

LAPORAN AKHIR PRAKTIKUM

Mata Praktikum : Rekayasa Perangkat Lunak 2
Kelas : 4IA26
Praktikum ke- : 3
Tanggal : 08 November 2024
Materi : MVC
NPM : 50421002
Nama : Abdan Syakur
Ketua Asisten :
Nama Asisten :
Paraf Asisten :
Jumlah Lembar : 14



LABORATORIUM TEKNIK INFORMATIKA

UNIVERSITAS GUNADARMA

2023/2024

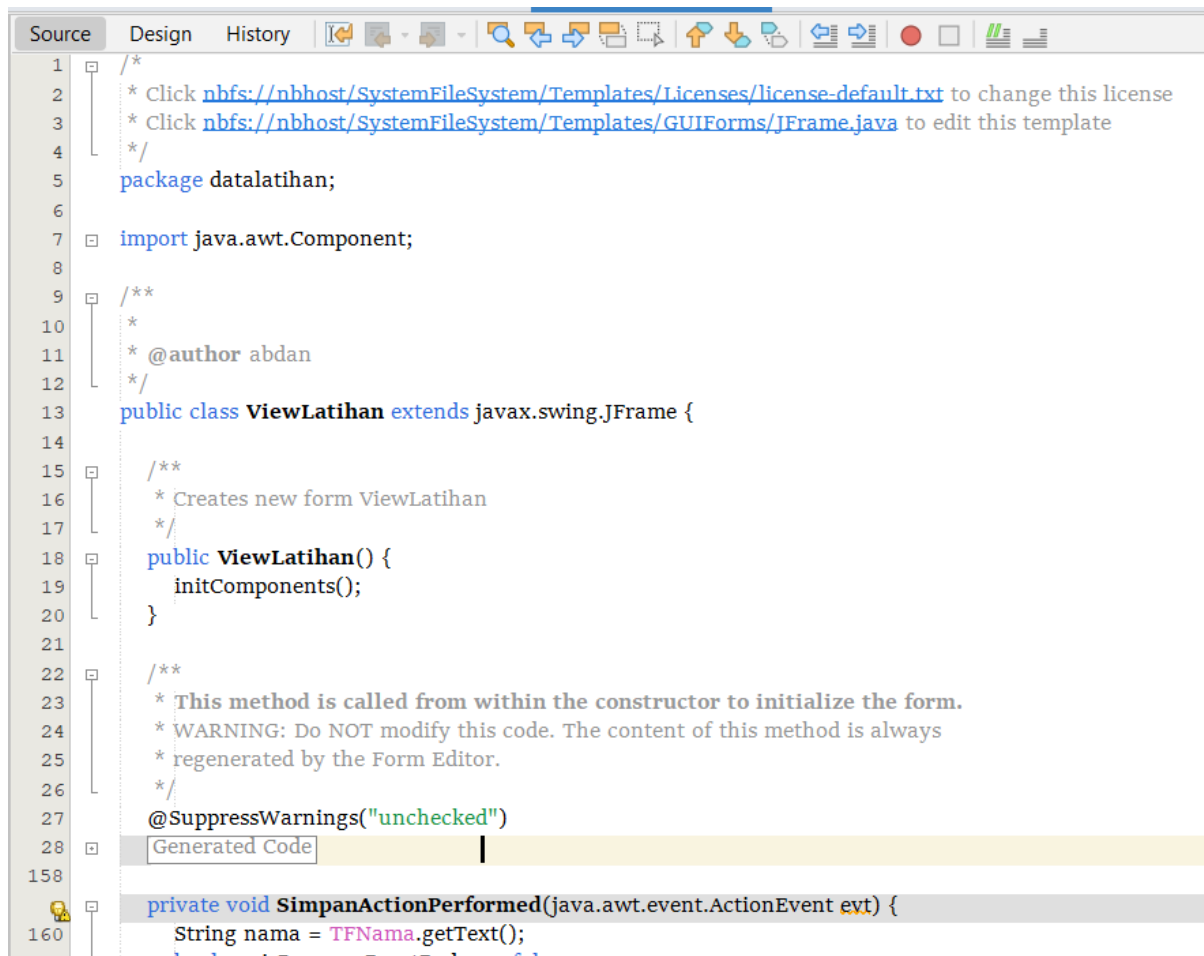
LISTING

```
Source History
1 package datalatihan;
2
3 import java.util.ArrayList;
4 import javax.swing.table.DefaultTableModel;
5
6 public class ControllerLatihan {
7     private ArrayList<ModelLatihan> arrayData;
8     private DefaultTableModel tableList;
9
10    public ControllerLatihan() {
11        arrayData = new ArrayList<ModelLatihan>();
12
13        String[] columnNames = {"Nama", "Jenis Latihan", "Waktu Latihan (menit)"};
14        tableList = new DefaultTableModel(columnNames, 0);
15    }
16
17    public void insertData(String nama, boolean jenisLatihan, int waktuLatihan) {
18        ModelLatihan lth = new ModelLatihan(nama, jenisLatihan, waktuLatihan);
19        arrayData.add(lth);
20
21        String jenis = jenisLatihan ? "Jenis 1" : "Jenis 2";
22        tableList.addRow(new Object[]{nama, jenis, waktuLatihan});
23    }
24
25    public DefaultTableModel getTableList() {
26        return tableList;
27    }
28
29    public ArrayList<ModelLatihan> getData() {
30        return arrayData;
31    }
32
33    // Kelas ModelLatihan
34    private static class ModelLatihan {
35        private String Nama;
36        private boolean JenisLatihan;
37        private int WaktuLatihan;
38
39        public ModelLatihan(String nama, boolean jenisLatihan, int waktuLatihan) {
40            this.Nama = nama;
41            this.JenisLatihan = jenisLatihan;
42            this.WaktuLatihan = waktuLatihan;
43        }
44
45        public String getNama() {
46            return Nama;
47        }
48
49        public boolean isJenisLatihan() {
50            return JenisLatihan;
51        }
52
53        public int getWaktuLatihan() {
54            return WaktuLatihan;
55        }
56    }
57
58 }
```

```

1  /*
2   * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3   * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this template
4   */
5  package datalatihan;
6
7  public class DataLatihan {
8      private String Nama;
9      private boolean JenisLatihan;
10     private int WaktuLatihan;
11
12     public DataLatihan(String Nama, boolean JenisLatihan, int WaktuLatihan){
13         this.Nama = Nama;
14         this.JenisLatihan = JenisLatihan;
15         this.WaktuLatihan = WaktuLatihan;
16     }
17
18     public String getNama(){
19         return Nama;
20     }
21
22     public void setNama(String Nama){
23         this.Nama = Nama;
24     }
25
26     public boolean getJenisLatihan(){
27         return JenisLatihan;
28     }
29
30     public void setJenisLatihan(boolean JenisLatihan){
31         this.JenisLatihan = JenisLatihan;
32     }
33
34     public int getWaktuLatihan(){
35         return WaktuLatihan;
36     }
37
38     public void setWaktuLatihan(int WaktuLatihan){
39         this.WaktuLatihan = WaktuLatihan;
40     }
41
42 }
43

```



```
1  /*
2  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3  * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template
4  */
5  package datalatihan;
6
7  import java.awt.Component;
8
9  /**
10   *
11   * @author abdan
12   */
13  public class ViewLatihan extends javax.swing.JFrame {
14
15      /**
16       * Creates new form ViewLatihan
17       */
18      public ViewLatihan() {
19          initComponents();
20      }
21
22      /**
23       * This method is called from within the constructor to initialize the form.
24       * WARNING: Do NOT modify this code. The content of this method is always
25       * regenerated by the Form Editor.
26       */
27      @SuppressWarnings("unchecked")
28      Generated Code
29
30      private void SimpanActionPerformed(java.awt.event.ActionEvent evt) {
31          String nama = TFNama.getText();
32          // TODO add your handling code here:
33      }
```

```

159
160 private void SimpanActionPerformed(java.awt.event.ActionEvent evt) {
161     String nama = TFNama.getText();
162     boolean isPenurunBeratBadan = false;
163     if (JenisLatihan.getSelectedItem().toString().equals("Penurun Berat Badan")) {
164         isPenurunBeratBadan = true;
165     }
166     int waktuLatihan = Integer.parseInt(TFWaktuLatihan.getText());
167     lth.InsertData(nama, isPenurunBeratBadan, waktuLatihan);
168     tablelist.setViewPortView(lth.showData());
169 }
170
171 private void TFWaktuLatihanActionPerformed(java.awt.event.ActionEvent evt) {
172     // TODO add your handling code here:
173 }
174
175 private void JenisLatihanActionPerformed(java.awt.event.ActionEvent evt) {
176     // TODO add your handling code here:
177 }
178
179 private void TFNamaActionPerformed(java.awt.event.ActionEvent evt) {
180     // TODO add your handling code here:
181 }
182
183 /**
184  * @param args the command line arguments
185  */
186 public static void main(String args[]) {
187     /* Set the Nimbus look and feel */
188     //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
189     /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

```

Source Design History

```

190     * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
191     */
192     try {
193         for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
194             if ("Nimbus".equals(info.getName())) {
195                 javax.swing.UIManager.setLookAndFeel(info.getClassName());
196                 break;
197             }
198         }
199     } catch (ClassNotFoundException ex) {
200         java.util.logging.Logger.getLogger(ViewLatihan.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
201     } catch (InstantiationException ex) {
202         java.util.logging.Logger.getLogger(ViewLatihan.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
203     } catch (IllegalAccessException ex) {
204         java.util.logging.Logger.getLogger(ViewLatihan.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
205     } catch (javax.swing.UnsupportedLookAndFeelException ex) {
206         java.util.logging.Logger.getLogger(ViewLatihan.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
207     }
208     //</editor-fold>
209
210     /* Create and display the form */
211     java.awt.EventQueue.invokeLater(new Runnable() {
212         public void run() {
213             new ViewLatihan().setVisible(true);
214         }
215     });
216 }
217
218 // Variables declaration - do not modify
219 private javax.swing.JComboBox<String> JenisLatihan;
220 private javax.swing.JButton Simpan;

```

```
Source Design History
218 // Variables declaration - do not modify
219 private javax.swing.JComboBox<String> JenisLatihan;
220 private javax.swing.JButton Simpan;
221 private javax.swing.JTextField TFNama;
222 private javax.swing.JTextField TFWaktuLatihan;
223 private javax.swing.JLabel jLabel1;
224 private javax.swing.JLabel jLabel2;
225 private javax.swing.JLabel jLabel3;
226 private javax.swing.JLabel jLabel4;
227 private javax.swing.JTable jTable1;
228 private javax.swing.JScrollPane tablelist;
229 // End of variables declaration
230
231 private static class lth {
232
233     private static Component showData() {
234         throw new UnsupportedOperationException("Not supported yet."); // Generated from nbfs://nbh
235     }
236
237     private static void InsertData(String nama, boolean penurunBeratBadan, int waktuLatihan) {
238         throw new UnsupportedOperationException("Not supported yet."); // Generated from nbfs://nbh
239     }
240
241     public lth() {
242     }
243 }
244
245 }
```

Source Design History

Data Latihan

Nama

Jenis Latihan

Waktu Latihan

Nama	Jenis Latihan	Waktu Latihan
------	---------------	---------------

LOGIKA

```
Source History
1 package dataLatihan;
2
3 import java.util.ArrayList;
4 import javax.swing.table.DefaultTableModel;
5
6 public class ControllerLatihan {
7     private ArrayList<ModelLatihan> arrayData;
8     private DefaultTableModel tableList;
9
10    public ControllerLatihan() {
11        arrayData = new ArrayList<ModelLatihan>();
12
13        String[] columnNames = {"Nama", "Jenis Latihan", "Waktu Latihan (menit)"};
14        tableList = new DefaultTableModel(columnNames, 0);
15    }
16
17    public void insertData(String nama, boolean jenisLatihan, int waktuLatihan) {
18        ModelLatihan lth = new ModelLatihan(nama, jenisLatihan, waktuLatihan);
19        arrayData.add(lth);
```

Kode di atas adalah bagian dari program Java untuk mengelola data latihan menggunakan ArrayList dan DefaultTableModel. Kelas ControllerLatihan memiliki atribut arrayData, yaitu daftar objek ModelLatihan, serta tableList, tabel dengan kolom "Nama", "Jenis Latihan", dan "Waktu Latihan (menit)". Konstruktor ControllerLatihan menginisialisasi arrayData dan tableList. Metode insertData digunakan untuk menambahkan objek ModelLatihan baru ke dalam arrayData, dengan parameter nama, jenisLatihan, dan waktuLatihan.

```
21        String jenis = jenisLatihan ? "Jenis 1" : "Jenis 2";
22        tableList.addRow(new Object[]{nama, jenis, waktuLatihan});
23    }
24
25    public DefaultTableModel getTableList() {
26        return tableList;
27    }
28
29    public ArrayList<ModelLatihan> getData() {
30        return arrayData;
31    }
32
33    // Kelas ModelLatihan
34    private static class ModelLatihan {
35        private String Nama;
36        private boolean jenisLatihan;
37        private int WaktuLatihan;
38    }
```

Kode ini adalah lanjutan dari kelas ControllerLatihan, yang menambahkan data ke tableList dan menyediakan metode akses data. Pada metode insertData, variabel jenis ditentukan berdasarkan nilai boolean jenisLatihan, yang menghasilkan "Jenis 1" jika jenisLatihan bernilai true, dan "Jenis 2" jika false. Kemudian, baris baru ditambahkan ke tableList dengan nama,

jenis, dan waktu latihan. Metode `getTableList` mengembalikan objek `DefaultTableModel` untuk menampilkan data di tabel, sementara `getData` mengembalikan `arrayData`, yaitu daftar objek `ModelLatihan`. Kelas `ModelLatihan` (kelas bersarang statis) mendefinisikan atribut untuk menyimpan informasi tentang nama, jenis latihan, dan waktu latihan.

```
40 | □ | public ModelLatihan(String nama, boolean jenisLatihan, int waktuLatihan) {  
41 |   |     this>Nama = nama;  
42 |   |     this.JenisLatihan = jenisLatihan;  
43 |   |     this.WaktuLatihan = waktuLatihan;  
44 |   | }  
45 |  
46 | □ | public String getNama() {  
47 |   |     return>Nama;  
48 |   | }  
49 |  
50 | □ | public boolean isJenisLatihan() {  
51 |   |     return>JenisLatihan;  
52 |   | }  
53 |  
54 | □ | public int getWaktuLatihan() {  
55 |   |     return>WaktuLatihan;  
56 |   | }  
57 | }  
58 | }
```

Kelas `ModelLatihan` adalah kelas bersarang dalam `ControllerLatihan` yang merepresentasikan data latihan individual. Konstruktor `ModelLatihan` menerima tiga parameter (`nama`, `jenisLatihan`, dan `waktuLatihan`) untuk menginisialisasi atribut `Nama`, `JenisLatihan`, dan `WaktuLatihan`. Kelas ini juga menyediakan metode akses (getter) `getNama`, `isJenisLatihan`, dan `getWaktuLatihan` untuk mengambil nilai masing-masing atribut. Dengan demikian, `ModelLatihan` berfungsi sebagai model data yang digunakan oleh `ControllerLatihan` untuk menyimpan dan mengelola informasi latihan dalam daftar data dan tabel.


```

1  /*
2   * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3   * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this template
4   */
5  package datalatihan;
6
7  public class DataLatihan {
8      private String Nama;
9      private boolean JenisLatihan;
10     private int WaktuLatihan;
11
12     public DataLatihan(String Nama, boolean JenisLatihan, int WaktuLatihan){
13         this.Nama = Nama;
14         this.JenisLatihan = JenisLatihan;
15         this.WaktuLatihan = WaktuLatihan;
16     }
17
18     public String getNama(){
19         return Nama;
20     }
21
22     public void setNama(String Nama){
23         this.Nama = Nama;
24     }
25
26     public boolean getJenisLatihan(){
27         return JenisLatihan;
28     }
29
30     public void setJenisLatihan(boolean JenisLatihan){
31         this.JenisLatihan = JenisLatihan;
32     }

```

Kelas DataLatihan di atas adalah model data yang mewakili informasi tentang latihan. Kelas ini memiliki tiga atribut: Nama (tipe String), JenisLatihan (tipe boolean), dan WaktuLatihan (tipe int). Konstruktor DataLatihan menerima tiga parameter untuk menginisialisasi atribut tersebut. Kelas ini juga memiliki metode akses (getter) dan pengubah (setter) untuk setiap atribut, yaitu getNama dan setNama untuk Nama, serta getJenisLatihan dan setJenisLatihan untuk JenisLatihan. Kelas ini digunakan untuk menyimpan dan memodifikasi data latihan.

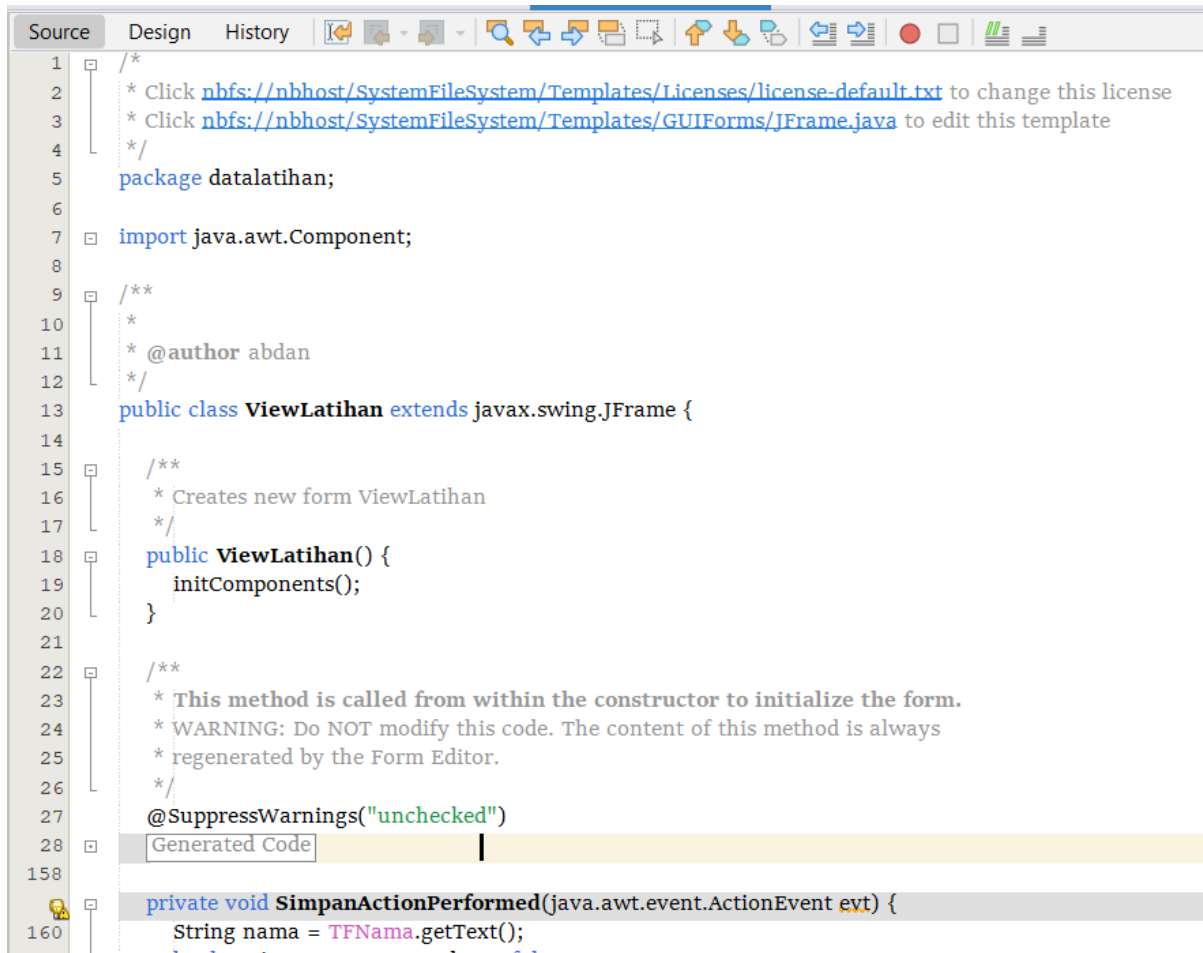
```

33
34     public int getWaktuLatihan(){
35         return WaktuLatihan;
36     }
37
38     public void setWaktuLatihan(int WaktuLatihan){
39         this.WaktuLatihan = WaktuLatihan;
40     }
41
42     }
43

```

Lanjutan kelas DataLatihan ini menambahkan metode getWaktuLatihan dan setWaktuLatihan untuk atribut WaktuLatihan. Metode getWaktuLatihan mengembalikan nilai WaktuLatihan (dalam satuan menit), sementara setWaktuLatihan memungkinkan pengaturan atau pembaruan

nilai WaktuLatihan. Dengan metode ini, kelas DataLatihan lengkap sebagai model data dengan kemampuan untuk menyimpan dan memodifikasi informasi terkait nama, jenis latihan, dan durasi waktu latihan.



```
1  /*
2  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3  * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template
4  */
5  package datalatihan;
6
7  import java.awt.Component;
8
9  /**
10   *
11   * @author abdan
12   */
13  public class ViewLatihan extends javax.swing.JFrame {
14
15      /**
16       * Creates new form ViewLatihan
17       */
18      public ViewLatihan() {
19          initComponents();
20      }
21
22      /**
23       * This method is called from within the constructor to initialize the form.
24       * WARNING: Do NOT modify this code. The content of this method is always
25       * regenerated by the Form Editor.
26       */
27      @SuppressWarnings("unchecked")
28      // Generated Code
29
30      private void SimpanActionPerformed(java.awt.event.ActionEvent evt) {
31          String nama = TFNama.getText();
32          JenisLatihan jLatihan = JLatihan.getSelectedItem();
33          WaktuLatihan wLatihan = TWaktuLatihan.getText();
34          DataLatihan dataLatihan = new DataLatihan(nama, jLatihan, wLatihan);
35          jTable1.add(dataLatihan);
36      }
37  }
```

Kelas ViewLatihan merupakan GUI berbasis JFrame untuk aplikasi latihan menggunakan komponen Swing di Java, yang menyediakan form input untuk memasukkan data latihan seperti nama, jenis latihan (dropdown dengan dua pilihan), dan waktu latihan dalam menit. Saat tombol "Simpan" ditekan, data dari field teks TFNama, combobox JenisLatihan, dan field TFWaktuLatihan diambil dan ditampilkan dalam tabel jTable1, yang berada di dalam komponen tablelist. Komponen utama dalam form ini terdiri dari input teks untuk nama, dropdown untuk jenis latihan, serta input teks untuk waktu latihan, dan tombol "Simpan" untuk menambahkan data ke tabel yang mencatat semua data latihan yang disimpan.

```

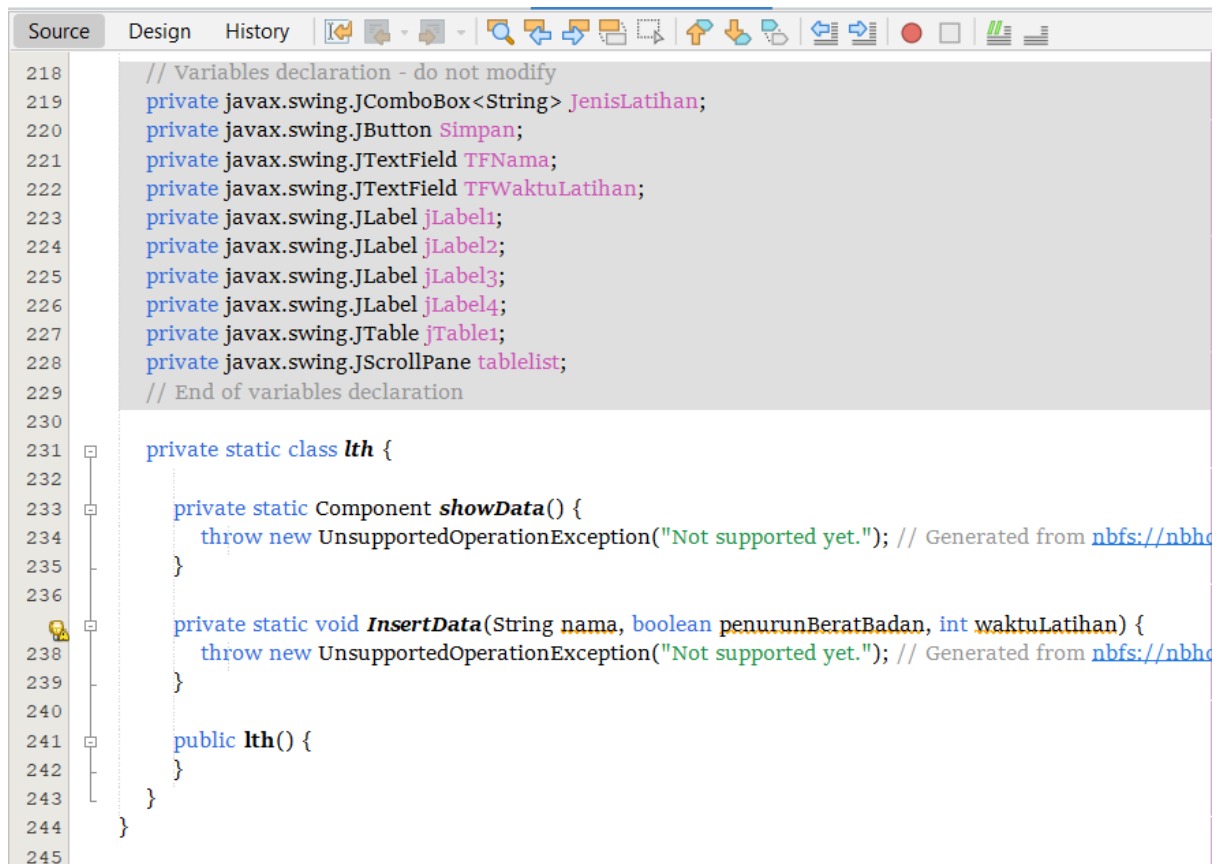
159 private void SimpanActionPerformed(java.awt.event.ActionEvent evt) {
160     String nama = TFNama.getText();
161     boolean isPenurunBeratBadan = false;
162     if (JenisLatihan.getSelectedItems().toString().equals("Penurun Berat Badan")) {
163         isPenurunBeratBadan = true;
164     }
165     int waktuLatihan = Integer.parseInt(TFWaktuLatihan.getText());
166
167     lth.InsertData(nama, isPenurunBeratBadan, waktuLatihan);
168     tablelist.setViewPortView(lth.showData());
169 }
170
171 private void TFWaktuLatihanActionPerformed(java.awt.event.ActionEvent evt) {
172     // TODO add your handling code here:
173 }
174
175 private void JenisLatihanActionPerformed(java.awt.event.ActionEvent evt) {
176     // TODO add your handling code here:
177 }
178
179 private void TFNamaActionPerformed(java.awt.event.ActionEvent evt) {
180     // TODO add your handling code here:
181 }
182
183 /**
184  * @param args the command line arguments
185  */
186 public static void main(String args[]) {
187     /* Set the Nimbus look and feel */
188     //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
189     /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

```

Method `SimpanActionPerformed` pada kelas `ViewLatihan` menangani event ketika tombol "Simpan" ditekan, mengambil nilai dari `TFNama` untuk nama, mengecek jenis latihan dari dropdown `JenisLatihan` (jika terpilih "Penurun Berat Badan", variabel `isPenurunBeratBadan` diset ke `true`), dan mengonversi input dari `TFWaktuLatihan` ke integer untuk menyimpan waktu latihan. Kemudian, data dimasukkan ke dalam model melalui `lth.InsertData()`, dan tabel diperbarui dengan `tablelist.setViewPortView(lth.showData())`. Metode-metode `TFWaktuLatihanActionPerformed`, `JenisLatihanActionPerformed`, dan `TFNamaActionPerformed` saat ini belum memiliki implementasi tambahan.

```
Source Design History
190  * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
191  */
192  try {
193      for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
194          if ("Nimbus".equals(info.getName())) {
195              javax.swing.UIManager.setLookAndFeel(info.getClassName());
196              break;
197          }
198      }
199  } catch (ClassNotFoundException ex) {
200      java.util.logging.Logger.getLogger(ViewLatihan.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
201  } catch (InstantiationException ex) {
202      java.util.logging.Logger.getLogger(ViewLatihan.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
203  } catch (IllegalAccessException ex) {
204      java.util.logging.Logger.getLogger(ViewLatihan.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
205  } catch (javax.swing.UnsupportedLookAndFeelException ex) {
206      java.util.logging.Logger.getLogger(ViewLatihan.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
207  }
208  //</editor-fold>
209
210  /* Create and display the form */
211  java.awt.EventQueue.invokeLater(new Runnable() {
212      public void run() {
213          new ViewLatihan().setVisible(true);
214      }
215  });
216  }
217
218  // Variables declaration - do not modify
219  private javax.swing.JComboBox<String> JenisLatihan;
220  private javax.swing.JButton Simpan;
```

Kode ini berfungsi untuk mengatur tampilan antarmuka grafis menggunakan tema "Nimbus" pada aplikasi Java Swing. Di dalam try block, kode ini mencari tema "Nimbus" di antara daftar tampilan yang terpasang dengan melakukan iterasi pada objek LookAndFeelInfo, dan jika ditemukan, tema tersebut diterapkan menggunakan javax.swing.UIManager.setLookAndFeel(). Jika terjadi kesalahan saat pengaturan tema, seperti ClassNotFoundException, InstantiationException, atau IllegalAccessException, maka akan dicatat menggunakan logger. Setelah itu, bagian java.awt.EventQueue.invokeLater() memastikan bahwa tampilan antarmuka grafis (ViewLatihan) dijalankan di thread yang sesuai sehingga aplikasi dapat ditampilkan kepada pengguna.



```
218 // Variables declaration - do not modify
219 private javax.swing.JComboBox<String> JenisLatihan;
220 private javax.swing.JButton Simpan;
221 private javax.swing.JTextField TFNama;
222 private javax.swing.JTextField TFWaktuLatihan;
223 private javax.swing.JLabel jLabel1;
224 private javax.swing.JLabel jLabel2;
225 private javax.swing.JLabel jLabel3;
226 private javax.swing.JLabel jLabel4;
227 private javax.swing.JTable jTable1;
228 private javax.swing.JScrollPane tablelist;
229 // End of variables declaration
230
231 private static class lth {
232
233     private static Component showData() {
234         throw new UnsupportedOperationException("Not supported yet."); // Generated from nbfs://nbh
235     }
236
237     private static void InsertData(String nama, boolean penurunBeratBadan, int waktuLatihan) {
238         throw new UnsupportedOperationException("Not supported yet."); // Generated from nbfs://nbh
239     }
240
241     public lth() {
242     }
243 }
244
245 }
```

Kode ini mendefinisikan serangkaian komponen GUI dalam aplikasi Java Swing yang mencakup JComboBox untuk memilih jenis latihan (JenisLatihan), JTextField untuk memasukkan nama (TFNama) dan waktu latihan (TFWaktuLatihan), serta JButton (Simpan) untuk menyimpan data. Terdapat juga JTable (jTable1) dan JScrollPane (tablelist) untuk menampilkan daftar data. Selain itu, ada kelas statis lth yang berisi dua metode: InsertData, yang seharusnya menangani penyimpanan data (meskipun saat ini melempar UnsupportedOperationException), dan showData, yang seharusnya menampilkan data tetapi juga belum diimplementasikan. Metode-metode ini kemungkinan besar adalah placeholder yang akan diimplementasikan di masa depan untuk fungsionalitas seperti menyimpan dan menampilkan data latihan di GUI.

OUTPUT

— □ ×

Data Latihan

Nama

Abdan

Jenis Latihan

Penurun Berat Badan ▾

Waktu Latihan

3

Simpan

Nama	Jenis Latihan	Waktu Latihan