

Tugas 2

Makassar, 27 Februari 2024

**LAPORAN  
PEMROGRAMAN BERORIENTASI OBJEK**



Nama : Narendra Awangga  
Stambuk : 13020220022  
Kelas : A1  
Dosen : Mardiyah Hasnawi, S.Kom.,M.T.

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS MUSLIM INDONESIA  
MAKASSAR  
2023**

1. Membuat aplikasi bahasa Java memasukkan dan menampilkan nim, nama, jurusan dan fakultas anda, data tersebut dimasukkan melalui keyboard!

### 1. Menggunakan Scanner

```
import java.util.Scanner;

public class Scan{
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.println("Masukkan NIM:");
        String nim = scanner.nextLine();

        System.out.println("Masukkan nama:");
        String nama = scanner.nextLine();

        System.out.println("Masukkan jurusan:");
        String jurusan = scanner.nextLine();

        System.out.println("Masukkan fakultas:" );
        String fakultas = scanner.nextLine();

        System.out.println("\n");

        System.out.println("Data yang dimasukkan: ");
        System.out.println("NIM: " + nim);
        System.out.println("Nama: " + nama);
        System.out.println("Jurusan: " + jurusan);
        System.out.println("Fakultas: " + fakultas);

        scanner.close();
    }
}
```

- **Penjelasan program**

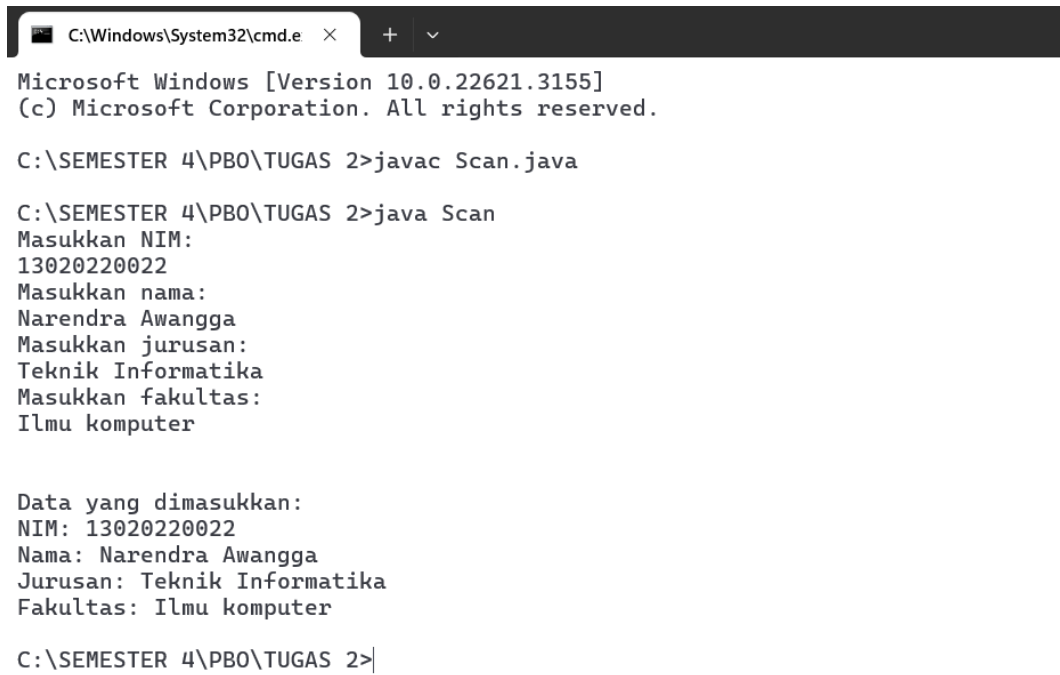
Program tersebut adalah sebuah program Java sederhana yang bertujuan untuk menerima input data dari pengguna menggunakan kelas **Scanner** dan mencetak kembali data yang dimasukkan tersebut.

Penjelasan setiap bagian program:

1. **import java.util.Scanner;**: Ini adalah pernyataan import yang mengimpor kelas **Scanner** dari paket **java.util**. Kelas **Scanner** digunakan untuk membaca input dari pengguna melalui konsol.
2. **public class Scan{**: Ini adalah deklarasi kelas utama yang disebut **Scan**.
3. **public static void main(String[] args) {**: Ini adalah metode utama (main method) di mana eksekusi program dimulai. Metode ini menerima argumen dalam bentuk array string.
4. **Scanner scanner = new Scanner(System.in);**: Ini membuat sebuah objek **Scanner** yang disebut **scanner** untuk membaca input dari **System.in**, yang mewakili input dari konsol.
5. **System.out.println("Masukkan NIM:");**: Ini mencetak pesan ke konsol meminta pengguna untuk memasukkan NIM.
6. **String nim = scanner.nextLine();**: Ini membaca input dari pengguna (NIM) menggunakan metode **nextLine()** dari objek **Scanner** dan menyimpannya dalam variabel **nim** yang memiliki tipe data **String**.
7. Langkah 5-6 diulang untuk meminta dan membaca input untuk nama, jurusan, dan fakultas.
8. **System.out.println("\n");**: Ini mencetak baris kosong untuk memisahkan input pengguna dari output.
9. Mencetak kembali data yang dimasukkan oleh pengguna dengan pesan yang sesuai menggunakan **System.out.println()**.
10. **scanner.close();**: Ini menutup objek **Scanner** setelah selesai membaca input dari pengguna untuk menghindari kebocoran sumber daya.

Tujuan program ini adalah untuk memungkinkan pengguna memasukkan data seperti NIM, nama, jurusan, dan fakultas, dan kemudian mencetak kembali data tersebut sebagai konfirmasi. Program ini menggunakan kelas **Scanner** untuk membaca input pengguna dan mencetak output menggunakan **System.out.println()**.

- **Output :**



```
C:\Windows\System32\cmd.e  X + v
Microsoft Windows [Version 10.0.22621.3155]
(c) Microsoft Corporation. All rights reserved.

C:\SEMESTER 4\PBO\TUGAS 2>javac Scan.java

C:\SEMESTER 4\PBO\TUGAS 2>java Scan
Masukkan NIM:
13020220022
Masukkan nama:
Narendra Awangga
Masukkan jurusan:
Teknik Informatika
Masukkan fakultas:
Ilmu komputer

Data yang dimasukkan:
NIM: 13020220022
Nama: Narendra Awangga
Jurusan: Teknik Informatika
Fakultas: Ilmu komputer

C:\SEMESTER 4\PBO\TUGAS 2>|
```

## 2. Menggunakan Class BufferedReader

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

public class BuffR{
    public static void main(String[] args) throws
IOException {
        BufferedReader reader = new BufferedReader(new
InputStreamReader(System.in));

        System.out.print("Masukkan NIM: ");
        String nim = reader.readLine();

        System.out.print("Masukkan nama: ");
        String nama = reader.readLine();

        System.out.print("Masukkan jurusan: ");
        String jurusan = reader.readLine();

        System.out.print("Masukkan fakultas: ");
        String fakultas = reader.readLine();

        System.out.println("\n");

        System.out.println("Data yang dimasukkan:");
        System.out.println("NIM: " + nim);
        System.out.println("Nama: " + nama);
        System.out.println("Jurusan: " + jurusan);
        System.out.println("Fakultas: " + fakultas);

        reader.close();
    }
}
```

- **Penjelasan program**

Program di atas merupakan program Java yang mengambil input dari pengguna menggunakan kelas **BufferedReader** dan mencetak kembali data yang dimasukkan oleh pengguna.

Penjelasan setiap bagian program:

1. **import java.io.BufferedReader;**, **import java.io.IOException;**, **import java.io.InputStreamReader;**: Ini adalah pernyataan import yang mengimpor kelas-kelas yang dibutuhkan dari paket **java.io** untuk penggunaan **BufferedReader**, **IOException**, dan **InputStreamReader**.
2. **public class BuffR{**: Ini adalah deklarasi kelas utama yang disebut **BuffR**.
3. **public static void main(String[] args) throws IOException {**: Ini adalah metode utama (main method) di mana eksekusi program dimulai. Metode ini melemparkan **IOException** untuk menangani kesalahan masukan/keluaran yang mungkin terjadi saat menggunakan **BufferedReader**.
4. **BufferedReader reader = new BufferedReader(new InputStreamReader(System.in));**: Ini membuat objek **BufferedReader** yang disebut **reader** untuk membaca input dari **System.in** (konsol) menggunakan **InputStreamReader**.
5. **System.out.print("Masukkan NIM: ");**: Ini mencetak pesan ke konsol meminta pengguna untuk memasukkan NIM tanpa baris baru (tanpa enter).
6. **String nim = reader.readLine();**: Ini membaca input dari pengguna (NIM) menggunakan metode **readLine()** dari objek **BufferedReader** dan menyimpannya dalam variabel **nim** yang memiliki tipe data **String**.
7. Langkah 5-6 diulang untuk meminta dan membaca input untuk nama, jurusan, dan fakultas.
8. **System.out.println("\n");**: Ini mencetak baris kosong untuk memisahkan input pengguna dari output.
9. Mencetak kembali data yang dimasukkan oleh pengguna dengan pesan yang sesuai menggunakan **System.out.println()**.
10. **reader.close();**: Ini menutup objek **BufferedReader** setelah selesai membaca input dari pengguna untuk menghindari kebocoran sumber daya.

Tujuan program ini mirip dengan program sebelumnya, yaitu memungkinkan pengguna untuk memasukkan data seperti NIM, nama, jurusan, dan fakultas, dan kemudian mencetak kembali data tersebut sebagai konfirmasi. Program ini menggunakan kelas `BufferedReader` untuk membaca input pengguna dan mencetak output menggunakan `System.out.println()`.

- **Output**



```
C:\Windows\System32\cmd.e  X  +  v
Microsoft Windows [Version 10.0.22621.3155]
(c) Microsoft Corporation. All rights reserved.

C:\SEMESTER 4\PBO\TUGAS 2>javac BuffR.java

C:\SEMESTER 4\PBO\TUGAS 2>java BuffR
Masukkan NIM: 13020220022
Masukkan nama: Narendra Awangga
Masukkan jurusan: Teknik informatika
Masukkan fakultas: Ilmu komputer

Data yang dimasukkan:
NIM: 13020220022
Nama: Narendra Awangga
Jurusan: Teknik informatika
Fakultas: Ilmu komputer

C:\SEMESTER 4\PBO\TUGAS 2>|
```

### 3. Menggunakan Class JOptionPane.

```
import javax.swing.JOptionPane;

public class JOpti{
    public static void main(String[] args) {
        String nim =
JOptionPane.showInputDialog("Masukkan NIM:");
        String nama =
JOptionPane.showInputDialog("Masukkan nama:");
        String jurusan =
JOptionPane.showInputDialog("Masukkan jurusan:");
        String fakultas =
JOptionPane.showInputDialog("Masukkan fakultas:");

        String message = "Data yang dimasukkan:\n" +
            "NIM: " + nim + "\n" +
            "Nama: " + nama + "\n" +
            "Jurusan: " + jurusan + "\n" +
            "Fakultas: " + fakultas;

        JOptionPane.showMessageDialog(null, message);
    }
}
```

- **Penjelasan program**

Program ini bertujuan untuk mengambil input dari pengguna berupa NIM, nama, jurusan, dan fakultas mahasiswa menggunakan dialog input JOptionPane, dan kemudian menampilkan data yang dimasukkan kembali kepada pengguna menggunakan dialog message JOptionPane.

**Penjelasan setiap bagian program:**

1. **import javax.swing.JOptionPane;:** Pernyataan import untuk mengimpor kelas **JOptionPane** dari paket **javax.swing**, yang akan digunakan untuk menampilkan dialog input dan message.



2. **public class JOpti{**: Deklarasi kelas utama yang disebut **JOpti**.
3. **public static void main(String[] args) {**: Metode utama (main method) di mana eksekusi program dimulai.
4. **String nim = JOptionPane.showInputDialog("Masukkan NIM:");**: Menggunakan metode **showInputDialog()** dari kelas **JOptionPane** untuk menampilkan dialog input dan meminta pengguna memasukkan NIM. Nilai yang dimasukkan oleh pengguna kemudian disimpan dalam variabel **nim** yang memiliki tipe data String.
5. **Langkah 4 diulang untuk meminta dan menyimpan input pengguna untuk nama, jurusan, dan fakultas.**
6. **String message = "Data yang dimasukkan:\n" + ... + fakultas;**: Membangun sebuah pesan yang berisi data yang dimasukkan oleh pengguna, dengan menggunakan nilai yang disimpan dalam variabel **nim**, **nama**, **jurusan**, dan **fakultas**. Pesan ini akan ditampilkan kepada pengguna sebagai konfirmasi.
7. **JOptionPane.showMessageDialog(null, message);**: Menggunakan metode **showMessageDialog()** dari kelas **JOptionPane** untuk menampilkan pesan yang dibuat sebelumnya kepada pengguna dalam sebuah dialog message.
8. **Program selesai.**

**2. Buat Program menggunakan bahasa java untuk Konversi Waktu (Jam:Menit:Detik) dari masukan/input detik!**

```
import java.util.Scanner;

public class Waktu {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Masukkan total detik: ");
        int totalDetik = scanner.nextInt();
        int detikSekarang = totalDetik % 60;
        int totalMenit = totalDetik / 60;
        int menitSekarang = totalMenit % 60;
        int totalJam = totalMenit / 60;
        int jamSekarang = totalJam % 24;

        System.out.println("Waktu sekarang: " +
            jamSekarang + ":" + menitSekarang + ":" +
            detikSekarang);

        scanner.close();
    }
}
```

Program di atas memiliki tujuan untuk mengonversi jumlah detik yang dimasukkan oleh pengguna menjadi waktu dalam format jam:menit:detik, kemudian mencetak waktu tersebut.

Berikut adalah penjelasan setiap bagian dari program tersebut:

1. **import java.util.Scanner;**: Ini adalah pernyataan import yang mengimpor kelas Scanner dari paket java.util. Kelas Scanner digunakan untuk membaca input dari pengguna.

2. **public class Waktu {**: Program ini berada dalam kelas yang disebut Waktu. Ini adalah deklarasi kelas utama.
3. **public static void main(String[] args) {**: Ini adalah metode utama yang dieksekusi ketika program dimulai. Semua kode program yang sebenarnya berada di dalam metode ini.
4. **Scanner scanner = new Scanner(System.in);**: Membuat objek Scanner yang digunakan untuk membaca input dari pengguna melalui konsol.
5. **System.out.print("Masukkan total detik: ");**: Mencetak pesan ke konsol untuk meminta pengguna memasukkan jumlah total detik.
6. **int totalDetik = scanner.nextInt();**: Menggunakan objek Scanner untuk mengambil input pengguna sebagai bilangan bulat dan menyimpannya dalam variabel **totalDetik**.
7. **int detikSekarang = totalDetik % 60;**: Menghitung sisa detik setelah membagi **totalDetik** dengan 60, yang akan memberikan detik saat ini.
8. **int totalMenit = totalDetik / 60;**: Menghitung total menit dengan membagi **totalDetik** dengan 60.
9. **int menitSekarang = totalMenit % 60;**: Menghitung sisa menit setelah membagi **totalMenit** dengan 60, yang akan memberikan menit saat ini.
10. **int totalJam = totalMenit / 60;**: Menghitung total jam dengan membagi **totalMenit** dengan 60.
11. **int jamSekarang = totalJam % 24;**: Menghitung sisa jam setelah membagi **totalJam** dengan 24, yang akan memberikan jam saat ini.
12. **System.out.println("Waktu sekarang: " + jamSekarang + ":" + menitSekarang + ":" + detikSekarang);**: Mencetak waktu saat ini (jam, menit, dan detik) ke konsol.
13. **scanner.close();**: Ini menutup objek Scanner setelah selesai digunakan untuk membersihkan sumber daya.