

Laporan Tugas 4 (Evaluasi Praktikum)

Nama : Nur Ikhwan Alfiansyah

Nim : 13020220131

Kelas : A4 – TI

Evaluasi Praktikum Modul 3 (Struktur Kontrol dan Array)

1. Apakah perbedaan antara struktur kontrol percabangan if-else dan switch-case?

Jawaban : *If-else* merupakan struktur kontrol percabangan yang digunakan untuk memilih antara dua pilihan berdasarkan kondisi boolean, sedangkan *Switch-case* digunakan untuk memilih antara banyak pilihan berdasarkan nilai variabel yang spesifik.

2. Kapan digunakan struktur kontrol if-else dan switch-case

Jawaban : *If-else* digunakan ketika terdapat kondisi yang harus dievaluasi secara kondisional. Contoh, jika kondisi A terpenuhi, lakukan X, jika tidak, lakukan Y. Sedangkan *Switch-case* digunakan ketika terdapat banyak pilihan dengan nilai yang sudah diketahui sebelumnya.

3. Pada program 2, tambahkan perintah untuk memilih 2 opsi menggunakan kontrol switch..case. opsi pilihah 1=inputNilai() Pilihan 2=inputNilaiBaru()

Jawaban :

```
1  /*
2  Nama      : Nur Ikhwan Alfiansyah
3  Nim       : 13020220131
4  Kelas     : A4 -TI
5  Modul     : Modul 3, No 3.
6  Waktu Pengerjaan : 25/03/2024, 20:00 WITA
7  */
8
9  import java.util.Scanner;
10
11 public class TestNilai {
12     public static void main(String[] args) {
13         HitungRata hitung = new HitungRata();
14         Scanner input = new Scanner(System.in);
15         System.out.print("Masukkan Jumlah Data : ");
16         int banyakData = input.nextInt();
17         int nilai[] = new int[banyakData];
18         int pilihan;
19
20         System.out.println("Menu:");
21         System.out.println("1. Input Nilai");
22         System.out.println("2. Input Nilai Baru");
23         System.out.print("Pilih menu: ");
24         pilihan = input.nextInt();
25
26         switch(pilihan) {
27             case 1:
28                 System.out.println("Masukkan Nilai : ");
29                 hitung.inputNilai(nilai);
30                 System.out.println("\nDaftar Nilai : ");
31                 hitung.cetakNilai(nilai);
32                 System.out.println("Rata Nilai      : " + hitung.rataNilai(banyakData));
33                 break;
34             case 2:
35                 System.out.println("Masukkan Nilai Baru : ");
36                 hitung.inputNilaiBaru(banyakData);
37                 System.out.println("Daftar Nilai Baru : ");
38                 hitung.cetakNilaiBaru();
39                 break;
40             default:
41                 System.out.println("Pilihan tidak valid");
42         }
43     }
44 }
```

Output :

```
Masukkan Jumlah Data : 2
Menu:
1. Input Nilai
2. Input Nilai Baru
Pilih menu: 1
Masukkan Nilai :
80
89

Daftar Nilai :
80
89

Rata Nilai      : 84.5
```

4. Apakah perbedaan antara struktur kontrol perulangan *while* dan *do-while*?

Jawaban : Pada perulangan *while* memeriksa kondisi sebelum iterasi pertama, sementara *do-while* menjalankan iterasi pertama sebelum memeriksa kondisi, maksudnya *do-while* akan menjalankan satu iterasi tanpa melihat kondisi awalnya salah atau benar.

5. Kapan digunakan struktur kontrol *for*?

Jawaban : Struktur kontrol *for* sebaiknya digunakan ketika jumlah iterasi sudah diketahui sebelumnya atau ketika diperlukan iterasi dengan penambahan atau pengurangan nilai secara teratur.

6. Apakah perbedaan antara *Array* dan *ArrayList*?berilah contoh masing-masing!

Jawaban : *Array* memiliki ukuran tetap dalam artian dalam penambahan elemen harus terlebih dahulu mendeklarasikan ukuran awalnya.

Contoh *Array* :

```
1  /*
2  Nama           : Nur Ikhwan Alfiansyah
3  Nim            : 13020220131
4  Kelas          : A4 -TI
5  Modul          : Modul 3, No 6.
6  Waktu Pengerjaan : 25/03/2024, 21:00 WITA
7  */
8
9  int[] angka = new int [2];
10 angka[0] = 10;
11 angka[2] = 20;
```

ArrayList memiliki ukuran dinamis yang dapat disesuaikan atau dalam penambahan elemennya *ArrayList* tidak perlu mendeklarasikan ukuran awalnya.

Contoh *ArrayList* :

```
1  /*
2  Nama           : Nur Ikhwan Alfiansyah
3  Nim            : 13020220131
4  Kelas          : A4 -TI
5  Modul          : Modul 3, No 6.
6  Waktu Pengerjaan : 25/03/2024, 21:00 WITA
7  */
8
9  ArrayList<String> nama = new ArrayList<>();
10 nama.add("Fian");
11 nama.add("Didit");
```

7. Buatlah contoh program yang mengimplementasikan *HashMap* dengan memasukkan nilai dan key melalui keyboard!

Jawaban :

Output:

```
PS C:\Users\ASUS\Documents\UMI\SEMESTER 4\PB0\Tugas4\Tugas4_SourceCode> javac HashMap7.java
PS C:\Users\ASUS\Documents\UMI\SEMESTER 4\PB0\Tugas4\Tugas4_SourceCode> java HashMap7

=== Masukkan kunci-nilai ke dalam HashMap (ketik 'keluar' untuk mengakhiri) ===
Masukkan kunci(Karakter) : Fian
Masukkan nilai(Bil.Bulat) : 100
Masukkan kunci(Karakter) : Didit
Masukkan nilai(Bil.Bulat) : 94
Masukkan kunci(Karakter) : Fajrul
Masukkan nilai(Bil.Bulat) : 90
Masukkan kunci(Karakter) : keluar

Isi HashMap :
Kunci : Fian, Nilai : 100
Kunci : Didit, Nilai : 94
Kunci : Fajrul, Nilai : 90
```

Evaluasi Praktikum Modul 4 (Konsep Pemrograman Berorientasi Objek)

1. Berdasarkan ke tiga program di atas Class utama, Class Orang dan Class Mahasiswa, manakah yang menunjukkan konsep pewarisan dan polimorfisme! Jelaskan konsep tersebut sesuai program tersebut!

Jawaban :

Konsep pewarisan (inheritance) terjadi ketika sebuah class mewarisi atribut dan metode dari class lain. Pada Program ini, *Class Mahasiswa* mewarisi *Class Orang*, karena Mahasiswa adalah turunan dari Orang. Konsep ini menunjukkan bahwa *Class Mahasiswa* memiliki semua atribut dan method yang dimiliki oleh *Class Orang*.

Konsep polimorfisme terjadi ketika suatu objek dapat memiliki banyak bentuk, di mana beberapa method dengan nama yang sama tetapi parameter yang berbeda didefinisikan dalam satu *Class*. Pada Program ini, kita memiliki overriding, di mana method *Orang()* didefinisikan kembali dalam *Class Mahasiswa*.

Dalam program di atas, proses pewarisan terjadi di bagian berikut:

Class *Mahasiswa* mewarisi Class *Orang*, ditunjukkan oleh pernyataan *extends Orang*.

Proses polimorfisme terjadi di bagian berikut:

Override (Mengganti implementasi method superclass) dari constructor *Orang()* dalam kelas *Mahasiswa*, di mana constructor *Orang()* didefinisikan kembali dengan perilaku yang berbeda.

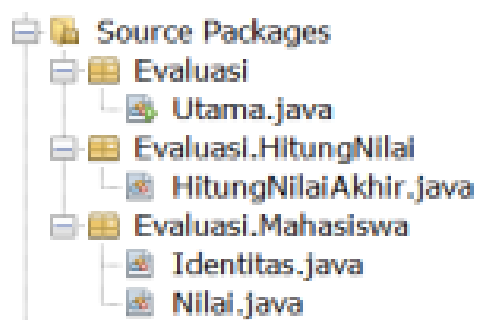
2. Tambahkan static pada method *info()* Class *Orang* dan Class *Mahasiswa* kemudian lakukan pemanggilan method *info()* pada program utama (Class utama)!

Jawaban :

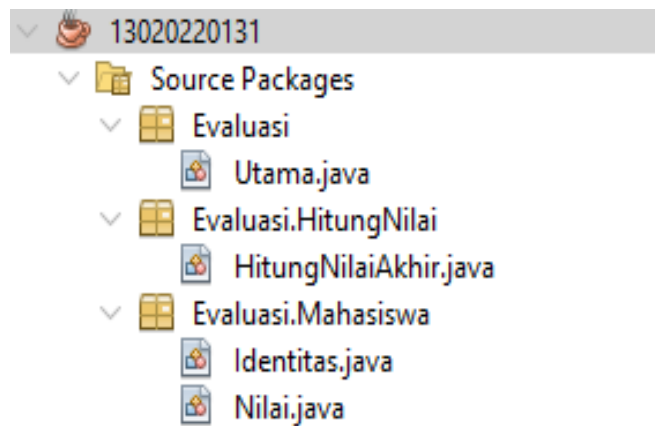
Output :

```
PS C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas4\Tugas4_SourceCode> javac Utamaa.java
PS C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas4\Tugas4_SourceCode> java Utamaa
Ini adalah info dari class Orang.
Ini adalah info dari class Mahasiswa.
```

3. Buatlah sebuah project dengan nama project stambuk anda dan buatlah pengorganisasian package dan class seperti berikut.



Jawaban :



Output:

