**LAPORAN PRAKTIKUM**

**PEMROGRAMAN II**

**MODUL 6**

****

**Graphical User Interfce (GUI)**

**Oleh:**

**Muhammad Ryan Rizky Rahmadi NIM. 2210817310001**

**PROGRAM STUDI TEKNOLOGI INFORMASI**

**FAKULTAS TEKNIK**

**UNIVERSITAS LAMBUNG MANGKURAT**

**DESEMBER 2023**

# LEMBAR PENGESAHAN

**LAPORAN PRAKTIKUM PEMROGRAMAN II**

**MODUL 6**

Laporan Praktikum Pemrograman II Modul 6 : Grapichal User Interface ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman II. Laporan Prakitkum ini dikerjakan oleh:

Nama Praktikan : Muhammad Ryan Rizky Rahmadi

NIM : 2210817310001

|  |  |
| --- | --- |
| Menyetujui,  Asisten Praktikum  Bachrul Uluum  NIM. 2010817210025 | Mengetahui,  Dosen Penanggung Jawab Praktikum  Andreyan Rizky Baskara, S.Kom., M.Kom.  NIP. 19930703 201903 1 011 |

# DAFTAR ISI

[LEMBAR PENGESAHAN 1](#_Toc153352739)

[DAFTAR ISI 2](#_Toc153352740)

[DAFTAR GAMBAR 3](#_Toc153352741)

[DAFTAR TABEL 4](#_Toc153352742)

[SOAL 1 5](#_Toc153352743)

[A. Source Code 7](#_Toc153352744)

[B. Output Program 10](#_Toc153352745)

[C. Pembahasan 10](#_Toc153352746)

[a. Mahasiswa 10](#_Toc153352747)

[b. Main 11](#_Toc153352748)

[LINK GIT HUB 11](#_Toc153352749)

# DAFTAR GAMBAR

[Gambar 1.1 Screenshoot Output Soal 1 10](#_Toc152090499)

# DAFTAR TABEL

[Tabel 2.1 Source Code Soal 1 Spehere 8](#_Toc153351591)

[Tabel 3.1 Source Code Soal 1 Shape 9](#_Toc153351592)

# SOAL 1

Diberikan class diagram seperti berikut: (isi program harus sesuai dengan class diagram)



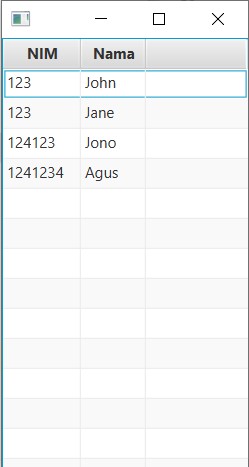
Implementasikan class diagram diatas menjadi class pada Bahasa pemrograman java. Class mahasiswa harus menerapkan setter dan getter.

Program harus menampilkan list data dalam bentuk tabel.

Kolom dari tabel adalah :

* NIM
* Nama

Kemudian buatlah 10 data secara *hardcode* untuk ditampilkan pada tabel Contoh program dapat dilihat sebagai berikut:



## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  25  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42 | package com.example.modul\_6;  import javafx.fxml.FXML;  import javafx.scene.control.Label;  public class Mahasiswa {  private int id;  private String name;  private String nim;  public Mahasiswa(int id, String name, String nim) {  this.id = id;  this.name = name;  this.nim = nim;  }  public int getId() {  return id;  }  public void setId(int id) {  this.id = id;  }  public String getName() {  return name;  }  public void setName(String name) {  this.name = name;  }  public String getNim() {  return nim;  }  public void setNim(String nim) {  this.nim=nim;  }  } |

Tabel 1.1 Source Code Soal 1 Mahasiswa

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74 | package com.example.modul\_6;  import javafx.application.Application;  import javafx.scene.Scene;  import javafx.scene.control.TableColumn;  import javafx.scene.control.TableView;  import javafx.scene.control.cell.PropertyValueFactory;  import javafx.scene.layout.VBox;  import javafx.stage.Stage;  public class Main extends Application {  public static void main(String[] args) {  launch(args);  }  @Override  public void start(Stage primaryStage) {  TableView tableView = new TableView();  TableColumn<Mahasiswa, String> column1 =  new TableColumn<>("NIM");  column1.setCellValueFactory(  new PropertyValueFactory<>("nim"));  TableColumn<Mahasiswa, String> column2 =  new TableColumn<>("Nama");  column2.setCellValueFactory(  new PropertyValueFactory<>("name"));  tableView.getColumns().add(column1);  tableView.getColumns().add(column2);  tableView.getItems().add(  new Mahasiswa(1, "Audrey", "7000"));  tableView.getItems().add(  new Mahasiswa(2, "Dandellion", "7001"));  tableView.getItems().add(  new Mahasiswa(3, "Deyya", "7002"));  tableView.getItems().add(  new Mahasiswa(4, "Aina", "7003"));  tableView.getItems().add(  new Mahasiswa(5, "Dawan", "7004"));  tableView.getItems().add(  new Mahasiswa(6, "Helenah", "7005"));  tableView.getItems().add(  new Mahasiswa(7, "Lidama", "7006"));  tableView.getItems().add(  new Mahasiswa(8, "Aninda", "7007"));  tableView.getItems().add(  new Mahasiswa(9, "Isanhabang", "7008"));  tableView.getItems().add(  new Mahasiswa(10, "IrfanStis", "7009"));  VBox vbox = new VBox(tableView);  Scene scene = new Scene(vbox);  primaryStage.setScene(scene);  primaryStage.show();  }  } |

Tabel 2.1 Source Code Soal 1 Main

## Output Program



Gambar 1.1 Screenshoot Output Soal 1

## Pembahasan

### Mahasiswa

Line 1 : *Package,* untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3 dan 4: *Import,*  suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 7 : *int,* tipe data *integer* atau menerima bilangan bulat.

Line 8 : *String*, tipe data teks yang berisikan karakter lebih.

Line 9 : *String*, tipe data teks yang berisikan karakter lebih.

Line 18, 22, 26,30, 35, dan 38 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 6 dan 42: {}, penanda awal dan akhir *block code*.*.*

### Main

Line 1 : *Package,* untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3 – 10 : *Import,*  suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 12 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 12 : *Extends ,*  kata kunci yang ditulis dengan kelas anak selama deklarasi kelas diikuti dengan nama kelas induk.

Line 12 dan 72 : {}, penanda awal dan akhir *block code*.

.

# LINK GIT HUB

<https://github.com/RylenRawr/PEMROGRAMAN-II/tree/main/MODUL%206>