

**LAPORAN
PEMROGRAMAN BERORIENTASI OBJEK**



Nama : Muh Ilham Nur Hidayat Akbar
Stambuk :13020220041
Kelas : A1
Dosen : Mardiyah Hasnawi, S.Kom.,M.T.

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER UNIVERSITAS
MUSLIM INDONESIA MAKASSAR
2024**

Evaluasi Praktikum Modul 3 Struktur Kontrol dan Array

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

Jawab:

Struktur kontrol percabangan if-else dan switch-case digunakan dalam pemrograman untuk membuat keputusan berdasarkan kondisi tertentu. Meskipun keduanya memiliki tujuan yang sama, ada beberapa perbedaan antara keduanya:

- Kondisi yang Diuji:
 - **if-else:** Dalam struktur if-else, kondisi yang diuji bisa berupa ekspresi boolean atau kondisi yang menghasilkan nilai boolean (true/false). Kondisi ini bisa sangat kompleks dan dapat mengevaluasi berbagai jenis kondisi.
 - **switch-case:** Dalam struktur switch-case, kondisi yang diuji biasanya berupa nilai yang memenuhi tipe data yang ditentukan (biasanya integer, karakter, atau enumerasi). Setiap case di switch membandingkan nilai yang sama terhadap kondisi switch.
- Eksekusi Kode:
 - **if-else:** Dalam struktur if-else, hanya satu blok kode yang dieksekusi berdasarkan kondisi yang terpenuhi. Jika kondisi di if terpenuhi, blok kode di dalamnya dijalankan. Jika tidak, blok kode di dalam else (jika ada) akan dijalankan.
 - **switch-case:** Dalam struktur switch-case, eksekusi kode bergantung pada nilai yang cocok dengan nilai dalam case. Ketika nilai dalam 'case' cocok dengan kondisi switch, blok kode yang sesuai dengan case tersebut dieksekusi. Kemudian, eksekusi kode keluar dari struktur switch-case.
- Fleksibilitas:
 - **if-else:** Struktur if-else lebih fleksibel karena kondisi yang diuji bisa sangat kompleks dan bervariasi dari satu kondisi ke kondisi lainnya. Ini memungkinkan untuk menguji berbagai kondisi dalam satu struktur kontrol.
 - **switch-case:** Struktur 'switch-case' lebih terbatas dalam hal fleksibilitas karena hanya dapat digunakan untuk membandingkan nilai yang sama terhadap kondisi 'switch'. Ini membuatnya lebih cocok untuk kasus-kasus di mana kita memiliki serangkaian nilai yang terbatas yang ingin kita bandingkan.

Meskipun keduanya memiliki perbedaan dalam penggunaan dan implementasi, keduanya dapat digunakan secara bersamaan untuk mencapai logika kontrol percabangan yang lebih kompleks. Pemilihan antara keduanya tergantung pada kebutuhan spesifik dari situasi pemrograman yang dihadapi.

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

Jawab:

- If-else:
 - Digunakan ketika ada kebutuhan untuk mengevaluasi kondisi yang kompleks atau ketika terdapat banyak kondisi yang memerlukan penanganan khusus.
 - Cocok digunakan jika kondisi yang diuji bergantung pada berbagai variabel atau jika logika kondisi kompleks.
 - Dapat menangani kondisi dengan lebih fleksibel, karena tidak terbatas pada jumlah kasus yang harus dievaluasi.

➤ Switch-case:

- Digunakan ketika terdapat satu ekspresi atau variabel yang harus dievaluasi terhadap beberapa nilai tetap.
- Cocok digunakan jika terdapat banyak kasus dengan nilai yang berbeda-beda dan hasilnya sederhana serta tidak kompleks.
- Biasanya lebih efisien daripada if-else ketika ada banyak kasus yang harus dipertimbangkan, karena strukturnya memungkinkan untuk penanganan kasus yang bersifat linear.

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

Pilihan 2=inputNilaiBaru()

Jawab:

➤ Codingan:

```
1 package testnilai;
2
3 /*
4  * @author 13020220041
5  * Nama : Muh Iham Nur hidayat Akbar
6  * Materi : Tugas 4 Evaluasi Praktikum 1
7  * Waktu : 27-03-2024, 15.05
8  */
9
10 import java.util.Scanner;
11
12 public class TestNilai {
13
14     public static void main(String[] args) {
15         HitungRata hitung = new HitungRata();
16         Scanner input = new Scanner(System.in);
17         int banyakData;
18
19         System.out.print("Masukkan Jumlah Data : ");
20         banyakData = input.nextInt();
21         int nilai[] = new int[banyakData];
22
23         System.out.println("Menu:");
24         System.out.println("1. Input Nilai");
25         System.out.println("2. Input Nilai Baru");
26         System.out.print("Pilih opsi: ");
27         int pilihan = input.nextInt();
28
29         switch (pilihan) {
30             case 1:
31                 System.out.print("Masukkan Nilai : ");
32                 hitung.inputNilai(nilai);
33                 System.out.print("Daftar Nilai : ");
34                 hitung.cetakNilai(nilai);
35                 System.out.println("Rata Nilai : "+ hitung.rataNilai(banyakData));
36                 break;
37             case 2:
38                 System.out.print("Masukkan Nilai Baru: ");
39                 hitung.inputNilaiBaru(banyakData);
40                 System.out.print("Daftar Nilai Baru : ");
41                 hitung.cetakNilaiBaru();
42                 break;
43             default:
44                 System.out.println("Pilihan tidak valid!");
45         }
46     }
47 }
```

➤ **Class HitungRata:**

```
1 package testnilai;
2
3 /*
4  @author 13020220041
5  Nama    : Muh Iham Nur hidayat Akbar
6  Materi  : Tugas 4 Evaluasi Praktikum 1
7  Waktu   : 27-03-2024, 15.05
8  */
9
10 import java.util.Scanner;
11
12 public class TestNilai {
13
14     public static void main(String[] args) {
15         HitungRata hitung = new HitungRata();
16         Scanner input = new Scanner(System.in);
17         int banyakData;
18
19         System.out.print("Masukkan Jumlah Data : ");
20         banyakData = input.nextInt();
21         int nilai[] = new int[banyakData];
22
23         System.out.println("Menu:");
24         System.out.println("1. Input Nilai");
25         System.out.println("2. Input Nilai Baru");
26         System.out.print("Pilih opsi: ");
27         int pilihan = input.nextInt();
28
29         switch (pilihan) {
30             case 1:
31                 System.out.print("Masukkan Nilai : ");
32                 hitung.inputNilai(nilai);
33                 System.out.print("Daftar Nilai : ");
34                 hitung.cetakNilai(nilai);
35                 System.out.println("Rata Nilai : "+ hitung.rataNilai(banyakData));
36                 break;
37             case 2:
38                 System.out.print("Masukkan Nilai Baru: ");
39                 hitung.inputNilaiBaru(banyakData);
40                 System.out.print("Daftar Nilai Baru : ");
41                 hitung.cetakNilaiBaru();
42                 break;
43             default:
44                 System.out.println("Pilihan tidak valid!");
45         }
46     }
47 }
```

➤ **Output Codingan:**

```
Output - TestNilai (run) x
run:
Masukkan Jumlah Data : 5
Menu:
1. Input Nilai
2. Input Nilai Baru
Pilih opsi: 1
Masukkan Nilai : 80
90
70
50
60
Daftar Nilai : 80      90      70      50      60
Rata Nilai : 70.0
BUILD SUCCESSFUL (total time: 47 seconds)
```

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

Jawab:

- **While:** Perulangan while digunakan untuk melakukan eksekusi terhadap satu atau lebih perintah selama kondisi perulangan masih bernilai true dan akan berhenti jika kondisi perulangan bernilai false.
- **Do While:** Sama halnya dengan perulangan while, namun yang membedakan antara keduanya adalah do-while akan mengeksekusi terlebih dahulu perintah dalam blok perulangan lalu memeriksa kondisi dari perulangan tersebut. Perulangan do-while akan berhenti jika kondisi yang diberikan telah bernilai false.

5. Kapan digunakan struktur kontrol for?

Jawab:

- **For :** Perulangan for digunakan untuk mengeksekusi satu atau lebih perintah dalam blok for secara berulang-ulang. Perulangan for biasanya digunakan untuk mengulang beberapa perintah dengan jumlah iterasi yang telah ditentukan

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

Jawab:

- **Array:**
 - Array adalah kumpulan elemen dengan tipe data yang sama yang disimpan dalam memori secara berurutan.
 - Ukuran array ditentukan pada saat deklarasi dan tidak dapat diubah setelahnya.
 - Untuk mengakses elemen array, Anda menggunakan indeks.
- **Bentuk umum:**
`TipeData[] namaArray = new TipeData[ukuran];`
- **Contoh Array:**
`int[] angka = new int[3]; // Mendeklarasikan array integer dengan ukuran 3
angka[0] = 10;
angka[1] = 20;
angka[2] = 30;`
- **ArrayList:**
 - ArrayList adalah kelas dalam Java yang merupakan bagian dari paket java.util.
 - ArrayList mengimplementasikan struktur data list dengan ukuran dinamis yang dapat diubah sesuai kebutuhan.
 - Anda tidak perlu menentukan ukuran awal saat mendeklarasikan ArrayList, dan ukurannya akan diperluas secara otomatis ketika diperlukan.
 - Untuk mengakses elemen ArrayList, Anda menggunakan metode seperti get(), set(), dll.
- **Bentuk umum:**
`ArrayList<TipeData> namaArrayList = new ArrayList<>();`
- **Contoh implementasi dari class ArrayList :**
`import java.util.ArrayList;

// Mendeklarasikan ArrayList yang berisi angka integer
ArrayList<Integer> angkaList = new ArrayList<>();

// Menambahkan elemen ke ArrayList
angkaList.add(10);
angkaList.add(20);
angkaList.add(30);`

Perbedaan utama antara keduanya adalah bahwa Array memiliki ukuran tetap yang ditentukan saat deklarasi, sedangkan ArrayList memiliki ukuran dinamis yang dapat disesuaikan

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

Jawab:

➤ **Codingan:**

```
import java.util.HashMap;
import java.util.Map;
import java.util.Scanner;

public class HashMapX {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Map<String, Integer> hashMap = new HashMap<>();

        System.out.println("Masukkan jumlah data yang ingin dimasukkan:");
        int n = scanner.nextInt();

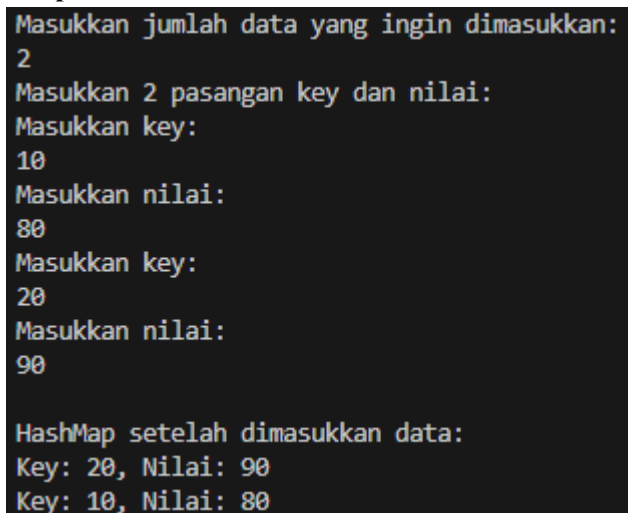
        System.out.println("Masukkan " + n + " pasangan key dan nilai:");

        for (int i = 0; i < n; i++) {
            System.out.println("Masukkan key:");
            String key = scanner.next();
            System.out.println("Masukkan nilai:");
            int value = scanner.nextInt();
            hashMap.put(key, value);
        }

        System.out.println("\nHashMap setelah dimasukkan data:");
        for (Map.Entry<String, Integer> entry : hashMap.entrySet()) {
            System.out.println("Key: " + entry.getKey() + ", Nilai: " + entry.getValue());
        }

        scanner.close();
    }
}
```

➤ **Output:**



```
Masukkan jumlah data yang ingin dimasukkan:
2
Masukkan 2 pasangan key dan nilai:
Masukkan key:
10
Masukkan nilai:
80
Masukkan key:
20
Masukkan nilai:
90

HashMap setelah dimasukkan data:
Key: 20, Nilai: 90
Key: 10, Nilai: 80
```

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!
Jawab:

➤ Dari ketiga program tersebut, yang menunjukkan konsep pewarisan dan polimorfisme adalah Class Mahasiswa.

➤ Pewarisan (Inheritance):

Pewarisan adalah konsep di mana sebuah class dapat mewarisi properti dan metode dari class lainnya. Dalam program tersebut, Class Mahasiswa merupakan turunan dari Class Orang karena Mahasiswa dideklarasikan menggunakan kata kunci extends Orang. Dengan demikian, Mahasiswa mewarisi properti dan metode dari Orang, termasuk properti nama. Ini berarti Mahasiswa memiliki semua yang dimiliki oleh Orang, ditambah dengan properti tambahan yang didefinisikan dalam kelas Mahasiswa.

➤ Polimorfisme:

Polimorfisme adalah konsep di mana sebuah objek dapat memiliki banyak bentuk. Dalam konteks program tersebut, polimorfisme dapat diilustrasikan dengan cara Mahasiswa memiliki metode Orang() yang digunakan untuk menginisialisasi objek Mahasiswa. Meskipun Mahasiswa adalah turunan dari Orang, saat Mahasiswa dibuat, konstruktor Orang dipanggil secara implisit. Ini menunjukkan polimorfisme di mana objek Mahasiswa, meskipun diturunkan dari Orang, dapat memiliki perilaku yang berbeda dari Orang.

Dengan demikian, program tersebut mencakup konsep pewarisan dan polimorfisme dengan menggunakan kelas Mahasiswa yang mewarisi sifat dari kelas Orang dan menunjukkan perilaku yang berbeda dalam konteks konstruktor.

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

Jawab:

➤ Utama.java

```
1 package Pertemuan2Modul413020220041;
2
3 /*
4  @author 13020220041
5  Nama : Muh Ilham Nur hidayat Akbar
6  Materi : Tugas 4 Evaluasi Praktikum 1
7  Waktu : 27-03-2024, 15.50
8  */
9
10 public class Orang {
11     public String nama;
12
13     public Orang() {
14         this.nama = "Ilham";
15     }
16
17     public Orang(String nama) {
18         this.nama = nama;
19     }
20
21     public static void info() {
22         System.out.println("Ini adalah class Orang");
23     }
24 }
```

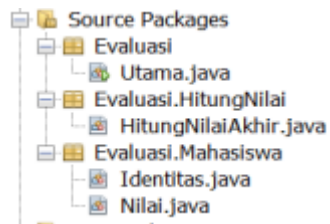
➤ Orang.java

```
1 package Pertemuan2Modul413020220041;
2
3 /*
4  * @author 13020220041
5  * Nama : Muh Ilham Nur hidayat Akbar
6  * Materi : Tugas 4 Evaluasi Praktikum 1
7  * Waktu : 27-03-2024, 15.50
8  */
9
10 public class Orang {
11     public String nama;
12
13     public Orang() {
14         this.nama = "Ilham";
15     }
16
17     public Orang(String nama) {
18         this.nama = nama;
19     }
20
21     public static void info() {
22         System.out.println("Ini adalah class Orang");
23     }
24 }
```

➤ Mahasiswa.java

```
1 package Pertemuan2Modul413020220041;
2
3 /*
4  * @author 13020220014
5  * Nama : Muh Ilham Nur hidayat Akbar
6  * Materi : Tugas 4 Evaluasi Praktikum 1
7  * Waktu : 27-03-2024, 16.00
8  */
9
10 public class Mahasiswa extends Orang {
11     private String stb;
12
13     public Mahasiswa() {
14         super();
15         this.stb = "13020220041";
16     }
17
18     public Mahasiswa(String stb, String nama) {
19         this.nama = nama;
20         this.stb = stb;
21     }
22
23     public static void info() {
24         System.out.println("Ini adalah class Mahasiswa");
25     }
26 }
```


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



Setelah itu Lengkapi Program pada soal yang diberikan

➤ **Codingan Identitas.java**

```
1  package Evaluasi.Mahasiswa;
2
3  /*
4   @author 13020220041
5   Nama : Muh Ilham Nur Hidayat Akbar
6   Materi : Tugas 4 Evaluasi Praktikum 1
7   Waktu : 27-03-2024, 16.05
8   */
9
10 public class Identitas {
11     public String nama;
12     public String stambuk;
13
14     public void setNama(String nama) {
15         this.nama = nama;
16     }
17
18     public String getNama() {
19         return nama;
20     }
21
22     public void setStambuk(String stambuk) {
23         this.stambuk = stambuk;
24     }
25
26     public String getStambuk() {
27         return stambuk;
28     }
29 }
```

➤ **Codingan Nilai.java**

```
1  package Evaluasi.Mahasiswa;
2
3  /*
4  @author 13020220041
5  Nama : Muh Ilham Nur Hidayat Akbar
6  Materi : Tugas 4 Evaluasi Praktikum 1
7  Waktu : 27-03-2024, 16.25
8  */
9
10 public class Nilai {
11     private int tugas1;
12     private int tugas2;
13     private int mid;
14     private int uas;
15
16
17     public void setTugas1(int tugas1) {
18         this.tugas1 = tugas1;
19     }
20
21     public int getTugas1() {
22         return tugas1;
23     }
24
25     public void setTugas2(int tugas2) {
26         this.tugas2 = tugas2;
27     }
28
29     public int getTugas2() {
30         return tugas2;
31     }
32
33     public void setMid(int mid) {
34         this.mid = mid;
35     }
36
37     public int getMid() {
38         return mid;
39     }
40
41
42     public void setUas(int uas) {
43         this.uas = uas;
44     }
45
46     public int getUas() {
47         return uas;
48     }
49 }
```

➤ Codingan HitungNilaiAkhir.java

```
1 package Evaluasi.HitungNilai;
2
3 /*
4  * @author 13020220041
5  * Nama : Muh Ilham Nur Hidayat Akbar
6  * Materi : Tugas 4 Evaluasi Praktikum 1
7  * Waktu : 27-03-2024, 16.45
8  */
9
10 public class HitungNilaiAkhir {
11     public double nilaiTugas(int tugas1, int tugas2) {
12         double rataTugas = (tugas1 + tugas2) / 2.0;
13         return rataTugas;
14     }
15
16     public double nilaiAkhir(double tugas, int mid, int uas) {
17         double nilaiAkhir = (tugas * 0.4) + (mid * 0.3) + (uas * 0.3);
18         return nilaiAkhir;
19     }
20 }
```

➤ Codingan Utama.java

```
1 package evaluasi;
2
3 /*
4  * @author 13020220041
5  * Nama : Muh Ilham Nur Hidayat Akbar
6  * Materi : Tugas 4 Evaluasi Praktikum 1
7  * Waktu : 27-03-2024, 17.00
8  */
9
10 import Evaluasi.HitungNilai.HitungNilaiAkhir;
11 import Evaluasi.Mahasiswa.*;
12
13 import javax.swing.JOptionPane;
14
15
16 public class Utama {
17
18     public static void main(String[] args) {
19         String nama = JOptionPane.showInputDialog("Masukkan Nama:");
20         String stambuk = JOptionPane.showInputDialog("Masukkan Stambuk:");
21
22         int tugas1 = Integer.parseInt(JOptionPane.showInputDialog("Masukkan Tugas 1:"));
23         int tugas2 = Integer.parseInt(JOptionPane.showInputDialog("Masukkan Tugas 2:"));
24         int mid = Integer.parseInt(JOptionPane.showInputDialog("Masukkan Nilai Mid:"));
25         int uas = Integer.parseInt(JOptionPane.showInputDialog("Masukkan Nilai UAS:"));
26
27         Identitas mhs = new Identitas();
28         mhs.setNama(nama);
29         mhs.setStambuk(stambuk);
30
31         HitungNilaiAkhir hitungNilai = new HitungNilaiAkhir();
32         double tugas = hitungNilai.nilaiTugas(tugas1, tugas2);
33
34         double na = hitungNilai.nilaiAkhir(tugas, mid, uas);
35
36         JOptionPane.showMessageDialog(null, "Nama: " + mhs.getNama() + "\nStambuk: " + mhs.getStambuk());
37
38         JOptionPane.showMessageDialog(null, "Tugas: " + tugas + "\nMid: " + mid + "\nUAS: " + uas + "\nNilai Akhir: " + na);
39     }
40 }
```

➤ **Output Codingan:**

Input

Masukkan Nama:

Muh Ilham Nur Hidayat Akbar

OK Cancel

Input

Masukkan Stambuk:

13020220041

OK Cancel

Input

Masukkan Tugas 1:

97

OK Cancel

Input

Masukkan Tugas 2:

98

OK Cancel

Input

Masukkan Nilai Mid:

99

OK Cancel

Input

Masukkan Nilai UAS:

100

OK Cancel

Message

Nama: Muh Ilham Nur Hidayat Akbar
Stambuk: 13020220041

OK

Message

Tugas: 97.5
Mid: 99
UAS: 100
Nilai Akhir: 98.7

OK