LAPORAN PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJECT

PERIODE: SEMESTER GENAP 2024/2025

NAMA : NESSA NOVIANTI

NIM : 24104410007

PERIODE : SEMESTER GENAP 2024/2025

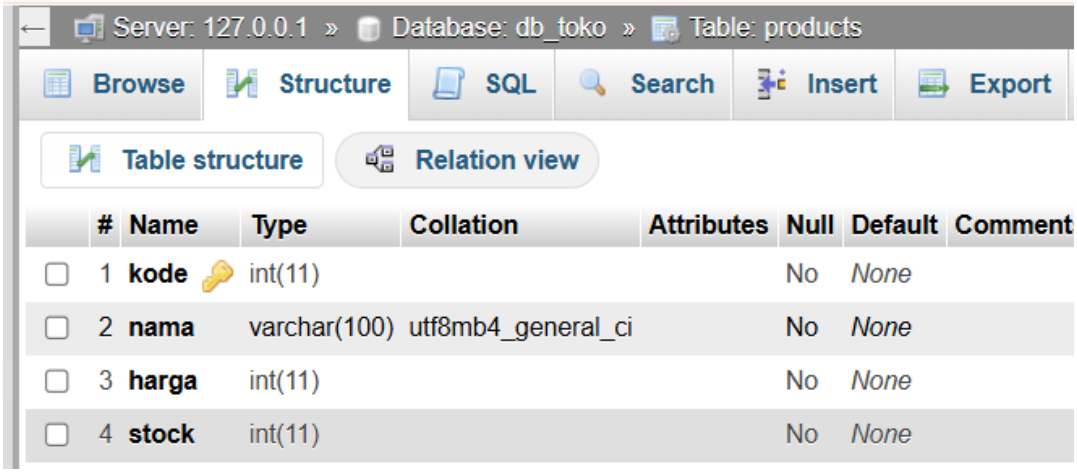
PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM BALITAR

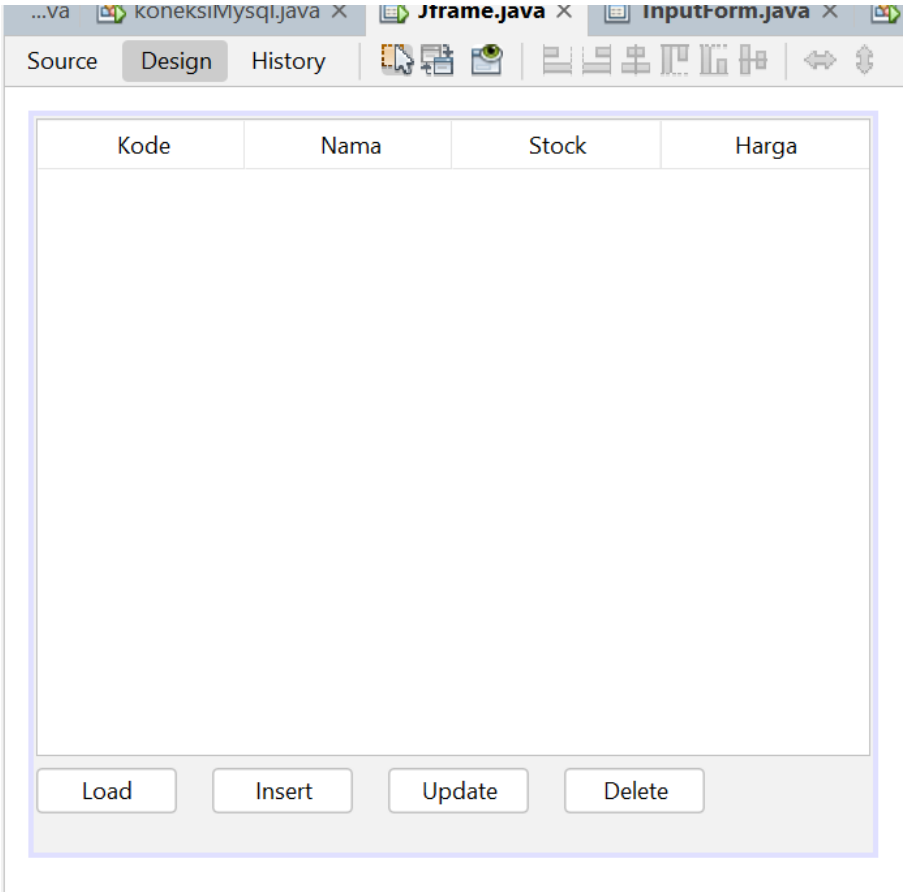
2024/2025

1. **Pembuatan Database dengan nama db_toko dan table product**

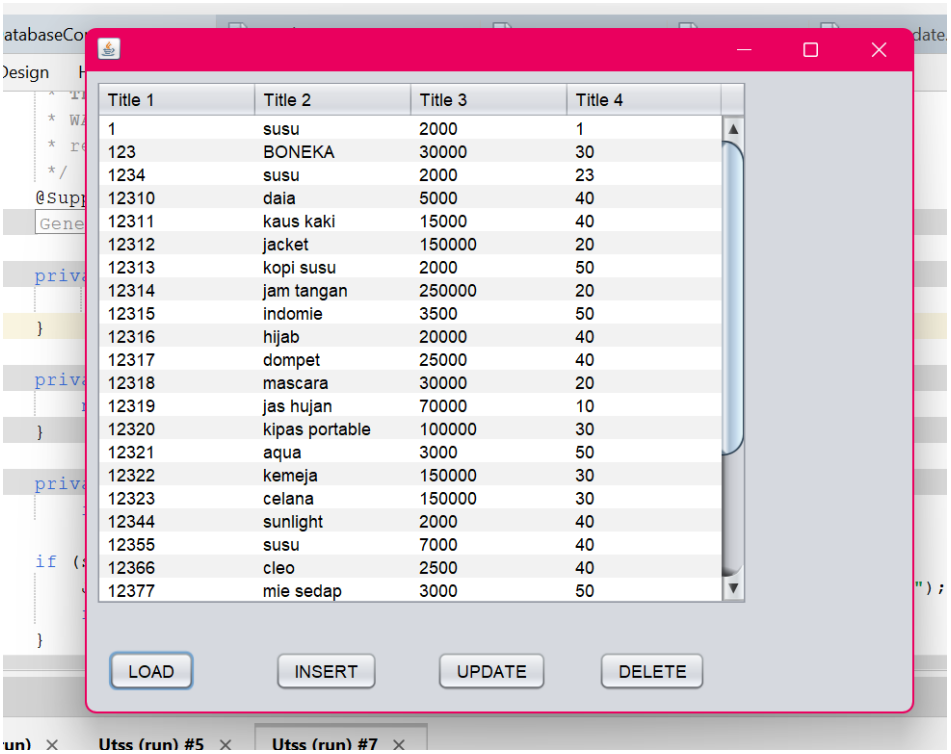


#	Name	Type	Collation	Attributes	Null	Default	Comment
1	**kode**	int(11)			No	*None*	
2	**nama**	varchar(100)	utf8mb4_general_ci		No	*None*	
3	**harga**	int(11)			No	*None*	
4	**stock**	int(11)			No	*None*	

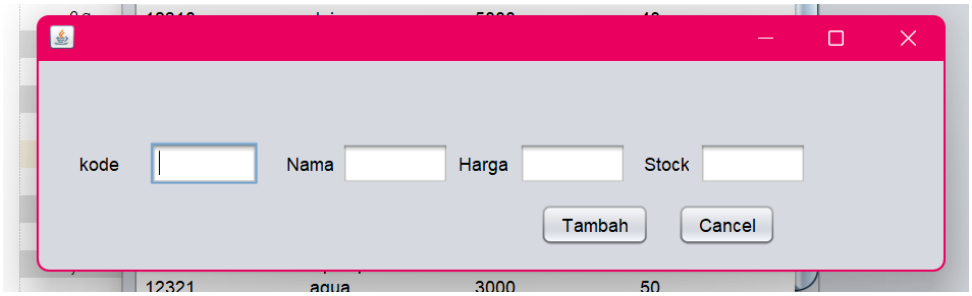
2. **Pembuatan JFrame yang dilengkapi dengan 4 JButton dan 1 JTable**



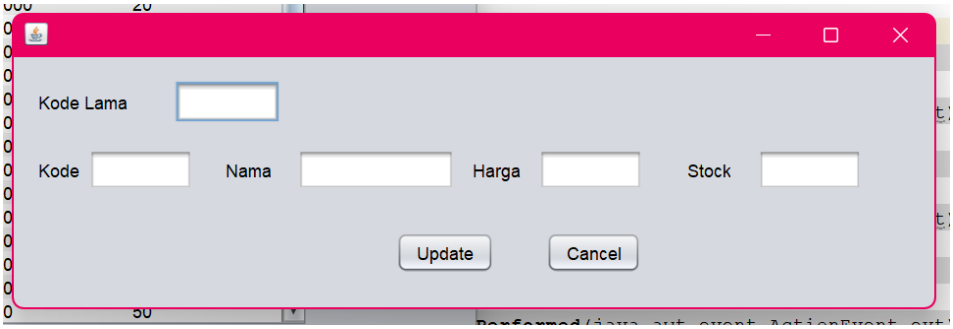
• **Load: seleksi semua data di tabel produk**



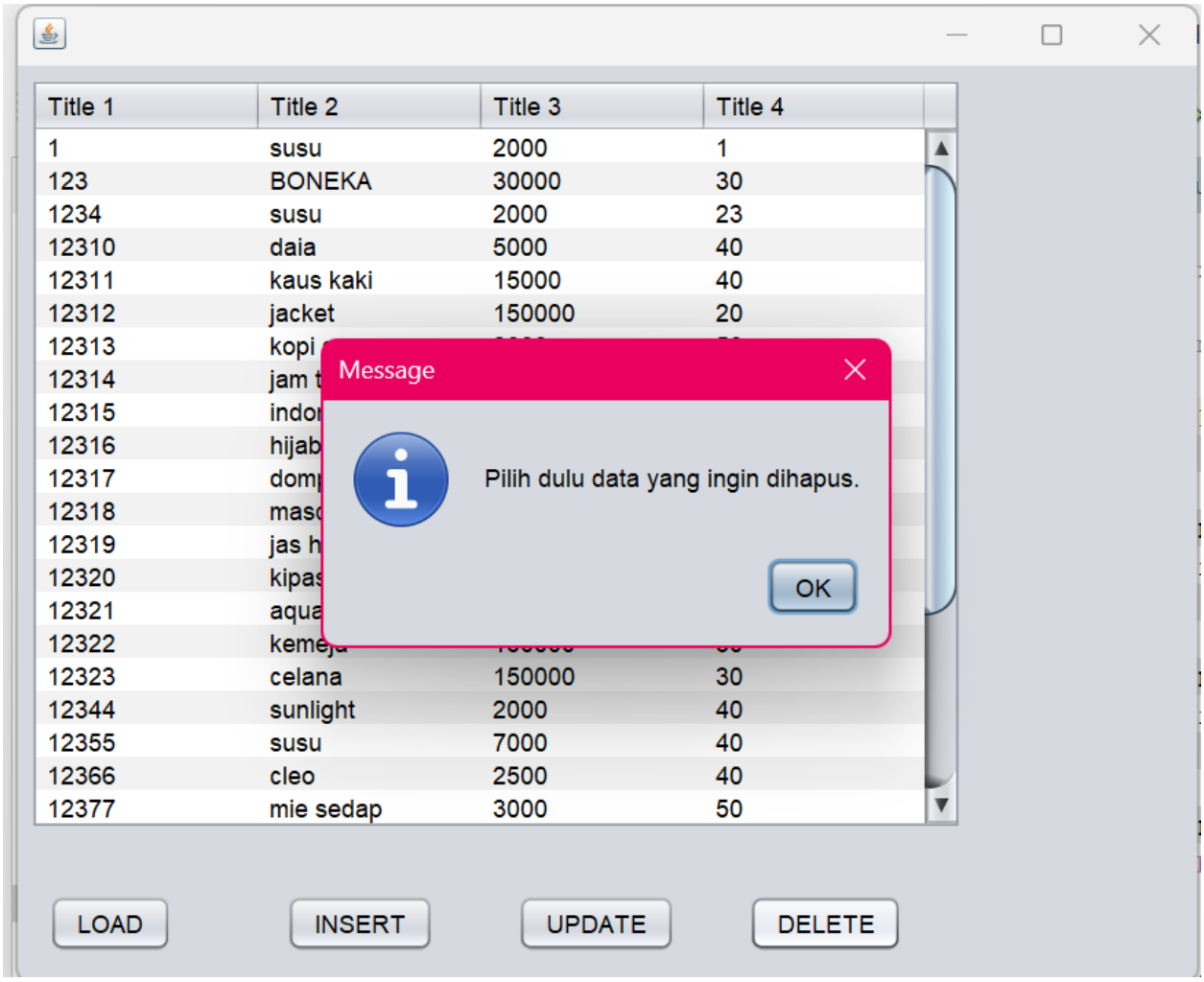
• **Insert: memasukkan satu produk baru**



• **Update: menyeleksi salah satu isi tabel dan melakukan update**



- **Delete: menyeleksi salah satu isi tabel dan melakukan delete**



3. Penerapan enkapsulasi, Inheritance, try-catch SQLException, GUI Swing project

a. Enkapsulasi

```
17      @Override
18      public void simpanKeDatabase() {
19          try {
20              Connection conn = DatabaseConnection.getConnection();
21              PreparedStatement stmt = conn.prepareStatement(
22                  "INSERT INTO product (kode, nama, harga, stock) VALUES (?, ?, ?, ?)"
23              );
24              stmt.setInt(1, getKode());
25              stmt.setString(2, getNama());
26              stmt.setInt(3, getHarga());
27              stmt.setInt(4, getStok());
28              stmt.executeUpdate();
29              System.out.println("Produk berhasil disimpan ke database.");
30          } catch (SQLException e) {
31              System.err.println("Gagal menyimpan produk: " + e.getMessage());
32          }
33      }
34
35      Object gettock() {
36          throw new UnsupportedOperationException("Not supported yet."); // Generated from nk
37      }
38
39      Object getStock() {
40          throw new UnsupportedOperationException("Not supported yet."); // Generated from nk
41      }
42  }
```

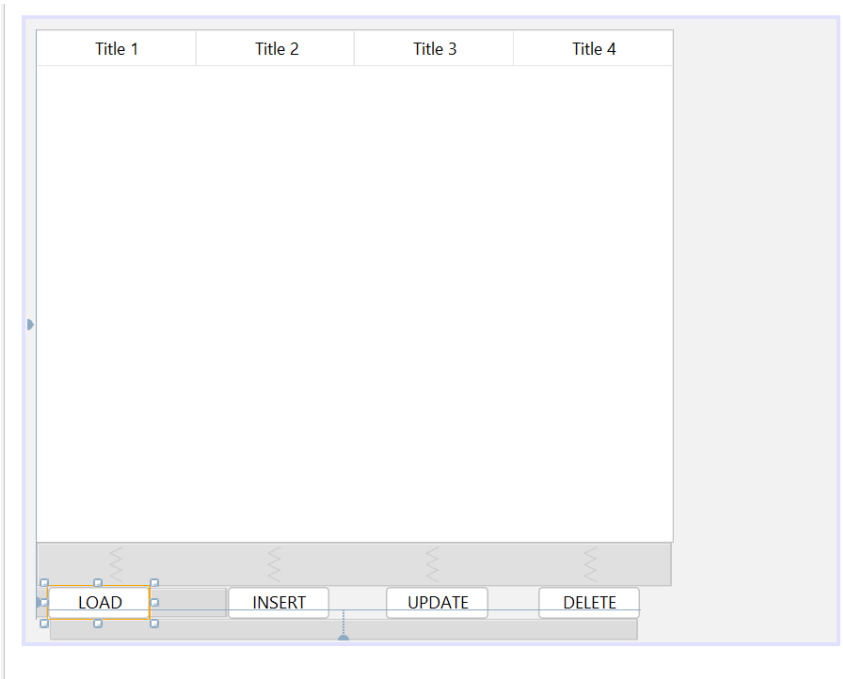
b. Inheritance

```
public class Product extends AbstrackProduk {

    public Product(int kode, String nama, int harga, int stok) {
        super(kode, nama, harga, stok);
    }

    @Override
    public void simpanKeDatabase() {
        try {
            Connection conn = DatabaseConnection.getConnection();
            PreparedStatement stmt = conn.prepareStatement(
                "INSERT INTO product (kode, nama, harga, stock) VALUES (?, ?, ?, ?)"
            );
            stmt.setInt(1, getKode());
            stmt.setString(2, getNama());
            stmt.setInt(3, getHarga());
            stmt.setInt(4, getStok());
            stmt.executeUpdate();
            System.out.println("Produk berhasil disimpan ke database.");
        } catch (SQLException e) {
            System.err.println("Gagal menyimpan produk: " + e.getMessage());
        }
    }
```

c. try-catch SQLException

```
    @Override
    public void simpanKeDatabase() {
        try {
            Connection conn = DatabaseConnection.getConnection();
            PreparedStatement stmt = conn.prepareStatement(
                "INSERT INTO product (kode, nama, harga, stock) VALUES (?, ?, ?, ?)"
            );
            stmt.setInt(1, getKode());
            stmt.setString(2, getNama());
            stmt.setInt(3, getHarga());
            stmt.setInt(4, getStok());
            stmt.executeUpdate();
            System.out.println("Produk berhasil disimpan ke database.");
        } catch (SQLException e) {
            System.err.println("Gagal menyimpan produk: " + e.getMessage());
        }
    }
```

d. GUI Swing



4. Penjelasan codingan dan output

A. Kelas kasir

/*

- Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
- Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template

*/ package UTS;

import java.sql.Connection; import java.sql.PreparedStatement; import java.sql.SQLException; import javax.swing.JOptionPane; import javax.swing.table.DefaultTableModel; import java.util.Vector; import javax.swing.table.DefaultTableModel; import java.sql.Statement; import java.sql.ResultSet;

/** *

- @author ASUS

*/ public class Kasir2 extends javax.swing.JFrame {

```
/**
 * Creates new form Kasir2
 */
public Kasir2() {
    initComponents();
}

/**
 * This method is called from within the constructor to initialize the
form.
 * WARNING: Do NOT modify this code. The content of this method is
always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {
```

```
    jScrollPane1 = new javax.swing.JScrollPane();
    jTable1 = new javax.swing.JTable();
    LOAD = new javax.swing.JButton();
    INSERT = new javax.swing.JButton();
    UPDATE = new javax.swing.JButton();
    DELETE = new javax.swing.JButton();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jTable1.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null}
        },
        new String [] {
            "Title 1", "Title 2", "Title 3", "Title 4"
        }
    ));
    jScrollPane1.setViewportView(jTable1);

    LOAD.setText("LOAD");
    LOAD.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            LOADActionPerformed(evt);
        }
    });

    INSERT.setText("INSERT");
    INSERT.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            INSERTActionPerformed(evt);
        }
    });

    UPDATE.setText("UPDATE");
```

```
    UPDATE.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            UPDATEActionPerformed(evt);
        }
    });

    DELETE.setText("DELETE");
    DELETE.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            DELETEActionPerformed(evt);
        }
    });

    javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment
.LEADING)
                .addComponent(jScrollPane1,
javax.swing.GroupLayout.PREFERRED_SIZE, 452,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGroup(layout.createSequentialGroup()
                    .addGap(9, 9, 9)
                    .addComponent(LOAD)
                    .addGap(55, 55, 55)
                    .addComponent(INSERT)
                    .addGap(40, 40, 40)
                    .addComponent(UPDATE)
                    .addGap(35, 35, 35)
                    .addComponent(DELETE)))
            .addContainerGap(114, Short.MAX_VALUE))
    );
    layout.setVerticalGroup(
```

```
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jScrollPane1,
javax.swing.GroupLayout.PREFERRED_SIZE, 364,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
31, Short.MAX_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment
.BASELINE)
                .addComponent(LOAD)
                .addComponent(INSERT)
                .addComponent(UPDATE)
                .addComponent(DELETE))
            .addGap(15, 15, 15))
    );

    pack();
}// </editor-fold>

private void INSERTActionPerformed(java.awt.event.ActionEvent evt) {
     new FormInput().setVisible(true);
}

private void UPDATEActionPerformed(java.awt.event.ActionEvent evt) {
    new FormUpdate().setVisible(true);
}

private void DELETEActionPerformed(java.awt.event.ActionEvent evt) {
    int selectedRow = jTable1.getSelectedRow();

if (selectedRow == -1) {
    JOptionPane.showMessageDialog(this, "Pilih dulu data yang ingin
dihapus.");
    return;
}
```

```
int confirm = JOptionPane.showConfirmDialog(this, "Yakin ingin
menghapus data ini?", "Konfirmasi Hapus", JOptionPane.YES_NO_OPTION);

if (confirm == JOptionPane.YES_OPTION) {
    try {
       int id = Integer.parseInt(jTable1.getValueAt(selectedRow,
0).toString()); // Ambil ID dari kolom pertama (index 0)
        Connection conn = DatabaseConnection.getConnection();
        String sql = "DELETE FROM product WHERE kode = ?";
        PreparedStatement stmt = conn.prepareStatement(sql);
        stmt.setInt(1, id);
        stmt.executeUpdate();

        // Hapus dari JTable juga
         DefaultTableModel model = (DefaultTableModel)
jTable1.getModel();
        model.removeRow(selectedRow);

        JOptionPane.showMessageDialog(this, "Data berhasil dihapus!");
    } catch (SQLException e) {
        JOptionPane.showMessageDialog(this, "Gagal hapus data: " +
e.getMessage());
    }
}
}

private void LOADActionPerformed(java.awt.event.ActionEvent evt) {



try (Connection conn = DatabaseConnection.DatabaseConnection(); Statement stmt =
conn.createStatement(); ResultSet rs = stmt.executeQuery("SELECT * FROM products")) {

        DefaultTableModel model = (DefaultTableModel)
jTable1.getModel();
         model.setRowCount(0);
        while (rs.next()) {
            Vector<Object> row = new Vector<>();
            row.add(rs.getInt("kode"));
```

```
            row.add(rs.getString("nama"));
            row.add(rs.getInt("harga"));
            row.add(rs.getInt("stock"));
            model.addRow(row);
        }

    } catch (SQLException ex) {
        JOptionPane.showMessageDialog(this, "Error saat memuat data: "
+ ex.getMessage(), "Database Error", JOptionPane.ERROR_MESSAGE);
        ex.printStackTrace();
    }
}
```

```
/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel
setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with
the default look and feel.
     * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.ht
ml
     */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {
```

```
java.util.logging.Logger.getLogger(Kasir2.class.getName()).log(java.ut
il.logging.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Kasir2.class.getName()).log(java.ut
il.logging.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Kasir2.class.getName()).log(java.ut
il.logging.Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Kasir2.class.getName()).log(java.ut
il.logging.Level.SEVERE, null, ex);
    }
    //</editor-fold>

    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Kasir2().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton DELETE;
private javax.swing.JButton INSERT;
private javax.swing.JButton LOAD;
private javax.swing.JButton UPDATE;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
// End of variables declaration


}
```

- ANALISA
  Kode tersebut merupakan implementasi aplikasi kasir berbasis Java Swing yang menyediakan tampilan grafis (GUI) untuk mengelola data produk pada sebuah database. Di dalam kelas **Kasir2**, yang terdapat tabel (JTable) yang digunakan untuk menampilkan daftar produk, serta empat tombol utama: **LOAD**, **INSERT**, **UPDATE**, dan **DELETE**.
  - **Tombol LOAD** akan mengambil seluruh data produk dari database (tabel products) menggunakan koneksi JDBC, lalu menampilkannya ke dalam JTable. Ini digunakan untuk menjalankan query SQL SELECT, lalu setiap data yang didapat akan dimasukkan ke model tabel.
  - **Tombol INSERT** akan membuka form input baru (FormInput) yang memungkinkan pengguna menambah data produk baru ke database.
  - **Tombol UPDATE** akan membuka form update (FormUpdate) yang memungkinkan pengguna mengubah data produk yang sudah ada.
  - **Tombol DELETE** akan menghapus data produk yang dipilih pada JTable. Sebelum menghapus, akan muncul dialog konfirmasi. Jika pengguna setuju, maka aplikasi akan mengambil kode produk dari baris yang dipilih, menjalankan query SQL DELETE ke database, dan menghapus baris tersebut dari JTable.

  Semua operasi database dilakukan menggunakan koneksi yang diambil dari kelas DatabaseConnection. Setiap aksi tombol di-handle dengan event handler masing-masing. Jika terjadi error saat operasi database (misal gagal koneksi atau query), akan muncul pesan error menggunakan JOptionPane.

B. FORM INPUT

```
package UTS;

import UTS.DatabaseConnection; import java.sql.Connection; import
java.sql.PreparedStatement; import java.sql.SQLException; import javax.swing.JOptionPane;

/*

- Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change
  this license
- Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this
  template

*/
```

```
/** *
```

- @author ASUS

```
*/ public class FormInput extends javax.swing.JFrame {

/**
 * Creates new form FormInput
 */
public FormInput() {
    initComponents();
}

/**
 * This method is called from within the constructor to initialize the
form.
 * WARNING: Do NOT modify this code. The content of this method is
always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jPanel1 = new javax.swing.JPanel();
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jLabel4 = new javax.swing.JLabel();
    txtKode = new javax.swing.JTextField();
    txtNama = new javax.swing.JTextField();
    txtHarga = new javax.swing.JTextField();
    txtStock = new javax.swing.JTextField();
    jButton1 = new javax.swing.JButton();
    jButton2 = new javax.swing.JButton();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jLabel1.setText("kode");
```

```
    jLabel2.setText("Nama");

    jLabel3.setText("Harga");

    jLabel4.setText("Stock");

    txtKode.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            txtKodeActionPerformed(evt);
        }
    });

    txtNama.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            txtNamaActionPerformed(evt);
        }
    });

    jButton1.setText("Tambah");
    jButton1.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton1ActionPerformed(evt);
        }
    });

    jButton2.setText("Cancel");
    jButton2.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton2ActionPerformed(evt);
        }
    });

    javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
    jPanel1.setLayout(jPanel1Layout);
    jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LE
```

```
ADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(27, 27, 27)
            .addComponent(jLabel1,
javax.swing.GroupLayout.PREFERRED_SIZE, 40,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(txtKode,
javax.swing.GroupLayout.PREFERRED_SIZE, 71,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addComponent(jLabel2)
            .addGap(5, 5, 5)
            .addComponent(txtNama,
javax.swing.GroupLayout.PREFERRED_SIZE, 71,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jLabel3)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(txtHarga,
javax.swing.GroupLayout.PREFERRED_SIZE, 71,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(11, 11, 11)
            .addComponent(jLabel4)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(txtStock,
javax.swing.GroupLayout.PREFERRED_SIZE, 71,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(90, Short.MAX_VALUE))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup()
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
            .addComponent(jButton1)
            .addGap(18, 18, 18)
```

```
            .addComponent(jButton2)
            .addGap(110, 110, 110))
    );
    jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LE
ADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(53, 53, 53)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Al
ignment.BASELINE)
                .addComponent(jLabel1)
                .addComponent(jLabel2)
                .addComponent(jLabel3)
                .addComponent(jLabel4)
                .addComponent(txtKode,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(txtNama,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(txtHarga,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(txtStock,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Al
ignment.BASELINE)
                .addComponent(jButton2)
                .addComponent(jButton1))
```

```
            .addContainerGap(16, Short.MAX_VALUE))
    );

    javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    );
    layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    );

    pack();
}// </editor-fold>

private void txtKodeActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
```

try (Connection conn = DatabaseConnection.DatabaseConnection()) { String sql = "INSERT INTO products (kode, nama, harga, stock) VALUES (?, ?, ?, ?)"; try (PreparedStatement pstmt = conn.prepareStatement(sql)) { int kode = Integer.parseInt(txtKode.getText()); String nama = txtNama.getText(); int harga = Integer.parseInt(txtHarga.getText()); int stock = Integer.parseInt(txtStock.getText());

```
            pstmt.setInt(1, kode);
             pstmt.setString(2, nama);
             pstmt.setInt(3, harga);
```

```
            pstmt.setInt(4, stock);

            int rowsAffected = pstmt.executeUpdate();
            if (rowsAffected > 0) {
                JOptionPane.showMessageDialog(this, "Data berhasil
ditambahkan!", "Sukses", JOptionPane.INFORMATION_MESSAGE);
               this.dispose();
            } else {
                JOptionPane.showMessageDialog(this, "Gagal menambahkan
data.", "Error", JOptionPane.ERROR_MESSAGE);
            }
        }
    } catch (SQLException | NumberFormatException ex) {
        JOptionPane.showMessageDialog(this, "Error: " +
ex.getMessage(), "Input Error", JOptionPane.ERROR_MESSAGE);
        ex.printStackTrace();
    }

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
   this.dispose();
}

private void txtNamaActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel
setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with
the default look and feel.
     * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.ht
```

```
ml
     */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(FormInput.class.getName()).log(java
.util.logging.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(FormInput.class.getName()).log(java
.util.logging.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(FormInput.class.getName()).log(java
.util.logging.Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(FormInput.class.getName()).log(java
.util.logging.Level.SEVERE, null, ex);
    }
    //</editor-fold>

    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new FormInput().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
```

```
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JPanel jPanel1;
private javax.swing.JTextField txtHarga;
private javax.swing.JTextField txtKode;
private javax.swing.JTextField txtNama;
private javax.swing.JTextField txtStock;
// End of variables declaration
```

}

- ANALISA
  Kelas FormInput dalam adalah formulir GUI berbasis Swing untuk menambahkan data produk baru (kode, nama, harga, stok) ke tabel products di database MySQL menggunakan JDBC. Formulir ini memiliki empat text field (txtKode, txtNama, txtHarga, txtStock) dan dua tombol: **Tambah** untuk menyimpan data ke database melalui query **INSERT** dengan **PreparedStatement** dan **Cancel** untuk menutup formulir. Data diambil dari text field, divalidasi (kode, harga, stok harus angka), dan disimpan menggunakan koneksi dari **DatabaseConnection.DatabaseConnection**(), dengan **try-with-resources** untuk menutup sumber daya secara otomatis. Pesan sukses atau error ditampilkan via **JOptionPane.**

C. FORM UPDATE

/*

- Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
- Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template

/ package UTS; import java.sql.Connection; import java.sql.PreparedStatement; import java.sql.SQLException; import javax.swing.JOptionPane; / *

- @author ASUS

*/ public class FormUpdate extends javax.swing.JFrame {

```
/**
 * Creates new form FormUpdate
 */
public FormUpdate() {
    initComponents();
}

/**
 * This method is called from within the constructor to initialize the
form.
 * WARNING: Do NOT modify this code. The content of this method is
always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    TfKodeBaru = new javax.swing.JTextField();
    tfNama = new javax.swing.JTextField();
    jLabel3 = new javax.swing.JLabel();
    tfHarga = new javax.swing.JTextField();
    jLabel4 = new javax.swing.JLabel();
    tfStock = new javax.swing.JTextField();
    jButton1 = new javax.swing.JButton();
    jButton2 = new javax.swing.JButton();
    jLabel5 = new javax.swing.JLabel();
    TfKodeLama = new javax.swing.JTextField();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jLabel1.setText("Kode");
```

```
    jLabel2.setText("Nama");

    TfKodeBaru.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            TfKodeBaruActionPerformed(evt);
        }
    });

    jLabel3.setText("Harga");

    jLabel4.setText("Stock");

    jButton1.setText("Update");
    jButton1.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton1ActionPerformed(evt);
        }
    });

    jButton2.setText("Cancel");
    jButton2.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton2ActionPerformed(evt);
        }
    });

    jLabel5.setText("Kode Lama");

    javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment
.LEADING)
                .addGroup(layout.createSequentialGroup()
```

```
                    .addGap(260, 260, 260)
                    .addComponent(jButton1)
                    .addGap(35, 35, 35)
                    .addComponent(jButton2))
                .addGroup(layout.createSequentialGroup()
                    .addGap(17, 17, 17)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment
.LEADING)
                        .addGroup(layout.createSequentialGroup()
                            .addComponent(jLabel5)
                            .addGap(32, 32, 32)
                            .addComponent(TfKodeLama,
javax.swing.GroupLayout.PREFERRED_SIZE, 71,
javax.swing.GroupLayout.PREFERRED_SIZE))
                        .addGroup(layout.createSequentialGroup()
                            .addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                            .addComponent(TfKodeBaru,
javax.swing.GroupLayout.PREFERRED_SIZE, 71,
javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addGap(22, 22, 22)
                            .addComponent(jLabel2)
                            .addGap(18, 18, 18)
                            .addComponent(tfNama,
javax.swing.GroupLayout.PREFERRED_SIZE, 107,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                            .addComponent(jLabel3)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                            .addComponent(tfHarga,
javax.swing.GroupLayout.PREFERRED_SIZE, 71,
javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addGap(30, 30, 30)
                            .addComponent(jLabel4)
                            .addGap(18, 18, 18)
```

```
                            .addComponent(tfStock,
javax.swing.GroupLayout.PREFERRED_SIZE, 71,
javax.swing.GroupLayout.PREFERRED_SIZE)))))
            .addContainerGap(50, Short.MAX_VALUE))
    );
    layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(16, 16, 16)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment
.BASELINE)
                .addComponent(jLabel5)
                .addComponent(TfKodeLama,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment
.BASELINE)
                .addComponent(jLabel1)
                .addComponent(jLabel2)
                .addComponent(TfKodeBaru,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(tfNama,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel3)
                .addComponent(tfHarga,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel4)
                .addComponent(tfStock,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
28, Short.MAX_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment
.BASELINE)
                .addComponent(jButton1)
                .addComponent(jButton2))
            .addGap(24, 24, 24))
    );

    pack();
}// </editor-fold>

private void TfKodeBaruActionPerformed(java.awt.event.ActionEvent evt)
{
    // TODO add your handling code here:
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    this.dispose();
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
int kodeBaru = Integer.parseInt(TfKodeBaru.getText());
int kodeLama = Integer.parseInt(TfKodeLama.getText());
String nama = tfNama.getText();
int harga = Integer.parseInt(tfHarga.getText());
int stok = Integer.parseInt(tfStock.getText());
```

try { Connection conn = DatabaseConnection.getConnection(); // pastikan class koneksi sudah sesuai String sql = "UPDATE products SET kode = ?, nama = ?, harga = ?, stock = ? WHERE kode = ?"; PreparedStatement stmt = conn.prepareStatement(sql); stmt.setInt(1, kodeBaru); stmt.setString(2, nama); stmt.setInt(3, harga); stmt.setInt(4, stok); stmt.setInt(5, kodeLama);

```
int rowsUpdated = stmt.executeUpdate();
if (rowsUpdated > 0)
{
    JOptionPane.showMessageDialog(this, "Data berhasil diupdate.");
    this.dispose();
} else {
    JOptionPane.showMessageDialog(this, "Data gagal diupdate. Kode
lama tidak ditemukan.");
}

stmt.close();
conn.close();
```

} catch (NumberFormatException e) { JOptionPane.showMessageDialog(this, "Harga dan Stock harus berupa angka."); } catch (SQLException e) { JOptionPane.showMessageDialog(this, "Kesalahan database: " + e.getMessage()); } }

```
/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel
setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with
the default look and feel.
     * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.ht
ml
     */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
```

```
    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(FormUpdate.class.getName()).log(jav
a.util.logging.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(FormUpdate.class.getName()).log(jav
a.util.logging.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(FormUpdate.class.getName()).log(jav
a.util.logging.Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(FormUpdate.class.getName()).log(jav
a.util.logging.Level.SEVERE, null, ex);
    }
    //</editor-fold>

    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new FormUpdate().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JTextField TfKodeBaru;
private javax.swing.JTextField TfKodeLama;
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JTextField tfHarga;
private javax.swing.JTextField tfNama;
```

```
private javax.swing.JTextField tfStock;
// End of variables declaration


}
```

- ANALISA

  Kode tersebut merupakan program Java Swing yang menampilkan form untuk mengupdate data produk pada database, dengan input field untuk kode lama, kode baru, nama, harga, dan stok produk. Saat tombol **Update** ditekan, data dari form akan diambil, lalu dilakukan validasi dan eksekusi perintah SQL UPDATE untuk mengubah data produk berdasarkan kode lama di database; jika berhasil, pesan sukses ditampilkan dan form ditutup, sedangkan jika gagal atau terjadi error (misal input bukan angka atau kesalahan database), akan muncul pesan error yang sesuai. Tombol **Cancel** berfungsi untuk menutup form tanpa melakukan perubahan apa pun.

D. Data base Connection

```
/*
* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
*/
package UTS;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class DatabaseConnection {
```

```
    private static final String URL = "jdbc:mysql://localhost:3306/db_toko";
    private static final String USER = "root"; // Ganti dengan username MySQL Anda
    private static final String PASSWORD = ""; // Ganti dengan password MySQL Anda
    private static Connection connection = null;

    // Metode untuk mendapatkan koneksi
    public static Connection DatabaseConnection() throws SQLException {
        if (connection == null || connection.isClosed()) {
            try {
                // Load driver
                Class.forName("com.mysql.cj.jdbc.Driver");
                // Buat koneksi
                connection = DriverManager.getConnection(URL, USER, PASSWORD);
                System.out.println("Koneksi ke database berhasil!");
            } catch (ClassNotFoundException e) {
                System.err.println("JDBC Driver tidak ditemukan. Pastikan file JAR sudah ditambahkan
ke classpath.");
                e.printStackTrace();
            }
        }
        return connection;
    }

    // Metode untuk menutup koneksi
    public static void closeConnection() {
        if (connection != null) {
```

```
            try {
                connection.close();
                System.out.println("Koneksi database ditutup.");
            } catch (SQLException e) {
                e.printStackTrace();
            }
        }
    }

    static Connection getConnection() {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver"); // Pastikan driver sudah ada
            return DriverManager.getConnection("jdbc:mysql://localhost:3306/db_toko", "username",
"password");
    } catch (Exception e) {
            e.printStackTrace();
            return null;
    }
    }

}
```

- ANALISA
  Kelas DatabaseConnection digunakan untuk mengelola koneksi ke database MySQL (db_toko) menggunakan JDBC. Kelas ini menyediakan dua metode untuk mendapatkan koneksi (DatabaseConnection dan getConnection) dan satu metode untuk menutup koneksi (closeConnection). Metode DatabaseConnection menggunakan pola singleton sederhana untuk mengelola koneksi tunggal, sedangkan getConnection membuat koneksi baru setiap kali dipanggil.

E. Products

```
/*
* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
*/
package UTS;

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.SQLException;

public class Product extends AbstrackProduk {

    public Product(int kode, String nama, int harga, int stok) {
        super(kode, nama, harga, stok);
    }

    @Override
    public void simpanKeDatabase() {
        try {
            Connection conn = DatabaseConnection.getConnection();
            PreparedStatement stmt = conn.prepareStatement(
                "INSERT INTO product (kode, nama, harga, stock) VALUES (?, ?, ?, ?)"
```

```
            );
            stmt.setInt(1, getKode());
            stmt.setString(2, getNama());
            stmt.setInt(3, getHarga());
            stmt.setInt(4, getStok());
            stmt.executeUpdate();
            System.out.println("Produk berhasil disimpan ke database.");
        } catch (SQLException e) {
            System.err.println("Gagal menyimpan produk: " + e.getMessage());
        }
    }

    Object gettock() {
        throw new UnsupportedOperationException("Not supported yet."); // Generated from nbfs://nbhost/SystemFileSystem/Templates/Classes/Code/GeneratedMethodBody
    }

    Object getStock() {
        throw new UnsupportedOperationException("Not supported yet."); // Generated from nbfs://nbhost/SystemFileSystem/Templates/Classes/Code/GeneratedMethodBody
    }
}
```

- ANALISA
  Kelas ini merupakan implementasi dari kelas abstrak AbstrackProduk yang dirancang untuk menyimpan data produk (kode, nama, harga, stok) ke database yang telah disiapkan menggunakan JDBC. Metode simpanKeDatabase menangani penyimpanan data, sementara gettock dan getStock adalah stub yang belum diimplementasikan