

LAPORAN TUGAS
PEMROGRAMAN BERBASIS OBJEK
PERTEMUAN KEENAM 26 SEPTEMBER 2023



UIN SUNAN AMPEL
S U R A B A Y A

DOSEN PEMBIMBING

Bayu Adhi Nugroho, Ph.D.

(197905182014031001)

DISUSUN OLEH

Mochamad Roiyan Rintiarno

(09020622033)

UIN SUNAN AMPEL SURABAYA

TAHUN 2023

1. Tugas

→ Membuat CRUD sederhana menggunakan java dan PostgreSQL (Graphical User Interface), kemudian menambahkan perintah cetak dokumen

2. Pembahasan dan Isi

- a. Persiapan dan Langkah-langkah
 - 1. Menyiapkan file Source Code

Terdapat beberapa kelas yang nantinya akan dibuat, yakni kelas Output sebagai file utama kelas book, dan file Report untuk menampilkan hasil reportnya.

a. Kelas Book

```
1  /*
2  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3  * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
4  */
5  package Tampilan;
6
7  /**
8   *
9   * @author milea
10  */
11  public class Book {
12      private int id;
13      private String name;
14      private String pengarang;
15      private String jenis;
16
17      /**
18       * @return the id
19       */
20      public int getId() {
21          return id;
22      }
23
24      /**
25       * @param id the id to set
26       */
27      public void setId(int id) {
28          this.id = id;
29      }
30
31      /**
32       * @return the name
33       */
34      public String getName() {
35          return name;
36      }
37
38      /**
39       * @param name the name to set
40       */
41      public void setName(String name) {
42          this.name = name;
43      }
44
45      /**
46       * @return the pengarang
47       */
48      public String getPengarang() {
49          return pengarang;
50      }
51
52      /**
53       * @param pengarang the pengarang to set
54       */
55      public void setPengarang(String pengarang) {
56          this.pengarang = pengarang;
57      }
58
59      /**
60       * @return the jenis
61       */
62      public String getJenis() {
63          return jenis;
64      }
65
66      /**
67       * @param jenis the jenis to set
68       */
69      public void setJenis(String jenis) {
70          this.jenis = jenis;
71      }
72  }
73
```


b. File reportBuku

File reportBuku digunakan untuk mencetak hasil dari data yang telah ditambahkan sebelumnya.

➤ Designer



➤ Kode XML

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <jasperReport xmlns="http://jasperreports.sourceforge.net/jasperreports"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://jasperreports.sourceforge.net/jasperreports
    http://jasperreports.sourceforge.net/xsd/jasperreport.xsd" name="null" pageWidth="595" pageHeight="842"
    columnWidth="555" leftMargin="20" rightMargin="20" topMargin="20" bottomMargin="20" uid="3341c86-1297-
    4cb-07e0-b33291aa249">
3   <property name="ireport.zoom" value="1.0"/>
4   <property name="ireport.x" value="0"/>
5   <property name="ireport.y" value="0"/>
6   <style name="table">
7     <box>
8       <pen lineWidth="1.0" lineColor="#000000"/>
9     </box>
10  </style>
11  <style name="table_TH" mode="Opaque" backcolor="#CCB48F">
12    <box>
13      <pen lineWidth="0.5" lineColor="#000000"/>
14    </box>
15  </style>
16  <style name="table_CH" mode="Opaque" backcolor="#E6DCCF">
17    <box>
18      <pen lineWidth="0.5" lineColor="#000000"/>
19    </box>
20  </style>
21  <style name="table_TD" mode="Opaque" backcolor="#FFFFFF">
22    <box>
23      <pen lineWidth="0.5" lineColor="#000000"/>
24    </box>
25  </style>
26  <style name="table_1">
27    <box>
28      <pen lineWidth="1.0" lineColor="#000000"/>
29    </box>
30  </style>
31  <style name="table_1_TH" mode="Opaque" backcolor="#F0F0FF">
32    <box>
33      <pen lineWidth="0.5" lineColor="#000000"/>
34    </box>
35  </style>
36  <style name="table_1_CH" mode="Opaque" backcolor="#BFE1FF">
37    <box>
38      <pen lineWidth="0.5" lineColor="#000000"/>
39    </box>
40  </style>
41  <style name="table_1_TD" mode="Opaque" backcolor="#FFFFFF">
42    <box>
43      <pen lineWidth="0.5" lineColor="#000000"/>
44    </box>
45  </style>
46  <subDataset name="buku" uid="cd91f24c-9422-45a9-94a5-b6a329bf8a21">
47    <queryString language="SQL">
48      <![CDATA[select * from books;]]>
49    </queryString>
50    <field name="id" class="java.lang.Integer"/>
51    <field name="name" class="java.lang.String"/>
52    <field name="pengarang" class="java.lang.String"/>
53    <field name="jenis" class="java.lang.String"/>
54  </subDataset>
```

```

1  <queryString language="SQL">-
2  <![CDATA[SELECT * FROM books;]]>-
3  </queryString>-
4  <field name="id" class="java.lang.Integer"/>-
5  <field name="name" class="java.lang.String"/>-
6  <field name="pengarang" class="java.lang.String"/>-
7  <field name="jenis" class="java.lang.String"/>-
8  <background>-
9  <band splitType="Stretch"/>-
10 </background>-
11 <title>-
12 <band height="79" splitType="Stretch">-
13 <staticText>-
14 <reportElement x="92" y="11" width="395" height="68" uuid="b3548964-9423-4aef-bb32-
416868e89130"/>-
15 <textElement textAlignment="Center" verticalAlignment="Middle">-
16 <font size="20" isBold="true"/>-
17 </textElement>-
18 <text><![CDATA[DAFTAR BUKU
19 "Perputakaan Publik"]]></text>-
20 </staticText>-
21 </band>-
22 </title>-
23 <pageHeader>-
24 <band height="12" splitType="Stretch"/>-
25 </pageHeader>-
26 <columnHeader>-
27 <band height="55" splitType="Stretch">-
28 <componentElement>-
29 <reportElement key="table_1" style="table_1" x="35" y="8" width="478" height="58"
uuid="413360d5-4864-42eb-bc44-863338dbdcff"/>-
30 <jr:table xmlns:jr="http://jasperreports.sourceforge.net/jasperreports/components"
xmlns:schemaLocation="http://jasperreports.sourceforge.net/jasperreports/components
http://jasperreports.sourceforge.net/xsd/components.xsd">-
31 <datasetRun subDataset="buku" uuid="8dc38c1-bd9f-4598-8a29-8978fa93c3b7">-
32 <connectionExpression><![CDATA[{P(REPORT_CONNECTION)}]></connectionExpression>-
33 </datasetRun>-
34 <jr:column width="50" uuid="35ce84eb-1f1e-4b2f-92dd-5e7236636886">-
35 <jr:columnHeader style="table_1.CH" height="38" rowSpan="1">-
36 <staticText>-
37 <reportElement x="8" y="8" width="49" height="38" uuid="046421b1-fe9d-
438b-9d79-ae7d5fe774ff"/>-
38 <textElement textAlignment="Center" verticalAlignment="Middle"/>-
39 <text><![CDATA[{id}]]></text>-
40 </staticText>-
41 </jr:columnHeader>-
42 <jr:detailCell style="table_1.TD" height="20" rowSpan="1">-
43 <textField>-
44 <reportElement x="8" y="8" width="49" height="20" uuid="c42fe8f9-f888-
4b65-89eb-dc96963488c2"/>-
45 <textFieldExpression><![CDATA[{P(id)}]></textFieldExpression>-
46 </textField>-
47 </jr:detailCell>-
48 </jr:column>-
49 <jr:column width="145" uuid="c563dc99-8e65-41e6-aecd-348b910cfa40">-
50 <jr:columnHeader style="table_1.CH" height="38" rowSpan="1">-
51 <staticText>-
52 <reportElement x="8" y="8" width="145" height="38" uuid="0b502b45-98ba-
4f2a-88c3-76333316c989"/>-
53 <textElement textAlignment="Center" verticalAlignment="Middle"/>-
54 <text><![CDATA[{name}]]></text>-
55 </staticText>-
56 </jr:columnHeader>

```

```

1      <jr:detailCell style="table_1_TD" height="28" rowSpan="1">
2      <textField>
3      <reportElement x="0" y="0" width="145" height="28" uuid="e9bae84-a092-
4e98-9892-cbad43da578"/>
4      <textFieldExpression<![CDATA[${F(name)}]></textFieldExpression>
5      </textField>
6      </jr:detailCell>
7      </jr:column>
8      <jr:column width="192" uuid="23a18992-7384-4ba8-b286-15913289466c">
9      <jr:columnHeader style="table_1_CH" height="38" rowSpan="1">
10     <staticText>
11     <reportElement x="0" y="0" width="192" height="38" uuid="63cfed80-e480-
4ca7-9054-85f62415b9d7"/>
12     <textField textAlignment="Center" verticalAlignment="Middle"/>
13     <text>{{pengerang}}]&gt;&lt;/text&gt;
14     &lt;/staticText&gt;
15     &lt;/jr:columnHeader&gt;
16     &lt;jr:detailCell style="table_1_TD" height="28" rowSpan="1"&gt;
17     &lt;textField&gt;
18     &lt;reportElement x="0" y="0" width="192" height="28" uuid="de63a2b9-f649-
49ab-988d-f63f9358b7eb"/&gt;
19     &lt;textFieldExpression&lt;![CDATA[{{pengerang}}]&gt;&lt;/textFieldExpression&gt;
20     &lt;/textField&gt;
21     &lt;/jr:detailCell&gt;
22     &lt;/jr:column&gt;
23     &lt;jr:column width="98" uuid="39c4f643-a48d-42d1-aa77-647fe3588c68"&gt;
24     &lt;jr:columnHeader style="table_1_CH" height="38" rowSpan="1"&gt;
25     &lt;staticText&gt;
26     &lt;reportElement x="0" y="0" width="98" height="38" uuid="d38d240b-12fa-
42e6-9f78-8a47ebe92988"/&gt;
27     &lt;textField textAlignment="Center" verticalAlignment="Middle"/&gt;
28     &lt;text&gt;<![CDATA[{{jenis}}]&gt;&lt;/text&gt;
29     &lt;/staticText&gt;
30     &lt;/jr:columnHeader&gt;
31     &lt;jr:detailCell style="table_1_TD" height="28" rowSpan="1"&gt;
32     &lt;textField&gt;
33     &lt;reportElement x="0" y="0" width="98" height="28" uuid="579e3a8e-8bbd-
48e7-b4c2-dc45a1f4854f"/&gt;
34     &lt;textFieldExpression&lt;![CDATA[{{F(jenis)}}]&gt;&lt;/textFieldExpression&gt;
35     &lt;/textField&gt;
36     &lt;/jr:detailCell&gt;
37     &lt;/jr:column&gt;
38     &lt;/jr:table&gt;
39     &lt;/componentElement&gt;
40     &lt;/band&gt;
41     &lt;/columnHeader&gt;
42     &lt;detail&gt;
43     &lt;band height="5" splitType="Stretch"/&gt;
44     &lt;/detail&gt;
45     &lt;columnFooter&gt;
46     &lt;band height="6" splitType="Stretch"/&gt;
47     &lt;/columnFooter&gt;
48     &lt;pageFooter&gt;
49     &lt;band height="28" splitType="Stretch"&gt;
50     &lt;textField pattern="dd/MM/yyyy"&gt;
51     &lt;reportElement x="288" y="0" width="275" height="28" uuid="ba8ad427-125b-4358-bdfe-
1a74b680b2a9"/&gt;
52     &lt;textField textAlignment="Right"/&gt;
53     &lt;textFieldExpression&lt;![CDATA[{{D(cetak pada "new java.util.Date)}}]&gt;
54     &lt;/textFieldExpression&gt;
55     &lt;/textField&gt;
56     &lt;/band&gt;
57     &lt;/pageFooter&gt;
58     &lt;summary&gt;
59     &lt;band splitType="Stretch"/&gt;
60     &lt;/summary&gt;
61     &lt;/jasperReport&gt;
</pre>
</div>
<div data-bbox="228 563 371 582" data-label="Section-Header">
<h3>c. Kelas Output</h3>
</div>
<div data-bbox="258 598 889 670" data-label="Text">
<p>Untuk kelas Output sendiri kurang lebih hampir sama dengan sebelumnya, dikarenakan juga memuat yang namanya CRUD, tetapi ditambah satu fitur lagi, yakni mencetak dokumen dengan menggunakan package jasperReport.</p>
</div>
```

```

1 package Tampilan;
2
3 import java.sql.Connection;
4 import java.sql.DriverManager;
5 import java.sql.PreparedStatement;
6 import java.sql.ResultSet;
7 import java.sql.SQLException;
8 import java.sql.Statement;
9 import java.util.ArrayList;
10 import java.util.logging.Level;
11 import java.util.logging.Logger;
12 import javax.swing.JOptionPane;
13 import javax.swing.table.DefaultTableModel;
14
15 /**
16  *
17  * @author milea
18  */
19 public class Output extends javax.swing.JFrame {
20
21     private ArrayList<Book> dataBuku;
22
23     /**
24      * Creates new form Output
25      */
26     public Output() {
27         this.refresh();
28     }
29
30     private void refresh(){
31         try {
32             dataBuku = new ArrayList<>();
33             initComponents();
34             Connection conn = DriverManager.getConnection("jdbc:postgresql://localhost:5432/perpustakaan", "postgres", "postgres");
35             tampil(conn);
36         } catch (SQLException ex) {
37             Logger.getLogger(Output.class.getName()).log(Level.SEVERE, null, ex);
38         }
39     }
40
41     public void peringatan(String pesan) {
42         JOptionPane.showMessageDialog(rootPane, pesan);
43     }
44
45     private int masukkanData(Connection conn, int id, String name, String pengarang, String jenis) throws SQLException {
46         PreparedStatement pst = conn.prepareStatement("INSERT INTO books (id, name, pengarang, jenis) VALUES(?, ?, ?, ?)");
47         pst.setInt(1, id);
48         pst.setString(2, name);
49         pst.setString(3, pengarang);
50         pst.setString(4, jenis);
51         return pst.executeUpdate();
52     }
53
54     private int updateData(Connection conn, int id, String name, String pengarang, String jenis) throws SQLException {
55         PreparedStatement pst = conn.prepareStatement("UPDATE books SET name=?, pengarang=?, jenis=? WHERE id=?");
56         pst.setInt(4, id);
57         pst.setString(1, name);
58         pst.setString(2, pengarang);
59         pst.setString(3, jenis);
60         return pst.executeUpdate();
61     }
62
63     private int deleteData(Connection conn, int id) throws SQLException {
64         PreparedStatement pst = conn.prepareStatement("DELETE FROM books WHERE id=?");
65         pst.setInt(1, id);
66         return pst.executeUpdate();
67     }
68
69     private void resetField(){
70         inputID.setText("");
71         inputJudul.setText("");
72         inputJenisBuku.setText("");
73         inputPengarang.setText("");
74     }

```



```
1 private void tampil(Connection conn) {
2     dataBuku.clear();
3     try {
4         String sql = "select * from books ORDER BY id";
5         Statement st = conn.createStatement();
6         ResultSet rs = st.executeQuery(sql);
7         while (rs.next()) {
8             Book data = new Book();
9             data.setId(rs.getInt(1));
10            data.setName(String.valueOf(rs.getObject(2)));
11            data.setPengarang(String.valueOf(rs.getObject(3)));
12            data.setJenis(String.valueOf(rs.getObject(4)));
13            dataBuku.add(data);
14        }
15        DefaultTableModel model = (DefaultTableModel) tabelBuku.getModel();
16        model.setRowCount(0);
17        for (Book data : dataBuku) {
18            Object[] baris = new Object[4];
19            baris[0] = data.getId();
20            baris[1] = data.getName();
21            baris[2] = data.getPengarang();
22            baris[3] = data.getJenis();
23            model.addRow(baris);
24        }
25    } catch (SQLException ex) {
26        Logger.getLogger(Output.class.getName()).log(Level.SEVERE, null, ex);
27    }
28 }
```

[illegible]

```

1 private void inputIDActionPerformed(java.awt.event.ActionEvent evt) {
2     // TODO add your handling code here:
3 }
4
5 private void inputPengarangActionPerformed(java.awt.event.ActionEvent evt) {
6     // TODO add your handling code here:
7 }
8
9 private void inputJudulActionPerformed(java.awt.event.ActionEvent evt) {
10    // TODO add your handling code here:
11 }
12
13 private void inputJenisBukuActionPerformed(java.awt.event.ActionEvent evt) {
14    // TODO add your handling code here:
15 }
16
17 private void btnInsertActionPerformed(java.awt.event.ActionEvent evt) {
18    // TODO add your handling code here:
19    int id = Integer.parseInt(inputID.getText().trim());
20    String name = inputJudul.getText();
21    String pengarang = inputPengarang.getText();
22    String jenis = inputJenisBuku.getText();
23    if (!name.isEmpty() && !pengarang.isEmpty() && !jenis.isEmpty() && id > 0) {
24        try {
25            // TODO add your handling code here:
26            Connection conn = DriverManager.getConnection("jdbc:postgresql://localhost:5432/perpustakaan", "postgres", "postgres");
27            int k = masukkanData(conn, id, name, pengarang, jenis);
28            if (k > 0) {
29                tampil(conn);
30                this.peringatan("Simpan Berhasil");
31            } else {
32                this.peringatan("Simpan Gagal");
33            }
34        } catch (SQLException ex) {
35            Logger.getLogger(Output.class.getName()).log(Level.SEVERE, null, ex);
36        }
37    } else {
38        this.peringatan("Input Data Buku Gagal");
39    }
40    this.resetField();
41 }

```

```

1 private void btnRefreshActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btnRefreshActionPerformed
2     // TODO add your handling code here:
3     this.resetField();
4     this.refresh();
5 } //GEN-LAST:event_btnRefreshActionPerformed
6
7 private void btnUpdateActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btnUpdateActionPerformed
8     // TODO add your handling code here:
9     int id = Integer.parseInt(inputID.getText().trim());
10    String name = inputJudul.getText();
11    String pengarang = inputPengarang.getText();
12    String jenis = inputJenisBuku.getText();
13    if (!name.isEmpty() && !pengarang.isEmpty() && !jenis.isEmpty() && id > 0) {
14        try {
15            // TODO add your handling code here:
16            Connection conn = DriverManager.getConnection("jdbc:postgresql://localhost:5432/perpustakaan", "postgres", "postgres");
17            int k = updateData(conn, id, name, pengarang, jenis);
18            if (k > 0) {
19                tampil(conn);
20                this.peringatan("Update Berhasil");
21            } else {
22                this.peringatan("Update Gagal");
23            }
24        } catch (SQLException ex) {
25            Logger.getLogger(Output.class.getName()).log(Level.SEVERE, null, ex);
26        }
27    } else {
28        this.peringatan("Update Data Mahasiswa Gagal");
29    }
30    this.resetField();
31 }

```

```

1 import net.sf.jasperreports.engine.JRException;
2 import net.sf.jasperreports.engine.JasperCompileManager;
3 import net.sf.jasperreports.engine.JasperFillManager;
4 import net.sf.jasperreports.engine.JasperPrint;
5 import net.sf.jasperreports.engine.JasperReport;
6 import net.sf.jasperreports.view.JasperViewer;
7 private void btnCetakActionPerformed(java.awt.event.ActionEvent evt) {
8     // TODO add your handling code here:
9     try {
10        // TODO add your handling code here:
11        Connection conn = DriverManager.getConnection("jdbc:postgresql://localhost:5432/perpustakaan", "postgres", "postgres");
12
13        String jrxmlFile = new String("src/Tampilan/reportCoba.jrxml");
14        JasperReport jr = JasperCompileManager.compileReport(jrxmlFile);
15        JasperPrint jp = JasperFillManager.fillReport(jr, null, conn);
16        JasperViewer.viewReport(jp, false);
17    } catch (JRException ex) {
18        Logger.getLogger(Output.class.getName()).log(Level.SEVERE, null, ex);
19    } catch (SQLException ex) {
20        Logger.getLogger(Output.class.getName()).log(Level.SEVERE, null, ex);
21    }
22 }

```

Tidak lupa untuk mengimport file jar yang dibutuhkan terlebih dahulu, agar jasperReport dapat berjalan dengan semestinya.

Referenced Libraries

Specify referenced libraries of the project.

Path

| | |
|--|---|
| /home/milea/Documents/kuliah/pemrogramanBerbasisObyek/postgresql-42.6.0.jar | × |
| /home/milea/Downloads/commons-beanutils-1.9.4-bin/commons-beanutils-1.9.4/commons-beanutils-1.9.4-javadoc.jar | × |
| /home/milea/Downloads/commons-beanutils-1.9.4-bin/commons-beanutils-1.9.4/commons-beanutils-1.9.4-tests.jar | × |
| /home/milea/Downloads/commons-beanutils-1.9.4-bin/commons-beanutils-1.9.4/commons-beanutils-1.9.4.jar | × |
| /home/milea/Downloads/commons-collections4-4.4-bin/commons-collections4-4.4/commons-collections4-4.4-javadoc.jar | × |
| /home/milea/Downloads/commons-collections4-4.4-bin/commons-collections4-4.4/commons-collections4-4.4-tests.jar | × |
| /home/milea/Downloads/commons-collections4-4.4-bin/commons-collections4-4.4/commons-collections4-4.4.jar | × |
| /home/milea/Downloads/commons-digester-2.1-bin/commons-digester-2.1/commons-digester-2.1-javadoc.jar | × |
| /home/milea/Downloads/commons-digester-2.1-bin/commons-digester-2.1/commons-digester-2.1.jar | × |
| /home/milea/Downloads/commons-logging-1.2-bin/commons-logging-1.2/commons-logging-1.2-javadoc.jar | × |
| /home/milea/Downloads/commons-logging-1.2-bin/commons-logging-1.2/commons-logging-1.2.jar | × |
| /home/milea/Downloads/jasperreports-6.20.6.jar | × |

Add

```

1 private void btnDeleteActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_btnDeleteActionPerformed
2     // TODO add your handling code here:
3     int id = Integer.parseInt(inputID.getText().trim());
4     if (id > 0) {
5         try {
6             // TODO add your handling code here:
7             Connection conn = DriverManager.getConnection("jdbc:postgresql://localhost:5432/perpustakaan", "postgres", "postgres");
8             int k = deleteData(conn, id);
9             if (k > 0) {
10                 tampil(conn);
11                 this.peringatan("Hapus Berhasil");
12             } else {
13                 this.peringatan("Hapus Gagal");
14             }
15         } catch (SQLException ex) {
16             Logger.getLogger(Output.class.getName()).log(Level.SEVERE, null, ex);
17         }
18     } else {
19         this.peringatan("Delete Data Buku Gagal");
20     }
21     this.resetField();
22 }

```

```

1 public static void main(String args[]) {
2     /* Set the Nimbus look and feel */
3     //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
4     /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
5      * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
6      */
7     try {
8         for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
9             if ("Nimbus".equals(info.getName())) {
10                 javax.swing.UIManager.setLookAndFeel(info.getClassName());
11                 break;
12             }
13         }
14     } catch (ClassNotFoundException ex) {
15         java.util.logging.Logger.getLogger(Output.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
16     } catch (InstantiationException ex) {
17         java.util.logging.Logger.getLogger(Output.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
18     } catch (IllegalAccessException ex) {
19         java.util.logging.Logger.getLogger(Output.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
20     } catch (javax.swing.UnsupportedLookAndFeelException ex) {
21         java.util.logging.Logger.getLogger(Output.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
22     }
23     //</editor-fold>
24
25     /* Create and display the form */
26     java.awt.EventQueue.invokeLater(new Runnable() {
27         public void run() {
28             new Output().setVisible(true);
29         }
30     });
31 }
32
33 // Variables declaration - do not modify//GEN-BEGIN:variables
34 private javax.swing.JButton btnDelete;
35 private javax.swing.JButton btnInsert;
36 private javax.swing.JButton btnRefresh;
37 private javax.swing.JButton btnUpdate;
38 private javax.swing.JTextField inputID;
39 private javax.swing.JTextField inputJenisBuku;
40 private javax.swing.JTextField inputJudul;
41 private javax.swing.JTextField inputPengarang;
42 private javax.swing.JScrollPane jScrollPane1;
43 private javax.swing.JLabel label1;
44 private javax.swing.JLabel labelID;
45 private javax.swing.JLabel labelJenisBuku;
46 private javax.swing.JLabel labelJudul;
47 private javax.swing.JLabel labelPengarang;
48 private javax.swing.JTable tabelBuku;
49 // End of variables declaration//GEN-END:variables
50 }

```

2. Output

a. Tampilan Awal

Daftar Buku

ID Buku Judul

Pengarang Jenis Buku

| ID | Judul | Pengarang | Jenis |
|----|--------------------------|--------------|------------|
| 2 | bandung lautan air | mpu gandrang | fiksi |
| 4 | Mereka tak pernah me... | Tipe X | nostalgia |
| 5 | Bila kau tak disampingku | Sheila on 7 | Lirik lagu |
| 10 | Dan | Sheila | Lagu |
| 11 | Nemen | GuyonWaton | Lagu Jawa |

b. Ketika menambahkan data

Daftar Buku

ID Buku Judul

Pengarang Jenis Buku

| ID | Judul | Pengarang | Jenis |
|----|-------|-----------|------------|
| 2 | band | | fiksi |
| 4 | Mere | | nostalgia |
| 5 | Bila | | Lirik lagu |
| 10 | Dan | | Lagu |
| 11 | Nem | | Lagu Jawa |
| 12 | Bera | | Lagu Galau |

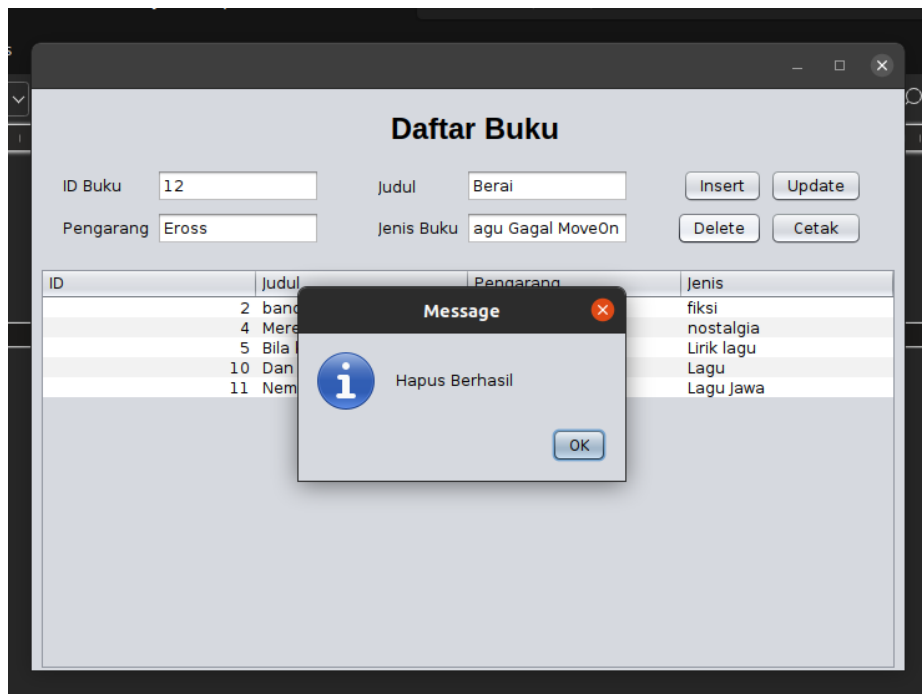
Message

Simpan Berhasil

c. Ketika mengubah data



d. Ketika menghapus data



e. Ketika mencetak data

Activities Apache NetBeans 19 Okt 1 23:49 JasperViewer

ID Buku
Pengarang

ID
2
4
5
10
11

DAFTAR BUKU "Perputakaan Publik"

| id | name | pengarang | jenis |
|----|----------------------------|-------------|-----------|
| 2 | bandung lautan air | mpu gandrng | fiksi |
| 4 | Mereka tak pernah mengerti | Tipe X | nostalgia |
| 5 | Bila kau tak disampingku | Sheila on 7 | Link lagu |
| 10 | Dan | Sheila | Lagu |
| 11 | Nemen | GuyonWaton | Lagu Jawa |

Print
Output - TugasPertemuan
Run: WARNING: An illegal reflective access operation has occurred
WARNING: Please consider reporting this to the vendor, since a future release might remove the support for the reflective access.
WARNING: Use --illegal-access=warn to enable warnings of this kind. Use --enable-preview to enable preview features.
Report Problems With This JAR

Page 1 of 1

INS

Activities Apache NetBeans 19 Okt 1 23:49 JasperViewer

ID Buku
Pengarang

ID
2
4
5
10
11

Dicetak pada Sun Oct 01 23:49:32 WIB 2023

Print
Output - TugasPertemuan
Run: WARNING: An illegal reflective access operation has occurred
WARNING: Please consider reporting this to the vendor, since a future release might remove the support for the reflective access.
WARNING: Use --illegal-access=warn to enable warnings of this kind. Use --enable-preview to enable preview features.
Report Problems With This JAR

Page 1 of 1

INS