

Laporan Hasil Praktikum

Pemrograman Desktop



Tugas 9

ALYA AIMAN SALSABILA ARIF

1817101379

Tingkat III Rekayasa Perangkat Lunak Kripto

Politeknik Siber dan Sandi Negara

2020/2021

1. Buat basis data “diagramdata”

```
MariaDB [(none)]> create database diagramdata;  
Query OK, 1 row affected (0.003 sec)
```

2. Buat tabel diagrambar

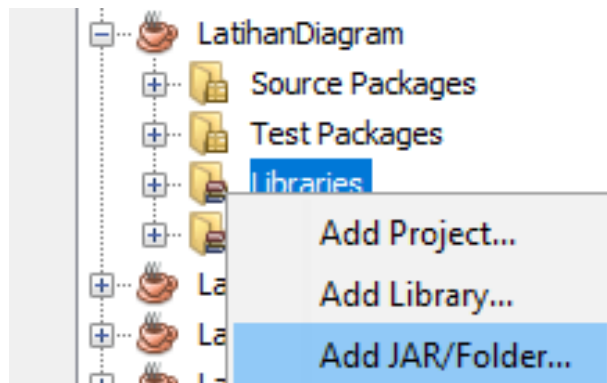
```
MariaDB [(none)]> use diagramdata  
Database changed  
MariaDB [diagramdata]> create table diagrambar (nilai1 int(11) not null,  
-> nilai2 int(11) not null,  
-> nilai3 int(11) not null);  
Query OK, 0 rows affected (0.790 sec)
```

3. Buat tabel diagrapie

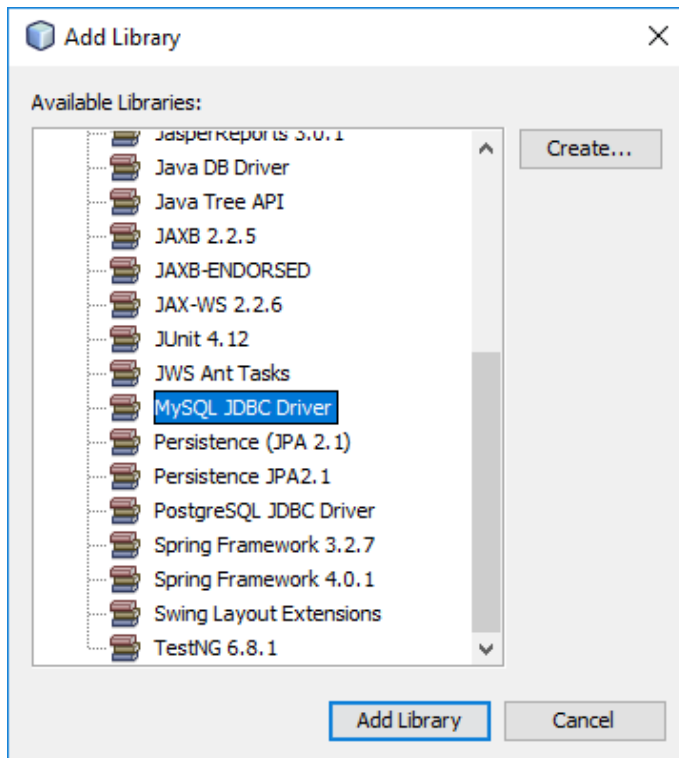
```
MariaDB [diagramdata]> create table diagrapie (nilai1 int(11) not null,  
-> nilai2 int(11) not null,  
-> nilai3 int(11) not null);  
Query OK, 0 rows affected (0.990 sec)
```

4. Buat projek baru dengan nama LatihanDiagram

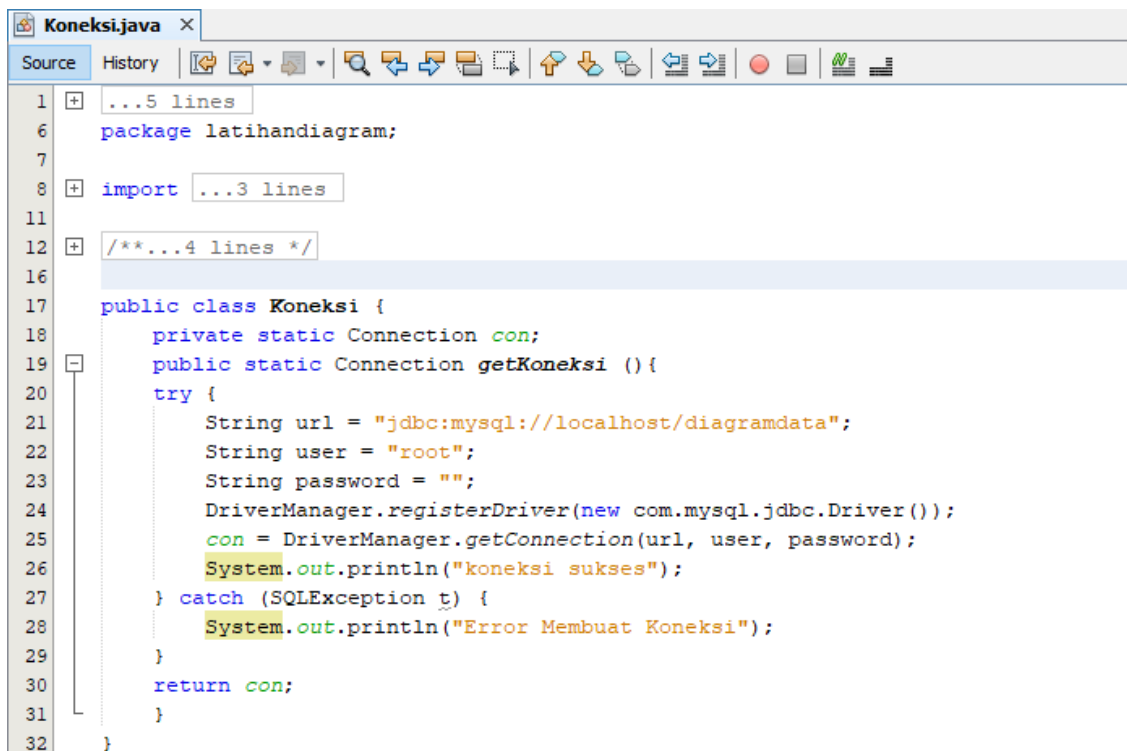
5. Tambahkan *library* dengan cara berikut



6. Tambahkan juga *library* MySQL JDBC Driver

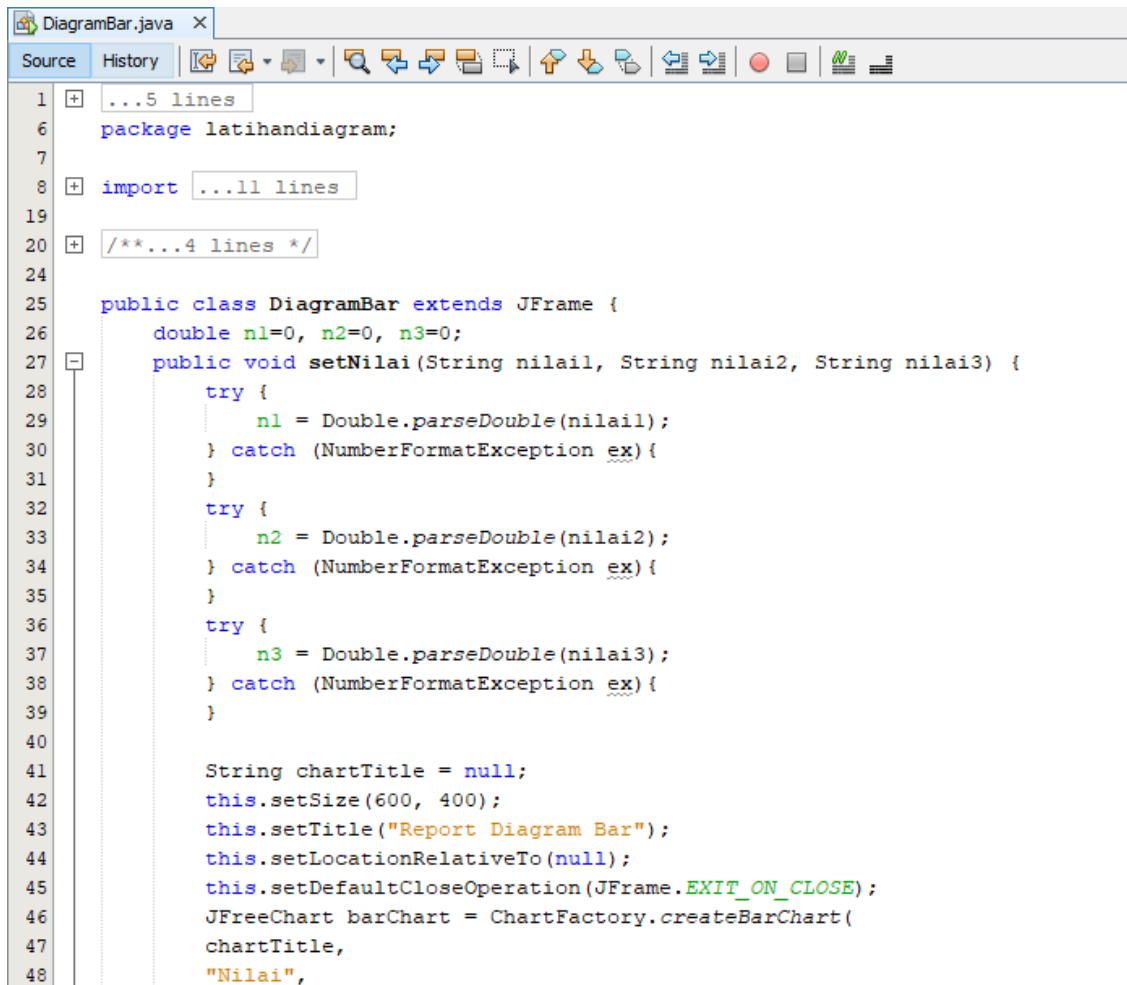


7. Buatlah kelas Koneksi dan masukkan *source code* sebagai berikut

The image shows a code editor window for a file named "Koneksi.java". The editor has a toolbar with various icons for editing and running code. The code is as follows:

```
1  ...5 lines
6  package latihandigram;
7
8  import ...3 lines
11
12  /**...4 lines */
16
17  public class Koneksi {
18      private static Connection con;
19      public static Connection getKoneksi () {
20          try {
21              String url = "jdbc:mysql://localhost/diagramdata";
22              String user = "root";
23              String password = "";
24              DriverManager.registerDriver(new com.mysql.jdbc.Driver());
25              con = DriverManager.getConnection(url, user, password);
26              System.out.println("koneksi sukses");
27          } catch (SQLException t) {
28              System.out.println("Error Membuat Koneksi");
29          }
30          return con;
31      }
32  }
```

8. Buatlah kelas DiagramBar dan masukkan *source code* sebagai berikut



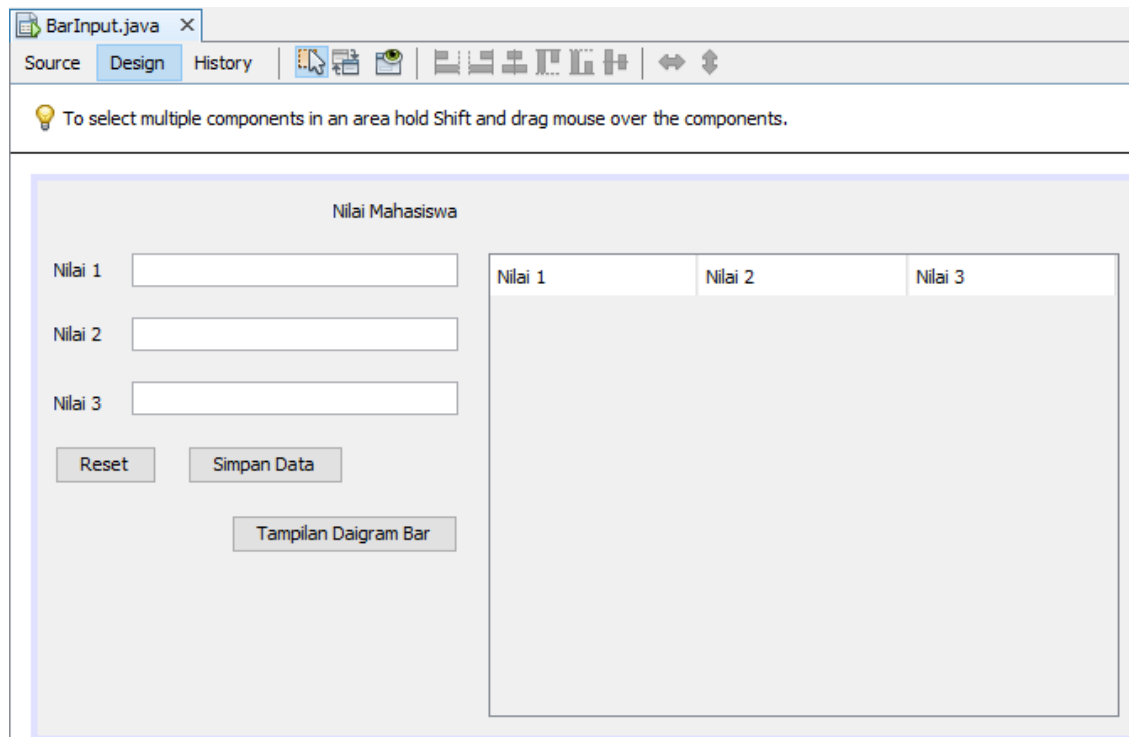
```
DiagramBar.java x
Source History
1  ...5 lines
6  package latihandigram;
7
8  import ...11 lines
19
20  /**...4 lines */
24
25  public class DiagramBar extends JFrame {
26      double n1=0, n2=0, n3=0;
27      public void setNilai(String nilai1, String nilai2, String nilai3) {
28          try {
29              n1 = Double.parseDouble(nilai1);
30          } catch (NumberFormatException ex) {
31          }
32          try {
33              n2 = Double.parseDouble(nilai2);
34          } catch (NumberFormatException ex) {
35          }
36          try {
37              n3 = Double.parseDouble(nilai3);
38          } catch (NumberFormatException ex) {
39          }
40
41          String chartTitle = null;
42          this.setSize(600, 400);
43          this.setTitle("Report Diagram Bar");
44          this.setLocationRelativeTo(null);
45          this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
46          JFreeChart barChart = ChartFactory.createBarChart(
47              chartTitle,
48              "Nilai",
```

```

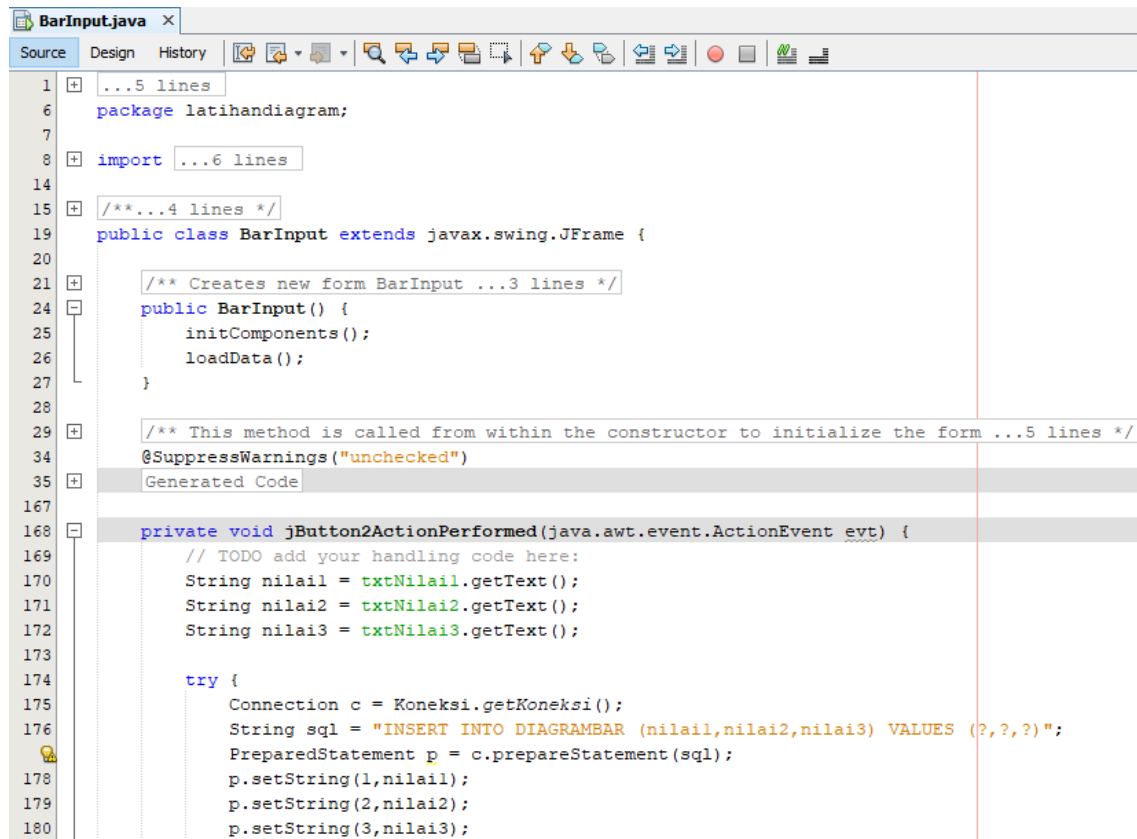
49         "Frekuensi",
50         createDataset(),
51         PlotOrientation.VERTICAL,
52         true, true, true);
53         ChartPanel chartPanel = new ChartPanel ( barChart );
54         chartPanel.setPreferredSize(new java.awt.Dimension( 800 , 370 ) );
55         setContentPane ( chartPanel );
56     }
57
58     private CategoryDataset createDataset() {
59         final DefaultCategoryDataset dataset = new DefaultCategoryDataset();
60         dataset.setValue(n1, "Nilai 1" , "");
61         dataset.setValue(n2, "Nilai 2" , "");
62         dataset.setValue(n3, "Nilai 3" , "");
63         return dataset;
64     }
65
66     public static void main (String args[]){
67         try {
68             UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());
69         } catch (Exception e) {
70         }
71
72         SwingUtilities.invokeLater(new Runnable() {
73             @Override
74             public void run() {
75                 BarInput bi = new BarInput();
76                 bi.setVisible(true);
77             }
78         });
79     }
80 }

```

9. Buatlah kelas BarInput dengan JFrame Form. Buatlah desain kelas BarInput.



10. Masukkan *source code* berikut pada kelas BarInput



```
1  ...5 lines
6  package latihandigram;
7
8  import ...6 lines
14
15  /**...4 lines */
19  public class BarInput extends javax.swing.JFrame {
20
21      /** Creates new form BarInput ...3 lines */
24      public BarInput() {
25          initComponents();
26          loadData();
27      }
28
29      /** This method is called from within the constructor to initialize the form ...5 lines */
34      @SuppressWarnings("unchecked")
35      Generated Code
167
168      private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
169          // TODO add your handling code here:
170          String nilai1 = txtNilai1.getText();
171          String nilai2 = txtNilai2.getText();
172          String nilai3 = txtNilai3.getText();
173
174          try {
175              Connection c = Koneksi.getKoneksi();
176              String sql = "INSERT INTO DIAGRAMBAR (nilai1, nilai2, nilai3) VALUES (?, ?, ?)";
177              PreparedStatement p = c.prepareStatement(sql);
178              p.setString(1, nilai1);
179              p.setString(2, nilai2);
180              p.setString(3, nilai3);
```

```

181         p.executeUpdate();
182         p.close();
183     } catch (SQLException e) {
184         System.out.println("Terjadi Kesalahan");
185     }
186     refreshTabel();
187     loadData();
188 }
189
190 private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
191     // TODO add your handling code here:
192     DiagramBar hasil = new DiagramBar();
193     hasil.setNilai(txtNilai1.getText(), txtNilai2.getText(), txtNilai3.getText());
194     hasil.setVisible(true);
195 }
196
197 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
198     // TODO add your handling code here:
199     refreshTabel();
200     loadData();
201     refreshText();
202 }
203
204 private void loadData(){
205     try{
206         Connection c = Koneksi.getKoneksi();
207         Statement s = c.createStatement();
208
209         String sql = "SELECT * FROM DIAGRAMBAR";
210         ResultSet r = s.executeQuery(sql);
211         while (r.next()) {

```

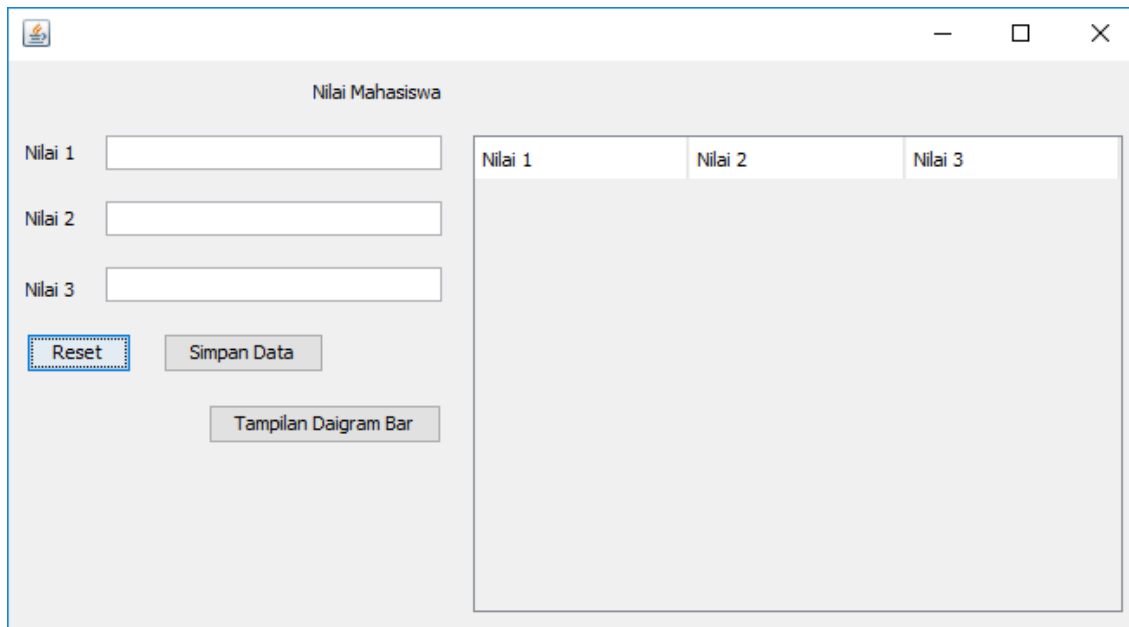


```

212         // lakukan penelusuran baris
213         String nilai1 = r.getString("nilai1");
214         String nilai2 = r.getString("nilai2");
215         String nilai3 = r.getString("nilai3");
216
217         String tbData[] = {nilai1, nilai2, nilai3};
218
219         DefaultTableModel tblModel = (DefaultTableModel)tabelNilai.getModel();
220         tblModel.addRow(tbData);
221     }
222     r.close();
223     s.close();
224     } catch (SQLException e) {
225         System.out.println("Terjadi Kesalahan");
226     }
227 }
228
229 private void refreshTabel() {
230     DefaultTableModel model =(DefaultTableModel)
231     tabelNilai.getModel();
232     while(model.getRowCount()>0) {
233         model.setRowCount(0);
234     }
235 }
236
237 private void refreshText() {
238     txtNilai1.setText("");
239     txtNilai2.setText("");
240     txtNilai3.setText("");
241 }
242
243 /**...3 lines */
244 public static void main(String args[]) {
245     /* Set the Nimbus look and feel */
246     Look and feel setting code (optional)
247
248     /* Create and display the form */
249     java.awt.EventQueue.invokeLater(new Runnable() {
250         public void run() {
251             new BarInput().setVisible(true);
252         }
253     });
254 }
255
256 // Variables declaration - do not modify
257 private javax.swing.JButton jButton1;
258 private javax.swing.JButton jButton2;
259 private javax.swing.JButton jButton3;
260 private javax.swing.JLabel jLabel1;
261 private javax.swing.JLabel jLabel2;
262 private javax.swing.JLabel jLabel3;
263 private javax.swing.JLabel jLabel4;
264 private javax.swing.JScrollPane jScrollPane1;
265 private javax.swing.JTable tabelNilai;
266 private javax.swing.JTextField txtNilai1;
267 private javax.swing.JTextField txtNilai2;
268 private javax.swing.JTextField txtNilai3;
269 // End of variables declaration
270
271 }

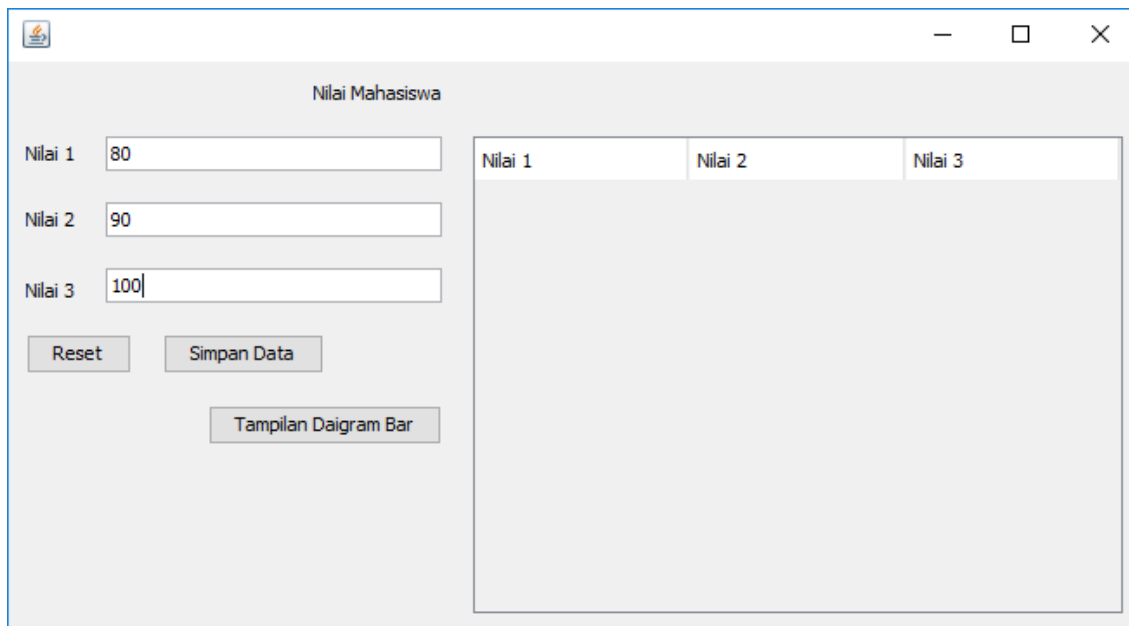
```

11. Jalankan program. Program akan menampilkan sebagai berikut



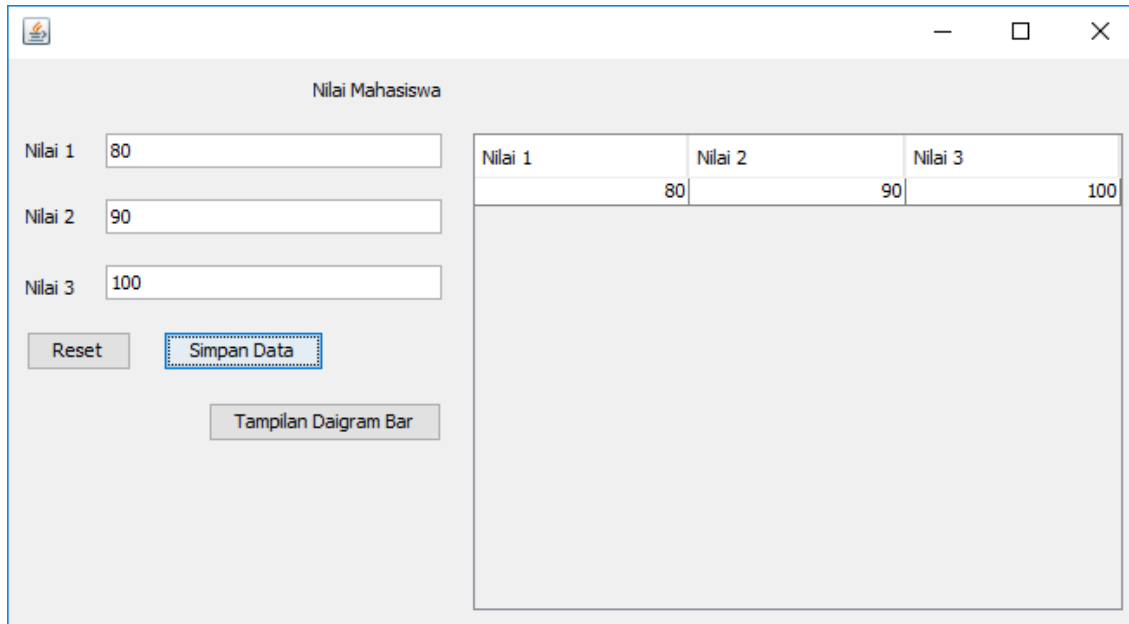
The screenshot shows a Java Swing window titled "Nilai Mahasiswa". On the left side, there are three text input fields labeled "Nilai 1", "Nilai 2", and "Nilai 3", all of which are currently empty. Below these fields are three buttons: "Reset" (highlighted with a blue border), "Simpan Data", and "Tampilan Daigram Bar". On the right side of the window, there is a table with three columns labeled "Nilai 1", "Nilai 2", and "Nilai 3". The table is currently empty, showing only the header row.

Masukkan nilai 1, nilai 2, dan nilai 3. Kemudian tekan *button* Simpan Data untuk menyimpan data



This screenshot shows the same "Nilai Mahasiswa" window after data has been entered. The "Nilai 1" field contains "80", the "Nilai 2" field contains "90", and the "Nilai 3" field contains "100". The "Reset" button is now a standard gray button. The "Simpan Data" and "Tampilan Daigram Bar" buttons remain unchanged. The table on the right is still empty.

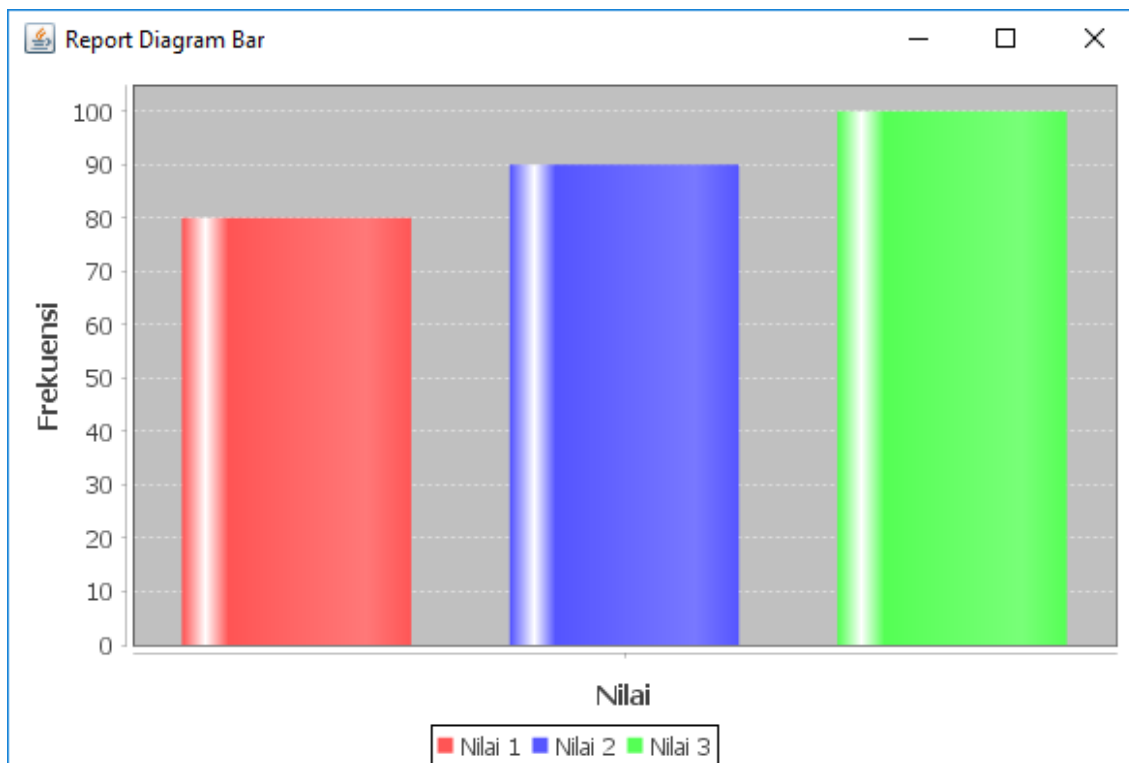
Program akan menampilkan seperti berikut. Tekan tombol *button* Tampilkan Diagram Bar untuk menampilkan diagram bar.



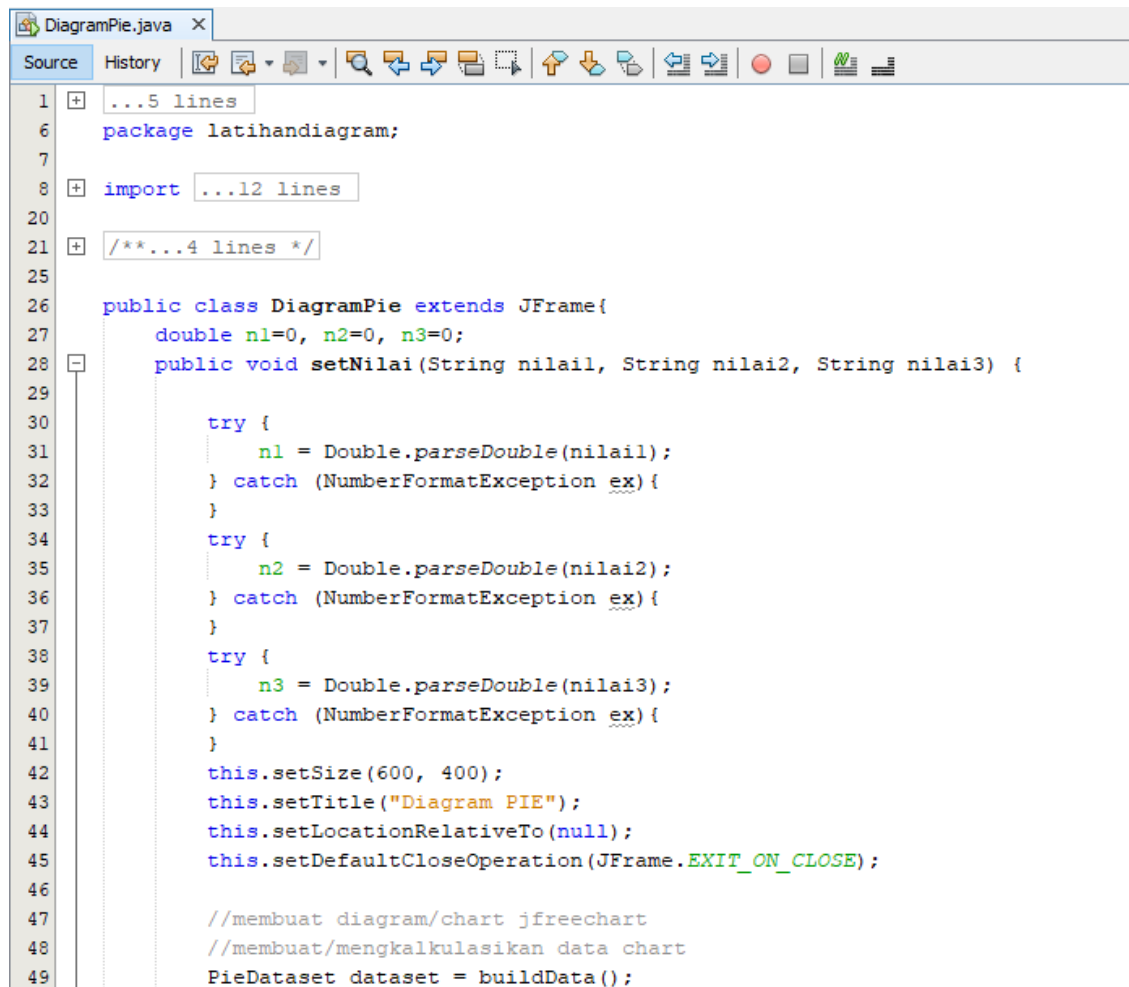
The screenshot shows a Java Swing window titled "Nilai Mahasiswa". On the left, there are three text input fields labeled "Nilai 1", "Nilai 2", and "Nilai 3" containing the values 80, 90, and 100 respectively. Below these fields are three buttons: "Reset", "Simpan Data" (highlighted with a blue dashed border), and "Tampilkan Daigram Bar". On the right, there is a table with three columns: "Nilai 1", "Nilai 2", and "Nilai 3", containing the values 80, 90, and 100. Below the table is a large empty rectangular area for the bar chart.

Nilai 1	Nilai 2	Nilai 3
80	90	100

Program akan menampilkan sebagai berikut



12. Buatlah kelas DiagramPie dan masukkan *source code* sebagai berikut



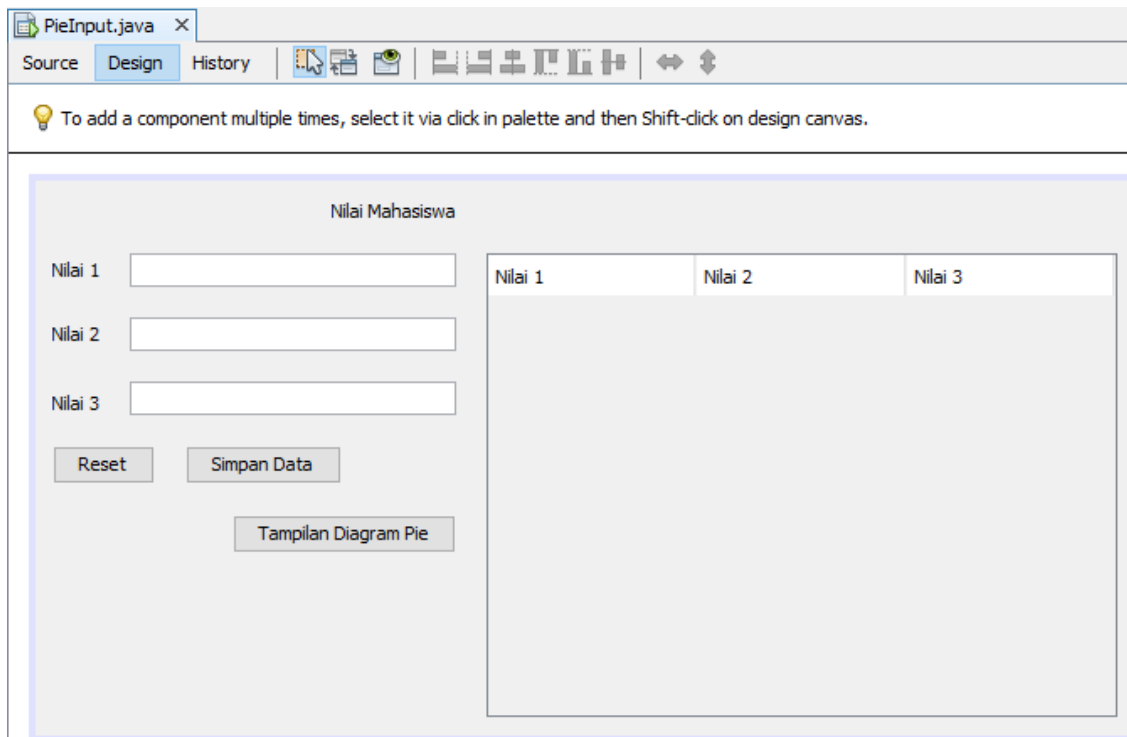
```
1  ...5 lines
6  package latihandigram;
7
8  import ...12 lines
20
21  /**...4 lines */
25
26  public class DiagramPie extends JFrame{
27      double n1=0, n2=0, n3=0;
28      public void setNilai(String nilai1, String nilai2, String nilai3) {
29
30          try {
31              n1 = Double.parseDouble(nilai1);
32          } catch (NumberFormatException ex){
33              }
34          try {
35              n2 = Double.parseDouble(nilai2);
36          } catch (NumberFormatException ex){
37              }
38          try {
39              n3 = Double.parseDouble(nilai3);
40          } catch (NumberFormatException ex){
41              }
42          this.setSize(600, 400);
43          this.setTitle("Diagram PIE");
44          this.setLocationRelativeTo(null);
45          this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
46
47          //membuat diagram/chart jfreechart
48          //membuat/mengkalkulasikan data chart
49          PieDataset dataset = buildData();
```

```

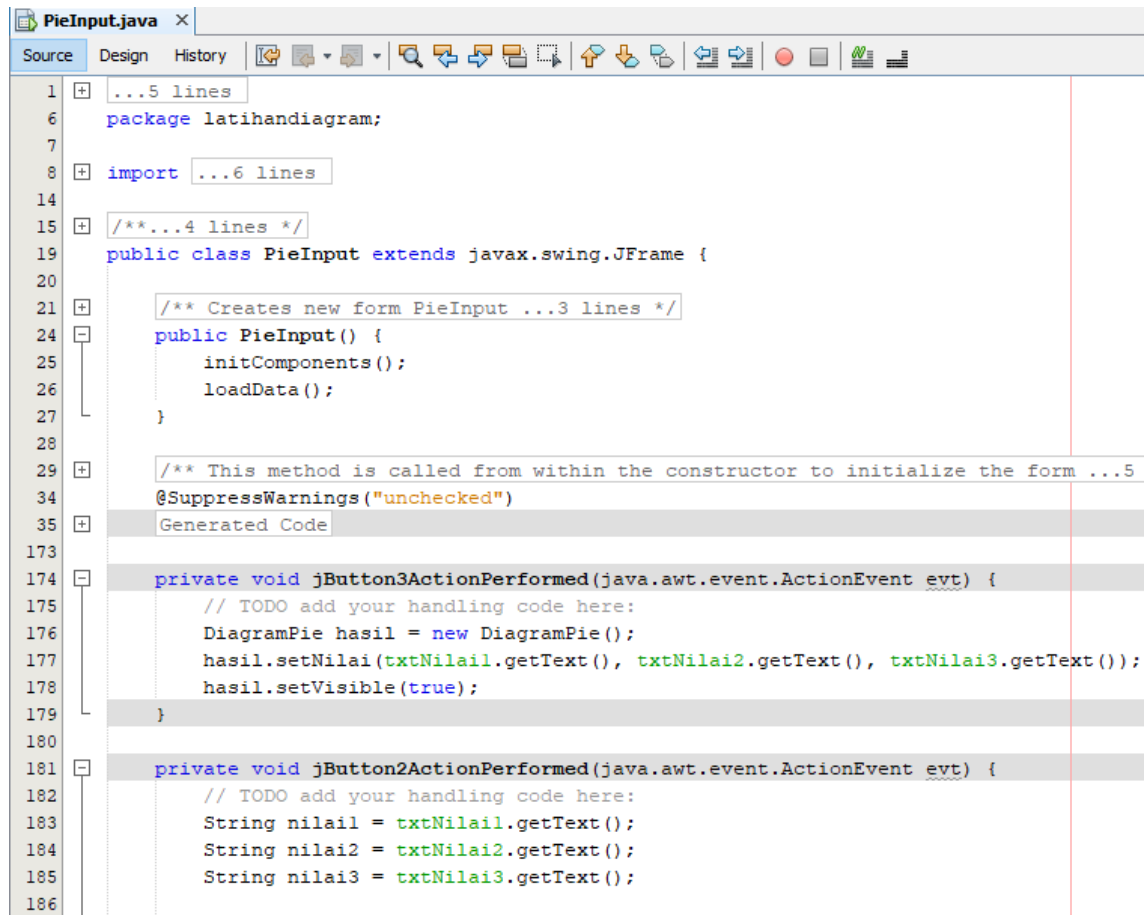
50 //membuat chart berdasarkan data yang ada pada dataset
51 JFreeChart chart = buildChart(dataset, "Data dari Grafik Input");
52 //membuat JPANEL untuk INPUT diagram PIE
53 JPanel pane = new JPanel(new GridLayout(1,1));
54 //menambahkan CHART 1 dan 2 ke PANEL
55 pane.add(new ChartPanel(chart));
56 //memasukkan chart panel ke dalam jpanel
57 this.setContentPane(pane);
58 }
59
60 private JFreeChart buildChart(PieDataset dataset, String judul) {
61     JFreeChart chart = ChartFactory.createPieChart3D(judul, //Judul Chart
62     dataset, //data chart yang akan ditampilkan
63     true, //jika iya maka chart/diagram akan ditampilkan
64     true,
65     false);
66     PiePlot3D plot = (PiePlot3D) chart.getPlot();
67     plot.setStartAngle(290);
68     plot.setDirection(Rotation.CLOCKWISE);
69     plot.setForegroundAlpha(0.5f);
70     return chart;
71 }
72
73 private PieDataset buildData() {
74     DefaultPieDataset dataSet = new DefaultPieDataset();
75     dataSet.setValue("Nilai 1", n1);
76     dataSet.setValue("Nilai 2", n2);
77     dataSet.setValue("Nilai 3", n3);
78
79     return dataSet;
80 }
81
82 public static void main(String[] args) {
83     try {
84         UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());
85     } catch (Exception e) {
86     }
87
88     SwingUtilities.invokeLater(new Runnable() {
89         @Override
90         public void run() {
91             new PieInput().setVisible(true);
92         }
93     });
94 }
95 }

```

13. Buatlah kelas PieInput dengan JFrame Form. Buatlah desain kelas PieInput.



14. Masukkan *source code* berikut pada kelas PieInput



```
1  ...5 lines
6  package latihandigram;
7
8  import ...6 lines
14
15  /**...4 lines */
19  public class PieInput extends javax.swing.JFrame {
20
21      /** Creates new form PieInput ...3 lines */
24      public PieInput() {
25          initComponents();
26          loadData();
27      }
28
29      /** This method is called from within the constructor to initialize the form ...5
34      @SuppressWarnings("unchecked")
35      Generated Code
173
174      private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
175          // TODO add your handling code here:
176          DiagramPie hasil = new DiagramPie();
177          hasil.setNilai(txtNilai1.getText(), txtNilai2.getText(), txtNilai3.getText());
178          hasil.setVisible(true);
179      }
180
181      private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
182          // TODO add your handling code here:
183          String nilai1 = txtNilai1.getText();
184          String nilai2 = txtNilai2.getText();
185          String nilai3 = txtNilai3.getText();
186
```

```

187     try {
188         Connection c = Koneksi.getKoneksi();
189         String sql = "INSERT INTO DIAGRAMPIE (nilai1,nilai2,nilai3) VALUES (?,?,?)";
190         PreparedStatement p = c.prepareStatement(sql);
191         p.setString(1,nilai1);
192         p.setString(2,nilai2);
193         p.setString(3,nilai3);
194         p.executeUpdate();
195         p.close();
196     } catch (SQLException e) {
197         System.out.println("Terjadi Kesalahan");
198     }
199     refreshTabel();
200     loadData();
201 }
202
203 private void txtNilai1ActionPerformed(java.awt.event.ActionEvent evt) {
204     // TODO add your handling code here:
205 }
206
207 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
208     // TODO add your handling code here:
209     refreshTabel();
210     loadData();
211     refreshText();
212 }
213
214 private void loadData() {
215     try{
216         Connection c = Koneksi.getKoneksi();
217         Statement s = c.createStatement();
218
219         String sql = "SELECT * FROM DIAGRAMBAR";
220         ResultSet r = s.executeQuery(sql);
221         while (r.next()) {
222             // lakukan penelusuran baris
223             String nilai1 = r.getString("nilai1");
224             String nilai2 = r.getString("nilai2");
225             String nilai3 = r.getString("nilai3");
226
227             String tbData[] = {nilai1, nilai2, nilai3};
228
229             DefaultTableModel tblModel = (DefaultTableModel)tabelNilai.getModel();
230             tblModel.addRow(tbData);
231         }
232         r.close();
233         s.close();
234     } catch (SQLException e) {
235         System.out.println("Terjadi Kesalahan");
236     }
237 }
238
239 private void refreshTabel() {
240     DefaultTableModel model =(DefaultTableModel)
241     tabelNilai.getModel();
242     while(model.getRowCount()>0){
243         model.setRowCount(0);
244     }
245 }
246
247 private void refreshText() {
248     txtNilai1.setText("");
249     txtNilai2.setText("");

```



```

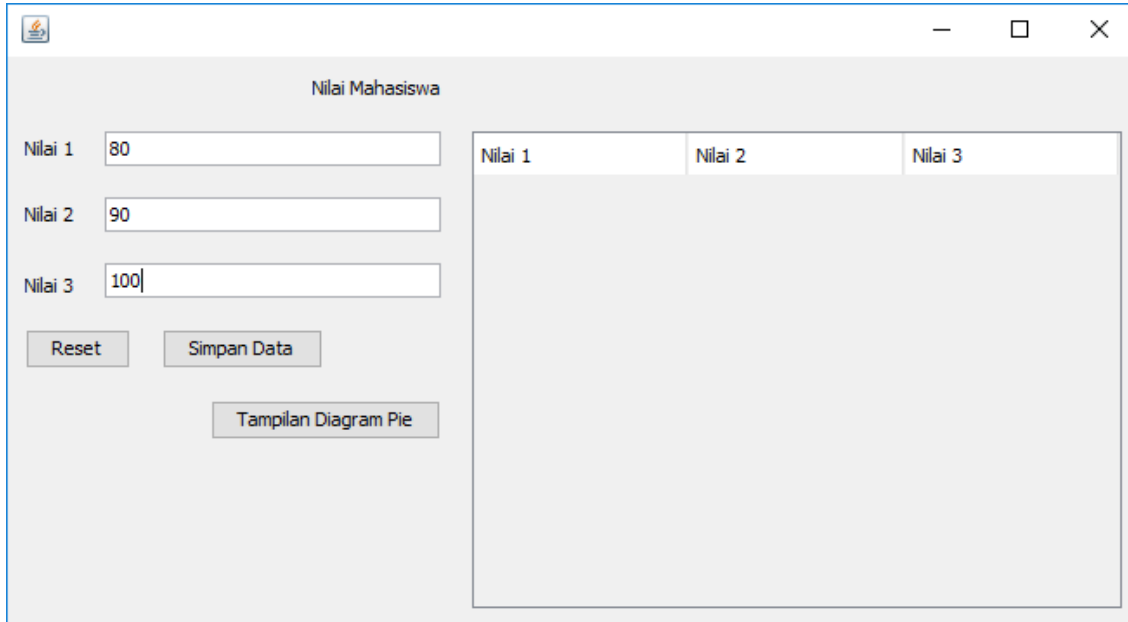
250     txtNilai3.setText("");
251 }
252
253 /**
254  * @param args the command line arguments
255  */
256 public static void main(String args[]) {
257     /* Set the Nimbus look and feel */
258     Look and feel setting code (optional)
259
260     /* Create and display the form */
261     java.awt.EventQueue.invokeLater(new Runnable() {
262         public void run() {
263             new PieInput().setVisible(true);
264         }
265     });
266 }
267
268 // Variables declaration - do not modify
269 private javax.swing.JButton jButton1;
270 private javax.swing.JButton jButton2;
271 private javax.swing.JButton jButton3;
272 private javax.swing.JLabel jLabel1;
273 private javax.swing.JLabel jLabel2;
274 private javax.swing.JLabel jLabel3;
275 private javax.swing.JLabel jLabel4;
276 private javax.swing.JScrollPane jScrollPane1;
277 private javax.swing.JTable tabelNilai;
278 private javax.swing.JTextField txtNilai1;
279 private javax.swing.JTextField txtNilai2;
280 private javax.swing.JTextField txtNilai3;
281 // End of variables declaration
282 }

```

15. Jalankan program. Program akan menampilkan sebagai berikut

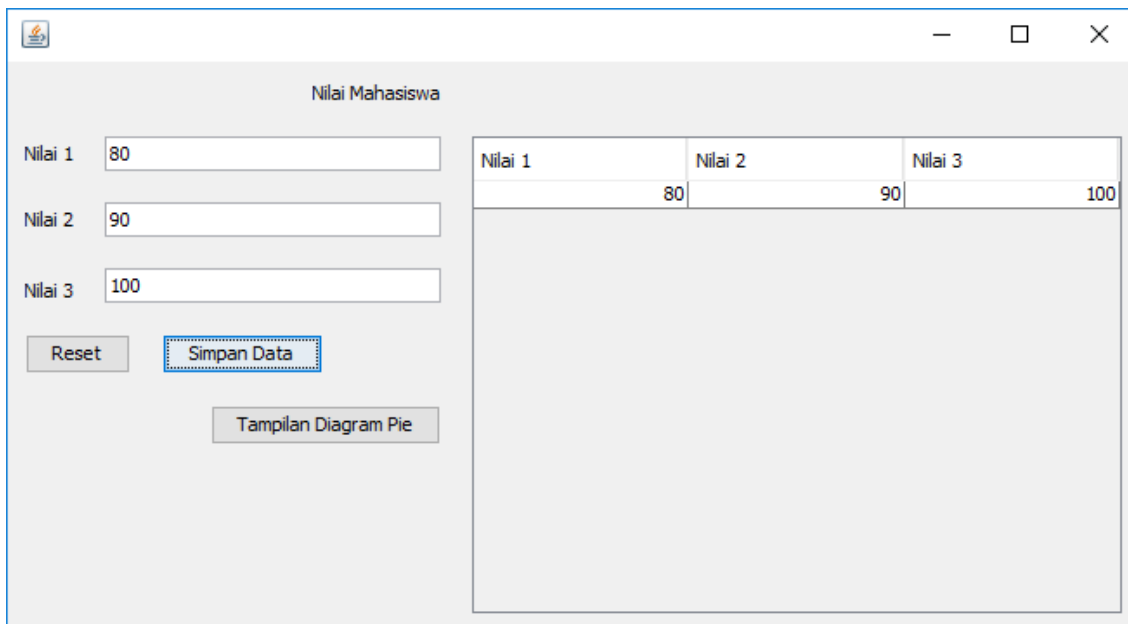
The screenshot shows a Java Swing window titled "Nilai Mahasiswa". On the left side, there are three text input fields labeled "Nilai 1", "Nilai 2", and "Nilai 3". Below these fields are three buttons: "Reset", "Simpan Data", and "Tampilan Diagram Pie". To the right of the input fields, there is a table with three columns labeled "Nilai 1", "Nilai 2", and "Nilai 3". The table is currently empty.

Masukkan nilai 1, nilai 2, dan nilai 3. Kemudian tekan *button* Simpan Data untuk menyimpan data



The screenshot shows a Java Swing window titled "Nilai Mahasiswa". On the left, there are three text input fields labeled "Nilai 1", "Nilai 2", and "Nilai 3" containing the values 80, 90, and 100 respectively. Below these fields are three buttons: "Reset", "Simpan Data", and "Tampilkan Diagram Pie". On the right side of the window is a large empty rectangular area, likely intended for displaying a pie chart.

Program akan menampilkan seperti berikut. Tekan tombol *button* Tampilkan Diagram Pie untuk menampilkan diagram pie.



This screenshot shows the same application window after the "Simpan Data" button has been clicked. The input fields still contain the values 80, 90, and 100. The "Simpan Data" button is now highlighted with a dashed blue border. The large rectangular area on the right now contains a table with the following data:

Nilai 1	Nilai 2	Nilai 3
80	90	100

Program akan menampilkan sebagai berikut

