



JOptionPane, JScrollPane, JTable

Praktikum Pemrograman II - 06



JOptionPane

Kelas JOptionPane digunakan untuk menyediakan kotak dialog standar seperti kotak dialog pesan, kotak dialog konfirmasi dan kotak dialog input. Kotak dialog ini dapat digunakan untuk menampilkan informasi atau mendapatkan inputan dari pengguna. Kelas JOptionPane merupakan turunan dari kelas JComponent.

Kelas JOptionPane menyediakan kotak dialog menggunakan method-method static, berikut adalah daftar method yang dapat kita gunakan.

Tabel 1 Daftar Method JOptionPane

Method	Deskripsi
static void showMessageDialog(Component parentComponent, Object message)	Method ini digunakan untuk membuat dialog untuk menampilkan pesan dengan judul default "Message".
static void showMessageDialog(Component parentComponent, Object message, String title, int messageType)	Method ini digunakan untuk membuat dialog untuk menampilkan pesan dengan judul yang ditentukan.
static int showConfirmDialog(Component parentComponent, Object message)	Method ini digunakan untuk membuat dialog dengan opsi Yes, No, dan Cancel; dengan judul default Select an Option.
static int showConfirmDialog(Component parentComponent, Object message, String title, int option)	Method ini digunakan untuk membuat dialog dengan opsi yang dapat dipilih dari konstanta yang disediakan dan judul yang dapat ditentukan sendiri.
static String showInputDialog(Component parentComponent, Object message)	Method digunakan untuk menampilkan dialog pertanyaan yang meminta input dari pengguna.

Latihan 1 – showMessageDialog

Buatlah kelas HelloMessageDialog dengan kode sebagai berikut

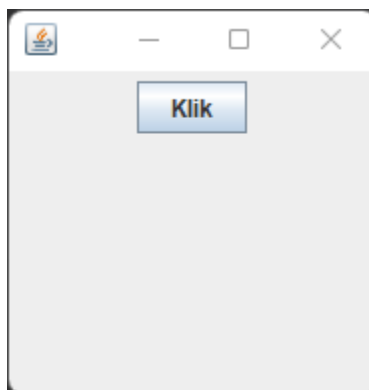
```

J HelloMessageDialog.java
1  import java.awt.*;
2  import java.awt.event.*;
3  import javax.swing.*;
4
5  public class HelloMessageDialog extends JFrame {
6
7      public HelloMessageDialog(){
8          this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
9          this.setLayout(new FlowLayout());
10
11         JButton button = new JButton("Klik");
12         button.setBounds(130,100,100, 40);
13
14         button.addActionListener(new ActionListener(){
15             public void actionPerformed(ActionEvent e){
16                 JOptionPane.showMessageDialog(HelloMessageDialog.this, "Halo, Selamat Datang di Praktikum Pemrograman II.");
17             }
18         });
19
20         this.add(button);
21
22         this.setSize(200,200);
23     }
24
25     public static void main(String[] args) {
26         javax.swing.SwingUtilities.invokeLater(new Runnable() {
27             public void run() {
28                 HelloMessageDialog h = new HelloMessageDialog();
29                 h.setVisible(true);
30             }
31         });
32     }
33 }
34

```

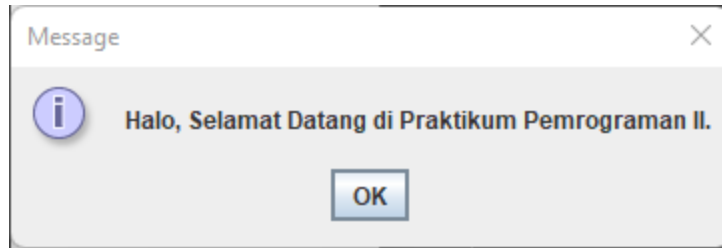
Gambar 1 Kode Kelas HelloMessageDialog

Kompilasi kemudian jalankan kelas tersebut, anda akan melihat hasil seperti gambar 2 berikut ini.



Gambar 2 Hasil Run Kelas HelloMessageDialog

Tekan tombol dengan tulisan Klik, maka akan muncul sebuah kotak dialog seperti pada gambar 3 berikut.



Gambar 3 Kotak Dialog Pesan

Latihan 2 – showMessageDialog

Cobalah ubah judul dan jenis dialog pesan dari latihan 1 dengan menggunakan method yang menerima parameter title dan messageType!

Berikut ini adalah konstanta yang diterima untuk parameter messageType:

- JOptionPane.ERROR_MESSAGE
- JOptionPane.INFORMATION_MESSAGE
- JOptionPane.WARNING_MESSAGE
- JOptionPane.QUESTION_MESSAGE
- JOptionPane.PLAIN_MESSAGE

Latihan 3 – showConfirmDialog

Buatlah kelas HelloConfirmDialog dengan kode sebagai berikut.

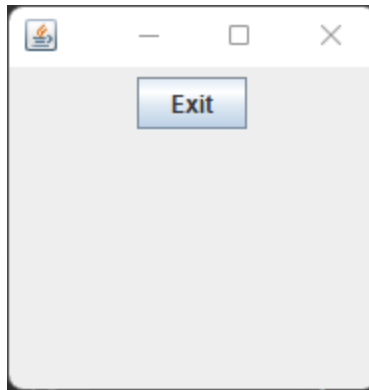
```

J HelloConfirmDialog.java
1  import java.awt.*;
2  import java.awt.event.*;
3  import javax.swing.*;
4
5  public class HelloConfirmDialog extends JFrame {
6
7      public HelloConfirmDialog(){
8          this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
9          this.setLayout(new FlowLayout());
10
11         JButton button = new JButton("Exit");
12         button.setBounds(130,100,100, 40);
13
14         button.addActionListener(new ActionListener(){
15             public void actionPerformed(ActionEvent e){
16                 int confirmation = JOptionPane.showConfirmDialog(HelloConfirmDialog.this,
17                     "Apakah anda yakin ingin keluar dari aplikasi?", "Konfirmasi", JOptionPane.YES_NO_OPTION);
18                 if (confirmation == JOptionPane.YES_OPTION)
19                 {
20                     System.exit(0);
21                 } else
22                 {
23                     JOptionPane.showMessageDialog(HelloConfirmDialog.this, "Anda menekan tombol NO");
24                 }
25             }
26         });
27
28         this.add(button);
29
30         this.setSize(200,200);
31     }
32
33     public static void main(String[] args) {
34         javax.swing.SwingUtilities.invokeLater(new Runnable() {
35             public void run() {
36                 HelloConfirmDialog h = new HelloConfirmDialog();
37                 h.setVisible(true);
38             }
39         });
40
41     }
42 }

```

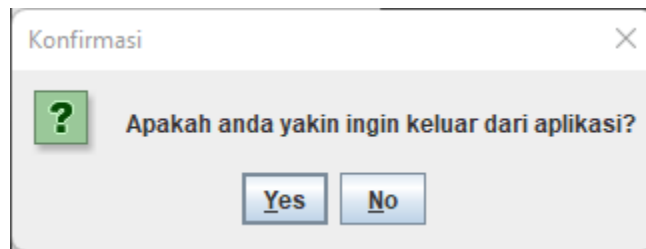
Gambar 4 Kode HelloConfirmDialog

Kompilasi dan jalankan kelas tersebut, maka akan muncul jendela seperti pada gambar 5.



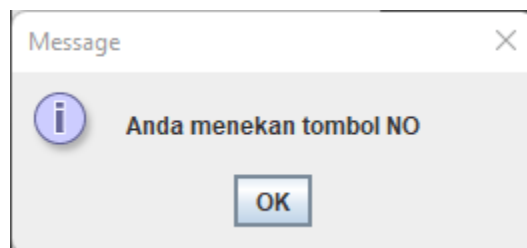
Gambar 5 Hasil Run Kelas HelloConfirmDialog

Ketika tombol Exit diklik, akan muncul sebuah dialog konfirmasi seperti pada gambar 6 berikut.



Gambar 6 Dialog Konfirmasi Exit

Jika kita klik tombol No, maka akan muncul dialog seperti pada gambar 7, sedangkan tombol Yes akan mengeluarkan kita dari aplikasi.



Gambar 7 Dialog Informasi Tombol No

Latihan 4 – `showInputDialog`

Buatlah kelas `HelloInputDialog` dengan kode sebagai berikut.

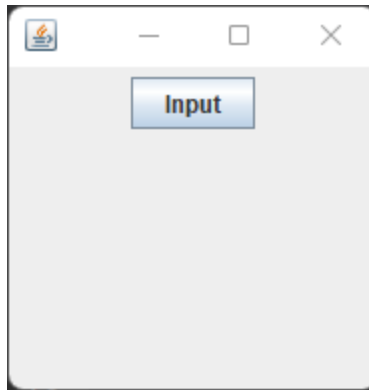
```

J HelloInputDialog.java
1  import java.awt.*;
2  import java.awt.event.*;
3  import javax.swing.*;
4
5  public class HelloInputDialog extends JFrame {
6
7      public HelloInputDialog(){
8          this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
9          this.setLayout(new FlowLayout());
10
11         JButton button = new JButton("Input");
12         button.setBounds(130,100,100, 40);
13
14         button.addActionListener(new ActionListener(){
15             public void actionPerformed(ActionEvent e){
16                 String nama = JOptionPane.showInputDialog(HelloInputDialog.this,
17                     "Inputkan nama anda");
18                 JOptionPane.showMessageDialog(HelloInputDialog.this, "Hello, "+nama);
19             }
20         });
21
22         this.add(button);
23
24         this.setSize(200,200);
25     }
26
27     public static void main(String[] args) {
28         javax.swing.SwingUtilities.invokeLater(new Runnable() {
29             public void run() {
30                 HelloInputDialog h = new HelloInputDialog();
31                 h.setVisible(true);
32             }
33         });
34     }
35 }
36

```

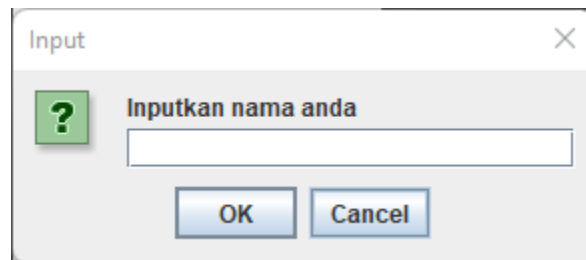
Gambar 8 Kode HelloInputDialog

Kompilasi dan Jalankan kelas HelloInputDialog yang akan memunculkan jendela seperti pada gambar 9 berikut.



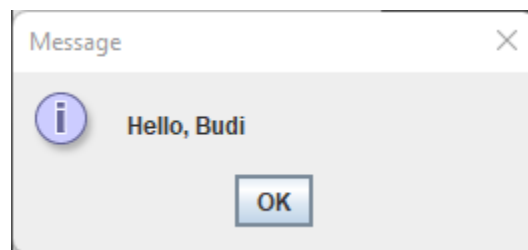
Gambar 9 Hasil Run HelloInputDialog

Ketika tombol Input diklik, akan muncul sebuah kotak dialog input seperti pada gambar 10.



Gambar 10 Dialog Input

Inputkan sebuah nama kemudian klik OK, akan muncul sebuah dialog seperti pada gambar 11.



Gambar 11 Dialog Pesan

JScrollPane

JScrollPane digunakan untuk membuat tampilan komponen yang dapat digulirkan. Ketika ukuran layar terbatas, kita menggunakan JScrollPane untuk menampilkan komponen yang besar atau komponen yang ukurannya dapat berubah secara dinamis.

Latihan 5 – ScrollPane

Kembali ke latihan mengenai TextArea, kita akan tambahkan penggunaan ScrollPane untuk menggulirkan TextArea yang menjadi komponen output. Buatlah sebuah kelas bernama HelloScrollPane kemudian isi dengan kode HelloTextArea yang sudah dimodifikasi sehingga menjadi seperti berikut.

```
J HelloScrollPane.java
1  import java.awt.event.*;
2  import javax.swing.*;
3
4  public class HelloScrollPane extends JFrame {
5
6      public HelloScrollPane(){
7          this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
8
9          JLabel labelInput = new JLabel("Input Nama:");
10         labelInput.setBounds(130,40,100,10);
11
12         JTextField textField = new JTextField();
13         textField.setBounds(130,60,100,30);
14
15         JButton button = new JButton("Klik");
16         button.setBounds(130,100,100, 40);
17
18         JTextArea txtOutput = new JTextArea("");
19         JScrollPane scrollableTextArea = new JScrollPane(txtOutput);
20         scrollableTextArea.setBounds(130,150,100,100);
21         scrollableTextArea.setHorizontalScrollBarPolicy(JScrollPane.HORIZONTAL_SCROLLBAR_ALWAYS);
22         scrollableTextArea.setVerticalScrollBarPolicy(JScrollPane.VERTICAL_SCROLLBAR_ALWAYS);
23
24         button.addActionListener(new ActionListener(){
25             public void actionPerformed(ActionEvent e){
26                 String nama = textField.getText();
27                 txtOutput.append("Hello "+nama+"\n");
28                 textField.setText("");
29             }
30         });
31
32         this.add(button);
33         this.add(textField);
34         this.add(labelInput);
35         this.add(scrollableTextArea);
36
37         this.setSize(400,500);
38         this.setLayout(null);
39     }
40 }
```

Gambar 12 Kode HelloScrollPane Bagian 1

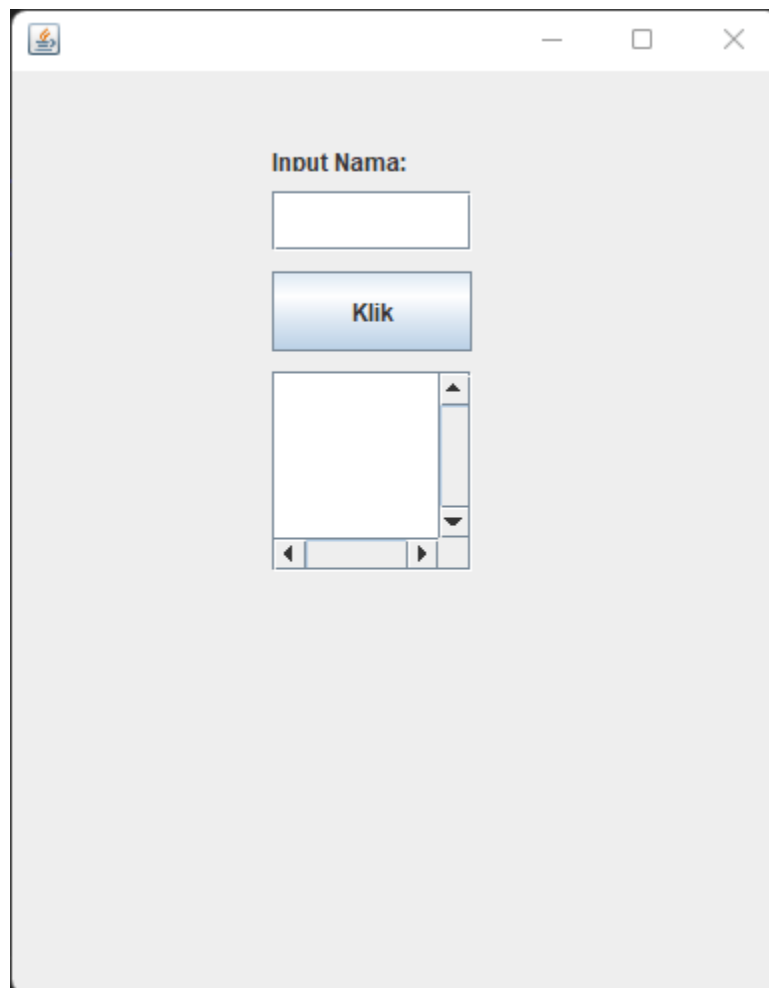
```

40
41     public static void main(String[] args) {
42         javax.swing.SwingUtilities.invokeLater(new Runnable() {
43             public void run() {
44                 HelloScrollPane h = new HelloScrollPane();
45                 h.setVisible(true);
46             }
47         });
48     }
49 }
50

```

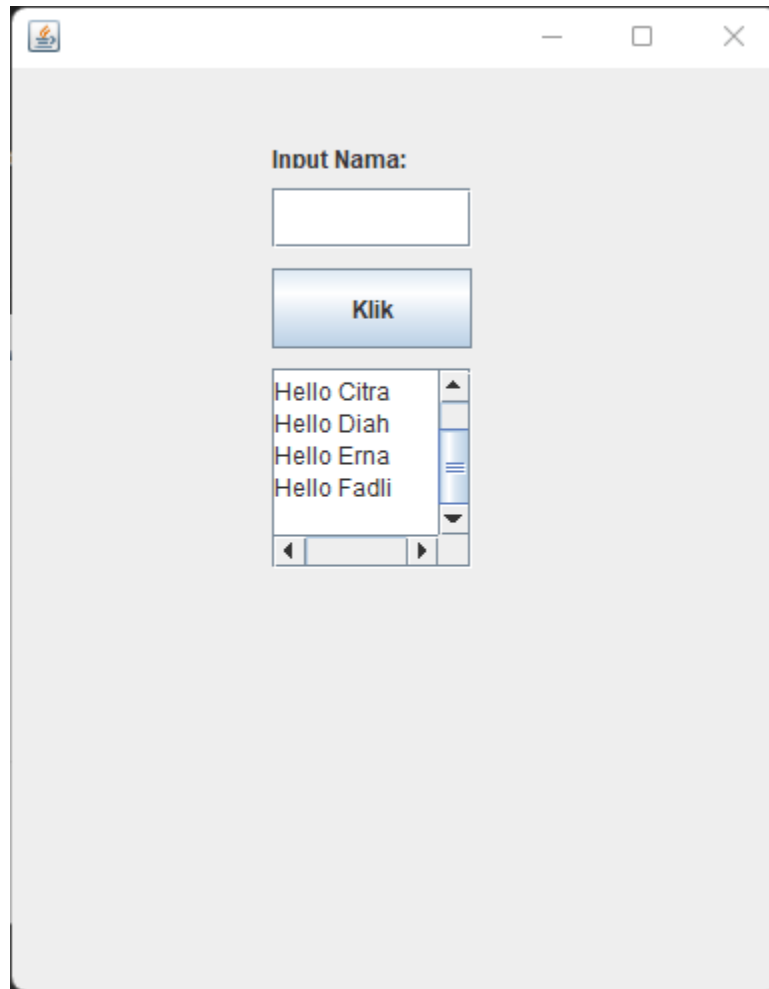
Gambar 13 Kode HelloScrollPane Bagian 2

Kompilasi dan jalankan kelas HelloScrollPane tersebut sehingga menampilkan jendela seperti pada gambar 14.



Gambar 14 Hasil Run Kelas HelloScrollPane

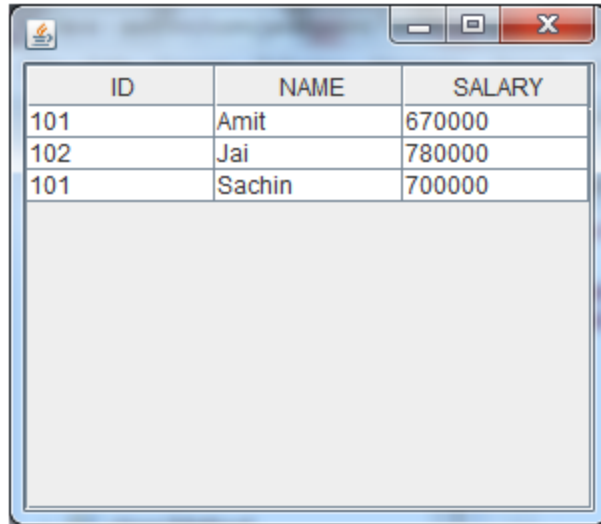
Cobalah inputkan beberapa nama sampai dengan sebuah scroll bar muncul untuk TextArea output seperti pada gambar 15.



Gambar 15 TextArea dengan Scroll Bar

JTable

Kelas JTable digunakan untuk menampilkan data dalam bentuk tabel. Terdiri dari baris dan kolom.



ID	NAME	SALARY
101	Amit	670000
102	Jai	780000
101	Sachin	700000

Gambar 16 Contoh JTable

Latihan 6 – JTable

Buatlah kelas bernama `MyTableModel` dengan kode sebagai berikut.

```

J MyTableModel.java
1  import javax.swing.table.*;
2  import java.util.ArrayList;
3  import java.util.List;
4
5  class MyTableModel extends AbstractTableModel {
6      private String[] columnNames = {"Nama", "Jenis Member"};
7      private ArrayList<ArrayList<String>> data = new ArrayList<ArrayList<String>>();
8
9      public int getColumnCount() {
10         return columnNames.length;
11     }
12
13     public int getRowCount() {
14         return data.size();
15     }
16
17     public String getColumnName(int col) {
18         return columnNames[col];
19     }
20
21     public Object getValueAt(int row, int col) {
22         List<String> rowItem = data.get(row);
23         return rowItem.get(col);
24     }
25
26     public boolean isCellEditable(int row, int col) {
27         return false;
28     }
29
30     public void add(ArrayList<String> value)
31     {
32         data.add(value);
33         fireTableRowsInserted(data.size() - 1, data.size() - 1);
34     }
35
36 }

```

Gambar 17 Kode MyTableModel

Kemudian, buatlah kelas bernama HelloTable dengan kode sebagai berikut.

J HelloTable.java

```
1  import java.awt.event.*;
2  import javax.swing.*;
3  import javax.swing.table.*;
4  import java.util.*;
5
6  public class HelloTable extends JFrame {
7
8      public HelloTable(){
9          this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
10
11          JLabel labelInput = new JLabel("Nama:");
12          labelInput.setBounds(15,40,350,10);
13
14          JTextField textField = new JTextField();
15          textField.setBounds(15,60,350,30);
16
17          JLabel labelRadio = new JLabel("Jenis Member:");
18          labelRadio.setBounds(15,100,350,10);
19
20          JRadioButton radioButton1 = new JRadioButton("Silver", true);
21          radioButton1.setBounds(15,115,350,30);
22
23          JRadioButton radioButton2 = new JRadioButton("Gold");
24          radioButton2.setBounds(15,145,350,30);
25
26          JRadioButton radioButton3 = new JRadioButton("Platinum");
27          radioButton3.setBounds(15,175,350,30);
28
29          ButtonGroup bg = new ButtonGroup();
30          bg.add(radioButton1);
31          bg.add(radioButton2);
32          bg.add(radioButton3);
33
34          JButton button = new JButton("Simpan");
35          button.setBounds(15,205,100,40);
36
```

Gambar 18 Kode Kelas HelloTable Bagian 1

```

J HelloTable.java
36
37     javax.swing.JTable table = new JTable();
38     JScrollPane scrollableTable = new JScrollPane(table);
39     scrollableTable.setBounds(15,250,350,200);
40
41     MyTableModel tableModel = new MyTableModel();
42     table.setModel(tableModel);
43
44     button.addActionListener(new ActionListener(){
45     public void actionPerformed(ActionEvent e){
46         String jenisMember = "";
47         if(radioButton1.isSelected()){
48             jenisMember = radioButton1.getText();
49         }
50         if(radioButton2.isSelected()){
51             jenisMember = radioButton2.getText();
52         }
53         if(radioButton3.isSelected()){
54             jenisMember = radioButton3.getText();
55         }
56
57         String nama = textField.getText();
58         tableModel.add(new ArrayList<>(Arrays.asList(nama, jenisMember)));
59     }
60 });
61
62     this.add(button);
63     this.add(textField);
64     this.add(labelRadio);
65     this.add(radioButton1);
66     this.add(radioButton2);
67     this.add(radioButton3);
68     this.add(labelInput);
69     this.add(scrollableTable);
70

```

Gambar 19 Kode Kelas HelloTable Bagian 2

```
70
71     this.setSize(400,500);
72     this.setLayout(null);
73 }
74
75 public static void main(String[] args) {
76     javax.swing.SwingUtilities.invokeLater(new Runnable() {
77         public void run() {
78             HelloTable h = new HelloTable();
79             h.setVisible(true);
80         }
81     });
82 }
83
84 }
```

Gambar 20 Kode Kelas HelloTable Bagian 3

Kompilasi kelas MyTableModel dan HelloTable, kemudian jalankan kelas HelloTable. Kelas HelloTable akan menampilkan jendela seperti pada gambar berikut.

Nama:

Jenis Member:

☒ Silver

☐ Gold

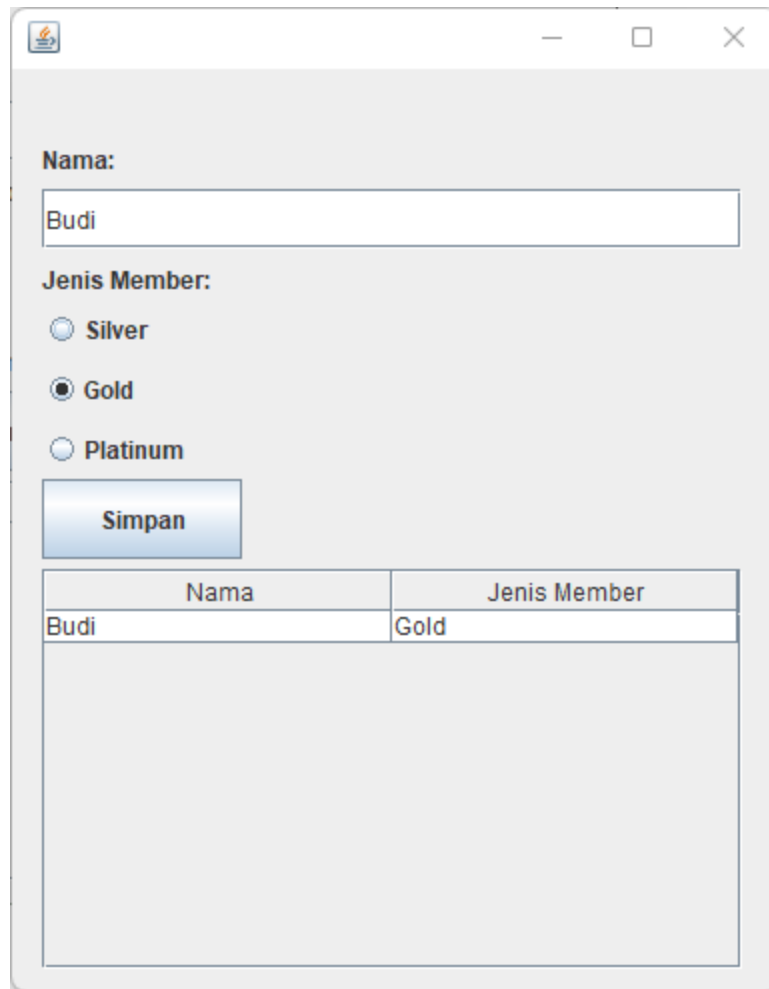
☐ Platinum

Simpan

Nama	Jenis Member
------	--------------

Gambar 21 Hasil Run HelloTable

Cobalah isi nama dan jenis member kemudian klik Simpan. Data dari inputan tersebut akan ditampung ke dalam tabel yang ada di bawah, seperti pada gambar 22 berikut ini.



Nama:

Budi

Jenis Member:

☐ Silver

☒ Gold

☐ Platinum

Simpan

Nama	Jenis Member
Budi	Gold

Gambar 22 Penggunaan Table

Latihan 7

Tambahkan penggunaan JTable untuk aplikasi biodata yang telah dibuat di tugas sebelumnya.