3 adalah 85	nilai akhir ke 3 adalah 85
Nilai akhir ke-4 adalah 79	nilai akhir ke 4 adalah 79
Nilai akhir ke-5 adalah 87	nilai akhir ke 5 adalah 87
Nilai akhir ke-6 adalah 93	nilai akhir ke 6 adalah 93
Nilai akhir ke-7 adalah 72	nilai akhir ke 7 adalah 72
Nilai akhir ke-8 adalah 86	nilai akhir ke 8 adalah 86
Nilai akhir ke-9 adalah 91	nilai akhir ke 9 adalah 91

Pertanyaan 2.

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

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

Jalankan program. Apakah terjadi perubahan? Mengapa demikian?

Jawab:

Tidak terjadi perubahan karena nilaiakhir.lenght memiliki nilai index 10

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

Jawab :

3. Untuk membatasi kondisi  $i$  adala kurang dari indek array nilai akhir

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!

Jawab :

Program akan membaca nilai yang di input satu persatu, kemudian nilai yang lebih dari 70 ,kemudian akan melakukan sout untuk nilai yang lebih dari 70.

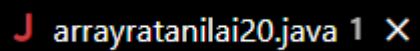
4. Modifikasi program agar menampilkan status kelulusan semua mahasiswa berdasarkan nilai, yaitu dengan menampilkan status mana mahasiswa yang lulus dan tidak lulus, seperti ilustrasi output berikut:

```
for (int i = 0; i < 10; i++){  
    if (nilaiakhir[i] > 70 ) {  
        System.out.println("mahasiswa ke " + i + " lulus " );  
    }  
    else System.out.println("mahasiswa ke " + i + " tidak lulus " );}
```

5. Push dan commit kode program ke github.

### Percobaan 3

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 arrayratanilai20 {  
    Run | Debug  
    public static void main(String[] args) {
```

3. Import dan deklarasikan Scanner untuk keperluan input.

```
import java.util.Scanner;  
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:

```
int[] nilaiMhs = new int[10];  
double total = 0;  
double rata2;
```

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

```
for (int i = 0; i < nilaiMhs.length; i++) {  
    System.out.print("Masukkan nilai mahasiswa ke-" + (i+1) + " : ");  
    nilaiMhs[i] = sc.nextInt();  
}
```

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

```
for (int i = 0; i < nilaiMhs.length; i++) {  
    total += nilaiMhs[i];  
}
```

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

```
rata2 = total / nilaiMhs.length;  
System.out.println("Rata rata nilai = " + rata2);
```

### Pertanyaan 3

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 ( $>70$ ).
2. 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:

```
Masukkan jumlah mahasiswa : 5  
Masukkan nilai mahasiswa ke-1 : 80  
Masukkan nilai mahasiswa ke-2 : 60  
Masukkan nilai mahasiswa ke-3 : 90  
Masukkan nilai mahasiswa ke-4 : 85  
Masukkan nilai mahasiswa ke-5 : 65  
Rata-rata nilai lulus = 85.0  
Rata-rata nilai tidak lulus = 62.5
```

3. Push dan commit kode program ke github.

### Percobaan 4

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



2. Tambahkan kode berikut ini:

```
1 public class SearchNilai {  
    Run | Debug  
2     public static void main(String[] args) {  
3         int[] arrNilai = {80, 85, 78, 96, 90, 82, 86};  
4         int key = 90;  
5         int hasil = 0;  
6  
7         for (int i = 0; i < arrNilai.length; i++){  
8             if (key == arrNilai[i]){  
9                 hasil = i;  
10                break;  
11            }  
12        }  
13        System.out.println();  
14        System.out.println("Nilai "+key+" ketemu di indeks ke-"+hasil);  
15        System.out.println();  
16    }  
17 }
```

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

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

#### Pertanyaan 4

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

Jawab :

Statement `break;` pada kode berfungsi untuk menghentikan loop pencarian segera setelah nilai yang dicari ditemukan, sehingga menghindari pemeriksaan elemen yang tidak perlu.

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.
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:
4. 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.
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:  
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.
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.

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.