

LAPORAN TUGAS 4
PEMROGRAMAN BERORIENTASI OBJEK



Nama : Edo Ranov Anjasmara
Nim : 13020220010
Dosen : Mardiyah Hasnawih, S.Kom., M.T., MTA

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

LAPORAN TUGAS 4 PEMROGRAMAN BERORIENTASI OBJEK TUGAS EVALUASI PRATIKUM

Evaluasi Pratikum Modul 3

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

- if-else: Struktur kontrol if-else digunakan untuk menguji kondisi tunggal. Jika kondisi bernilai benar, maka blok kode yang terkait dengan if akan dieksekusi. Jika kondisi bernilai salah, maka blok kode yang terkait dengan else akan dieksekusi (jika ada). Struktur ini fleksibel karena kondisi dapat dievaluasi ke dalam nilai boolean.
- switch-case: Struktur kontrol switch-case juga digunakan untuk pengujian kondisi, tetapi biasanya digunakan ketika ada banyak kemungkinan nilai untuk diperiksa. Switch-case mengevaluasi ekspresi tunggal dan mencocokkan nilai ekspresi tersebut dengan kasus-kasus yang sesuai. Jika ada kecocokan, blok kode yang terkait dengan kasus yang cocok akan dieksekusi. Salah satu kekhususan dari switch-case adalah bahwa nilai ekspresi harus bersifat konstan atau dapat dievaluasi ke dalam nilai konstan.

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

- if-else: Struktur if-else digunakan ketika hanya ada beberapa kemungkinan kondisi yang ingin diuji atau ketika kondisi yang ingin diuji adalah ekspresi boolean. Ini juga digunakan ketika kita perlu mengevaluasi suatu kondisi yang tidak memungkinkan untuk dilakukan dengan switch-case (misalnya, kondisi yang tidak memiliki nilai diskrit).
- switch-case: Struktur switch-case digunakan ketika ada banyak kemungkinan nilai untuk dievaluasi dan ketika ekspresi yang dievaluasi dapat diwakili secara diskrit (seperti bilangan bulat atau karakter). Ini biasanya membuat kode lebih mudah dibaca daripada rangkaian if-else yang panjang. Switch-case juga berguna ketika ingin melakukan tindakan berbeda berdasarkan nilai dari satu variabel atau ekspresi.

3. Pada program 2, tambahkan perintah untuk memilih 2 opsi menggunakan kontrol switch..case.

opsi pilihah 1=inputNilai()

Pilihan 2=inputNilaiBaru()

Kode:

```
E:\TUGAS\PBO\Tugas PBO\Tugas 4\Tugas4_SourceCode_Praktek\TestNilai.java - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
TestNilai.java HitungRata.java
1 // 13020220010, Edo Ranov Anjasmara, Sabtu/23.03.2024, 23:10
2 import java.util.Scanner;
3
4 public class TestNilai {
5     public static void main(String[] args) {
6         HitungRata hitung = new HitungRata();
7         Scanner input = new Scanner(System.in);
8         int banyakData;
9
10        System.out.print("Masukkan Jumlah Data : ");
11        banyakData = input.nextInt();
12        int nilai[] = new int[banyakData];
13
14        // Menu
15        System.out.println("Menu:");
16        System.out.println("1. Input Nilai");
17        System.out.println("2. Input Nilai Baru");
18        System.out.print("Pilih opsi: ");
19        int pilihan = input.nextInt();
20
21        switch (pilihan) {
22            case 1:
23                System.out.print("Masukkan Nilai : ");
24                hitung.inputNilai(nilai);
25                System.out.print("Daftar Nilai : ");
26                hitung.cetakNilai(nilai);
27                System.out.println("Rata Nilai : " + hitung.rataNilai(banyakData));
28                break;
29            case 2:
30                System.out.print("Masukkan Nilai Baru: ");
31                hitung.inputNilaiBaru(banyakData);
32                System.out.print("Daftar Nilai Baru : ");
33                hitung.cetakNilaiBaru();
34                break;
35            default:
36                System.out.println("Pilihan tidak valid!");
37        }
38    }
39 }
40
Jk length: 1.379 lines: 40 Ln: 11 Col: 38 Pos: 378 Windows (CR LF) UTF-8 INS
```

```
E:\TUGAS\PBO\Tugas PBO\Tugas 4\Tugas4_SourceCode_Praktek>java TestNilai
Masukkan Jumlah Data : 3
Menu:
1. Input Nilai
2. Input Nilai Baru
Pilih opsi: 1
Masukkan Nilai : 100
80
95
Daftar Nilai : 100      80      95
Rata Nilai : 91.66666666666667
```

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

Perbedaan antara struktur kontrol perulangan while dan do-while:

- while: Struktur perulangan while mengevaluasi kondisi sebelum menjalankan blok kode di dalamnya. Jika kondisi terpenuhi (true), maka blok kode akan dieksekusi. Jika kondisi awalnya salah (false), maka blok kode tidak akan dieksekusi sama sekali.
- do-while: Struktur perulangan do-while mengevaluasi kondisi setelah menjalankan blok kode di dalamnya. Artinya, blok kode akan dieksekusi sekali sebelum kondisi diuji. Jika kondisi terpenuhi (true) setelah eksekusi pertama, maka blok kode akan terus dieksekusi. Ini memastikan bahwa setidaknya satu iterasi akan dilakukan bahkan jika kondisi awalnya salah.

5. Kapan digunakan struktur kontrol for?

Kapan digunakan struktur kontrol for:

Struktur kontrol for digunakan ketika kita tahu jumlah iterasi yang akan dilakukan sebelumnya atau ketika kita ingin menggunakan loop untuk melakukan iterasi dengan baik melalui array, koleksi, atau rentang bilangan tertentu. Sintaks for juga cocok untuk melakukan inisialisasi, pengujian kondisi, dan peningkatan variabel loop dalam satu baris.

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

Perbedaan antara Array dan ArrayList:

- Array: Array adalah struktur data yang digunakan untuk menyimpan elemen-elemen yang memiliki tipe data yang sama. Ukuran array harus ditentukan saat deklarasi, dan ukurannya tetap setelah deklarasi. Elemen-elemen array dapat diakses menggunakan indeks. Array di Java merupakan struktur data yang memiliki tipe data statis.

Contoh:

```
int[] numbers = new int[5]; // Mendeklarasikan array dengan panjang 5
numbers[0] = 1;
numbers[1] = 2;
// ...
```

- ArrayList: ArrayList adalah kelas di Java yang merupakan implementasi dari antarmuka List dan menggunakan array sebagai penyimpanan data. ArrayList memungkinkan kita untuk menyimpan elemen-elemen dengan tipe data yang sama atau berbeda. Ukuran ArrayList bisa berubah sesuai dengan kebutuhan (dinamis). ArrayList menyediakan metode-metode yang memudahkan manipulasi data seperti penambahan, penghapusan, dan pencarian.

Contoh:

```
ArrayList<String> names = new ArrayList<>();
names.add("John");
names.add("Alice");
// ...
```

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

Kode:

```
1 // 13020220010, Edo Ranov Anjasmara, Sabtu/23.03.2024, 23:30
2 import java.util.HashMap;
3 import java.util.Map;
4 import java.util.Scanner;
5
6 public class ContohHash {
7     public static void main(String[] args) {
8         Scanner scanner = new Scanner(System.in);
9         Map<String, Integer> hashMap = new HashMap<>();
10
11         System.out.println("Masukkan jumlah data yang ingin dimasukkan:");
12         int jumlahData = scanner.nextInt();
13         scanner.nextLine(); // Membersihkan buffer
14
15         // Memasukkan nilai dan kunci (key) ke dalam HashMap
16         for (int i = 0; i < jumlahData; i++) {
17             System.out.println("Masukkan kunci (key) ke-" + (i + 1) + ":");
18             String key = scanner.nextLine();
19             System.out.println("Masukkan nilai ke-" + (i + 1) + ":");
20             int value = scanner.nextInt();
21             scanner.nextLine(); // Membersihkan buffer
22
23             hashMap.put(key, value);
24         }
25
26         // Menampilkan isi HashMap
27         System.out.println("Isi HashMap:");
28         for (Map.Entry<String, Integer> entry : hashMap.entrySet()) {
29             System.out.println("Key: " + entry.getKey() + ", Value: " + entry.getValue());
30         }
31     }
32 }
33
```

```

E:\TUGAS\PBO\Tugas PBO\Tugas 4\Tugas4_SourceCode_Praktek>javac ContohHash.java
E:\TUGAS\PBO\Tugas PBO\Tugas 4\Tugas4_SourceCode_Praktek>java ContohHash
Masukkan jumlah data yang ingin dimasukkan:
3
Masukkan kunci (key) ke-1:
4
Masukkan nilai ke-1:
100
Masukkan kunci (key) ke-2:
5
Masukkan nilai ke-2:
90
Masukkan kunci (key) ke-3:
6
Masukkan nilai ke-3:
80
Isi HashMap:
Key: 4, Value: 100
Key: 5, Value: 90
Key: 6, Value: 80

```

Evaluasi Pratikum 4

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!

Dalam ketiga program di atas, konsep pewarisan dan polimorfisme terlihat pada kelas Mahasiswa.

Pewarisan (Inheritance):

Kelas Mahasiswa merupakan subkelas dari kelas Orang, yang berarti Mahasiswa mewarisi semua atribut dan metode dari kelas Orang.

Pada konstruktor tanpa parameter di kelas Mahasiswa, terdapat pemanggilan `super()` yang mengacu pada konstruktor kelas induk (Orang), sehingga konstruktor kelas induk akan dieksekusi terlebih dahulu sebelum konstruktor kelas Mahasiswa.

Dengan kata lain, kelas Mahasiswa mengambil semua karakteristik yang dimiliki oleh kelas Orang dan menambahkan atribut tambahan stb.

Polimorfisme:

Terdapat dua versi dari konstruktor kelas Mahasiswa: satu tanpa parameter dan satu dengan dua parameter (nama dan stb).

Konsep polimorfisme memungkinkan penggunaan metode dengan nama yang sama di kelas yang berbeda.

Dalam hal ini, metode konstruktor `Mahasiswa()` tanpa parameter adalah contoh polimorfisme, di mana kelas Mahasiswa memiliki versi

konstruktor yang berbeda dari kelas induknya (Orang). Meskipun kelas Mahasiswa mewarisi konstruktor tanpa parameter dari kelas Orang, kelas Mahasiswa juga memiliki versi konstruktor tambahan yang memungkinkan inisialisasi atribut nama dan stb sekaligus.

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

```
public class Orang {
    public String nama;

    public Orang() {
        this.nama= "Aminah";
    }

    public Orang(String nama) {
        this.nama = nama;
    }

    public static void info() {
        System.out.println("Ini adalah informasi dari kelas Orang.");
    }
}

public class Mahasiswa extends Orang {
    private String stb;

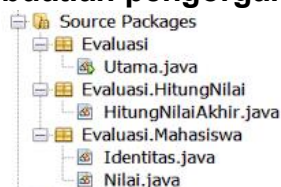
    public Mahasiswa() {
        super();
        this.stb = "1302002134"; //stambuk anda
    }

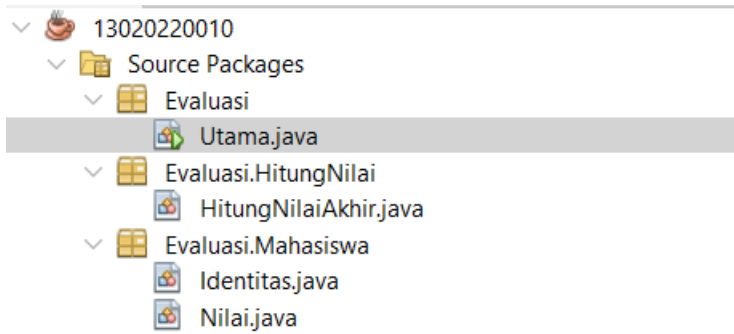
    public Mahasiswa(String stb, String nama) {
        this.nama = nama;
        this.stb = stb;
    }

    public static void info() {
        System.out.println("Ini adalah informasi dari kelas Mahasiswa.");
    }
}

public class Utama {
    public static void main(String[] args) {
        Orang.info(); // Memanggil method info() dari kelas Orang
        Mahasiswa.info(); // Memanggil method info() dari kelas Mahasiswa
    }
}
```

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





```
package Evaluasi;
import Evaluasi.HitungNilai.HitungNilaiAkhir;
import Evaluasi.Mahasiswa.*;

import javax.swing.JOptionPane;

/**
 *
 * @author STAMBUK:13020220010
 * NAMA : EDO RANOV ANJASMARA
 * MATERI : MODUL 4,EVALUASI PRATIKUM
 * WAKTU : RABU 27/03/2024 13:36
 */
public class Utama {

    public static void main(String[] args) {
        String nama = JOptionPane.showInputDialog("Masukkan Nama:");
        String stambuk = JOptionPane.showInputDialog("Masukkan Stambuk:");

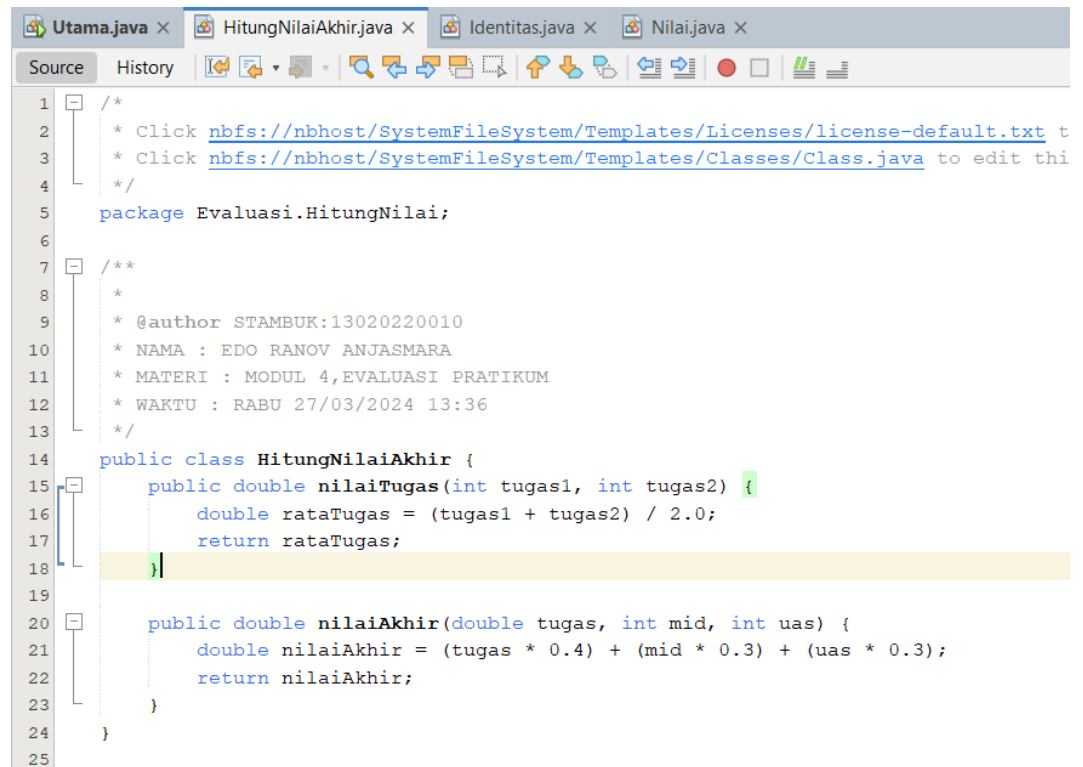
        int tugas1 = Integer.parseInt(JOptionPane.showInputDialog("Masukkan Tugas 1:"));
        int tugas2 = Integer.parseInt(JOptionPane.showInputDialog("Masukkan Tugas 2:"));
        int mid = Integer.parseInt(JOptionPane.showInputDialog("Masukkan Nilai Mid:"));
        int uas = Integer.parseInt(JOptionPane.showInputDialog("Masukkan Nilai UAS:"));

        Identitas mhs = new Identitas();
        mhs.setNama(nama);
        mhs.setStambuk(stambuk);

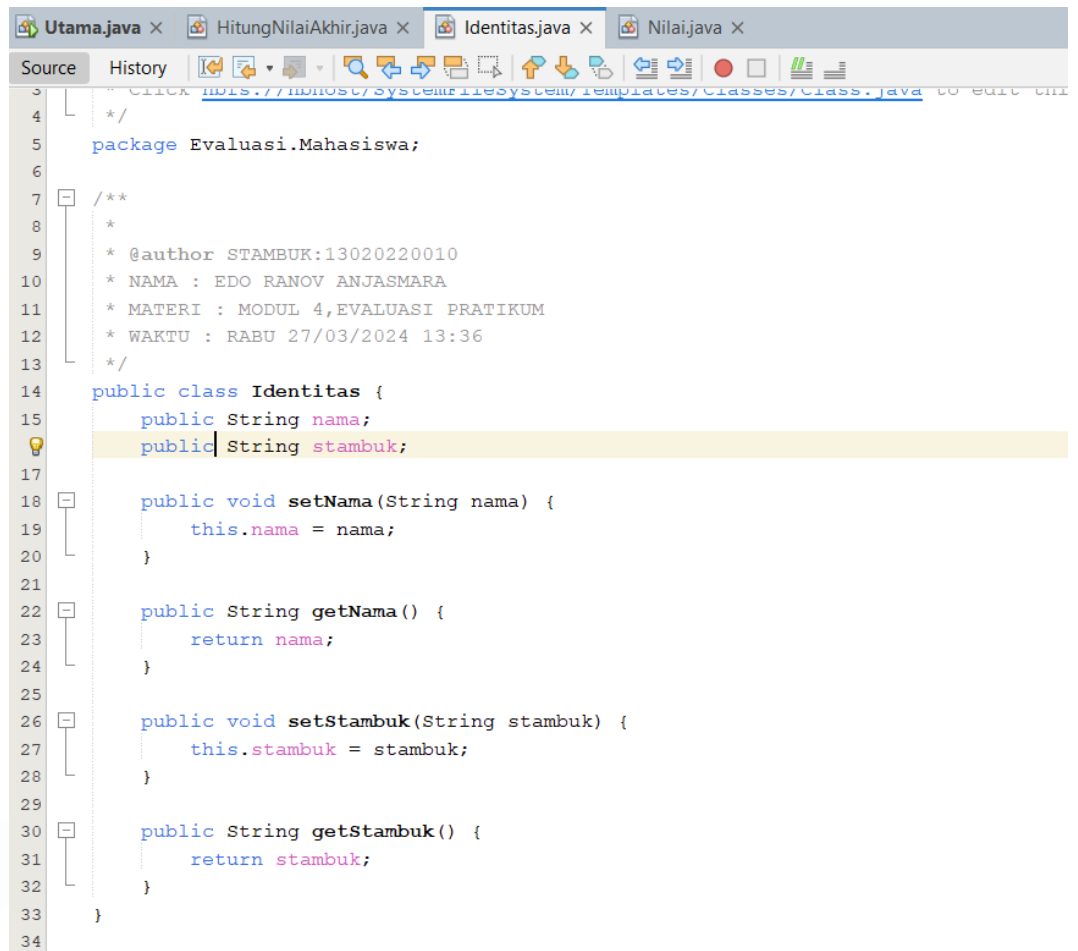
        HitungNilaiAkhir hitungNilai = new HitungNilaiAkhir();
        double tugas = hitungNilai.nilaiTugas(tugas1, tugas2);

        double na = hitungNilai.nilaiAkhir(tugas, mid, uas);

        JOptionPane.showMessageDialog(null, "Nama: " + mhs.getNama() + "\nStambuk: " + mhs.getStambuk());
        JOptionPane.showMessageDialog(null, "Tugas: " + tugas + "\nMid: " + mid + "\nUAS: " + uas + "\nNilai Akhir: " + na);
    }
}
```

```
1  /*
2  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to
3  * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this
4  */
5  package Evaluasi.HitungNilai;
6
7  /**
8   *
9   * @author STAMBUK:13020220010
10   * NAMA : EDO RANOV ANJASMARA
11   * MATERI : MODUL 4,EVALUASI PRATIKUM
12   * WAKTU : RABU 27/03/2024 13:36
13   */
14  public class HitungNilaiAkhir {
15      public double nilaiTugas(int tugas1, int tugas2) {
16          double rataTugas = (tugas1 + tugas2) / 2.0;
17          return rataTugas;
18      }
19
20      public double nilaiAkhir(double tugas, int mid, int uas) {
21          double nilaiAkhir = (tugas * 0.4) + (mid * 0.3) + (uas * 0.3);
22          return nilaiAkhir;
23      }
24  }
25
```



```
3  *  CLICK https://localhost/system/filesystem/templates/classes/class.java to edit this file
4  */
5  package Evaluasi.Mahasiswa;
6
7  /**
8   *
9   * @author STAMBUK:13020220010
10  * NAMA : EDO RANOV ANJASMARA
11  * MATERI : MODUL 4,EVALUASI PRATIKUM
12  * WAKTU : RABU 27/03/2024 13:36
13  */
14  public class Identitas {
15      public String nama;
16      public String stambuk;
17
18      public void setNama(String nama) {
19          this.nama = nama;
20      }
21
22      public String getNama() {
23          return nama;
24      }
25
26      public void setStambuk(String stambuk) {
27          this.stambuk = stambuk;
28      }
29
30      public String getStambuk() {
31          return stambuk;
32      }
33  }
34
```

```
Utama.java x HitungNilaiAkhir.java x Identitas.java x Nilai.java x
Source History
5 package Evaluasi.Mahasiswa;
6
7 /**
8  *
9  * @author STAMBUK:13020220010
10 * NAMA : EDO RANOV ANJASMARA
11 * MATERI : MODUL 4,EVALUASI PRATIKUM
12 * WAKTU : RABU 27/03/2024 13:38
13 */
14 public class Nilai {
15     private int tugas1;
16     private int tugas2;
17     private int mid;
18     private int uas;
19
20
21     public void setTugas1(int tugas1) {
22         this.tugas1 = tugas1;
23     }
24
25     public int getTugas1() {
26         return tugas1;
27     }
28
29     public void setTugas2(int tugas2) {
30         this.tugas2 = tugas2;
31     }
32
33     public int getTugas2() {
34         return tugas2;
35     }
36
37     public void setMid(int mid) {
38         this.mid = mid;
39     }
40
41     public int getMid() {
42         return mid;
43     }
44
45
46     public void setUas(int uas) {
47         this.uas = uas;
48     }
49
50     // Getter untuk variabel uas
51     public int getUas() {
52         return uas;
53     }
54 }
55
```

```
hitungNilaiAkhir hitungNilai = new HitungNilaiAkhir();  
le tugas = hitungNilai.nilaiTugas(tugas1, tugas2);  
  
le na = hitungNilai.nilaiAkhir(tugas, mid, uas);
```

ionPane.show Input X .getNama() + "\nStambu
ionPane.show ? Masukkan Nama:
Edo Ranov Anjasmara
OK Cancel

```
a = hitungNilai.nilaiAkhir(tugas, mid, uas);
```

ane.show Input X .getNama() + "\n
ane.show ? Masukkan Stambuk:
13020220010
OK Cancel

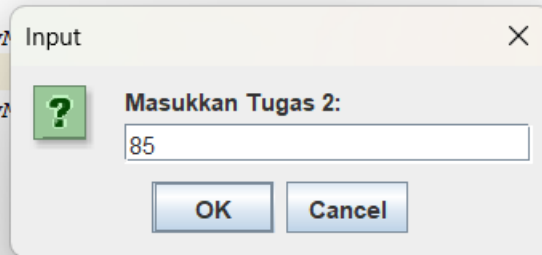
```
= hitungNilai.nilaiAkhir(tugas, mid, uas);
```

ne.show Input X .getNama()
ne.show ? Masukkan Tugas 1:
99
OK Cancel

```
le tugas = hitungNilai.nilaiTugas(tugas1, tugas2);
```

```
le na = hitungNilai.nilaiAkhir(tugas, mid, uas);
```

```
ionPane.showModal();  
ionPane.showModal();
```



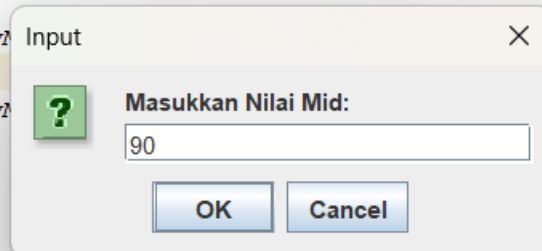
Input

Masukkan Tugas 2:

OK Cancel

```
na = hitungNilai.nilaiAkhir(tugas, mid, uas);
```

```
.Pane.showModal();  
.Pane.showModal();
```



Input

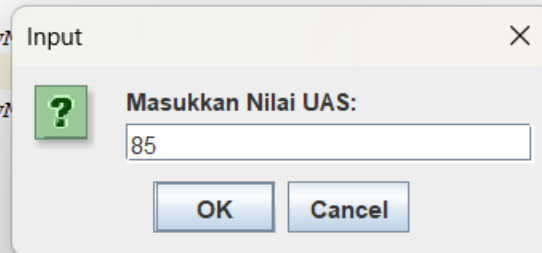
Masukkan Nilai Mid:

OK Cancel

```
double tugas = hitungNilai.nilaiTugas(tugas1, tugas2);
```

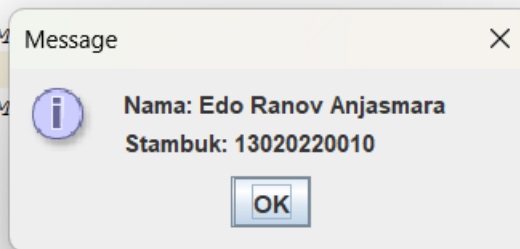
```
double na = hitungNilai.nilaiAkhir(tugas, mid, uas);
```

```
OptionPane.showMessageDialog(null, "Nama: " + nama + "\nStar  
tipe: " + tugas + "\nMid: " + mid + "\nUAS: " + uas + "\nNilai Akhir: " + na);
```



```
double na = hitungNilai.nilaiAkhir(tugas, mid, uas);
```

```
OptionPane.showMessageDialog(null, "Nama: " + nama + "\nStar  
tipe: " + tugas + "\nMid: " + mid + "\nUAS: " + uas + "\nNilai Akhir: " + na);
```



```
double na = hitungNilai.nilaiAkhir(tugas, mid, uas);
```

```
OptionPane.showMessageDialog(null, "Nama: " + nama + "\nStar  
tipe: " + tugas + "\nMid: " + mid + "\nUAS: " + uas + "\nNilai Akhir: " + na);
```

