

LAPORAN PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK
MODUL 5



Oleh :
Erick Evan Nicholas
2311103115
S1SI-07C

PROGRAM STUDI S1 SISTEM INFORMASI
FAKULTAS REKAYASA INDUSTRI
UNIVERSITAS TELKOM PURWOKERTO
2024

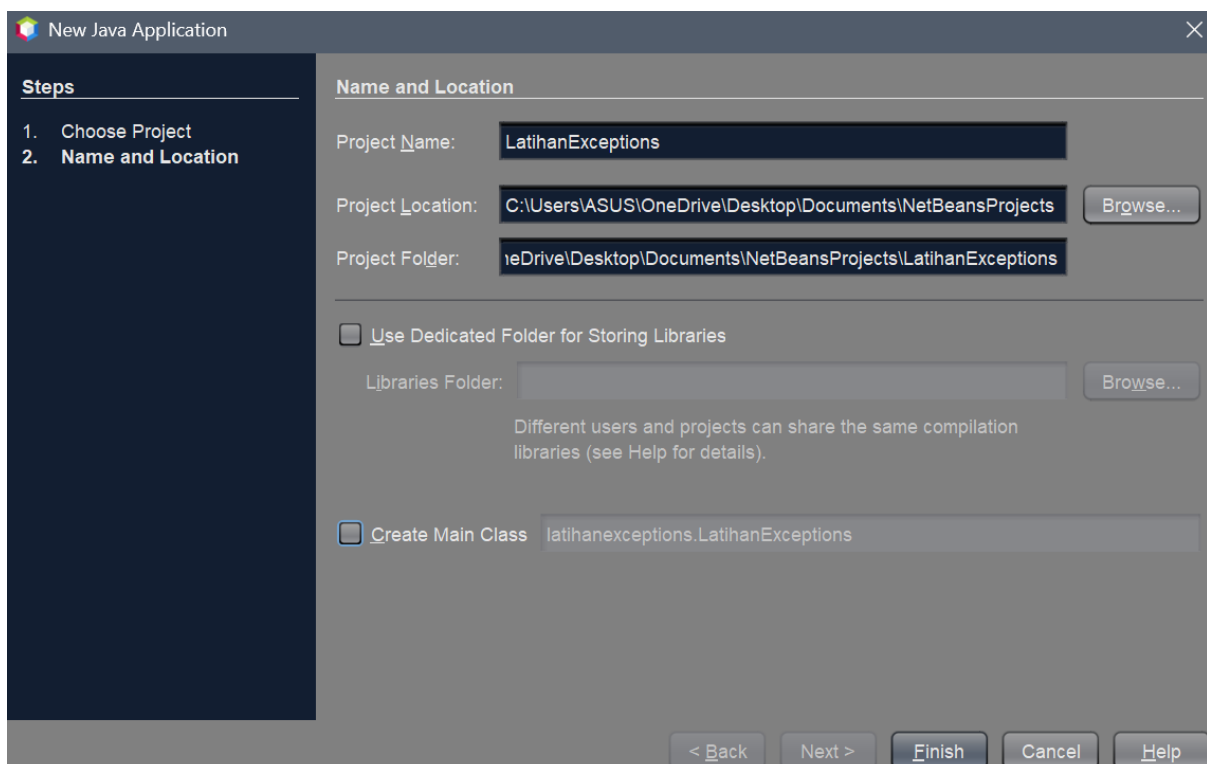
LATIHAN EXCEPTION HANDLING

A. Studi Kasus

Sistem Informasi Perpustakaan – Input Buku

B. Guided

1. Buat project baru pada Netbeans. Pada saat membuat project, hilangkan centang pada *Create Main Class*



2. Tambahkan package baru dengan nama **perpustakaan** dengan cara klik kanan pada *Source Packages* → *New* → *Java Package...*
3. Pada *package* perpustakaan, buat Class baru dengan nama **Buku**

Buku.java

```
package Perpustakaan;

/**
 * 2311103115
 * @author Erick Evan Nicholas
 */
```

```

* SISI-07C
*/
public class Buku {
    private String judul;
    private String pengarang;
    private String isbn;

    public Buku(String judul, String pengarang, String isbn){
        this.judul = judul;
        this.pengarang = pengarang;
        this.isbn = isbn;
    }
    public String getJudul() {
        return judul;
    }
    public String getPengarang() {
        return pengarang;
    }
    public String getIsbn() {
        return isbn;
    }
}

```

4. Tambahkan tampilan program untuk input data dengan cara klik kanan pada *package* perpustakaan → New → JFrame Form... , beri nama **InputBuku**
5. Buat tampilan seperti dibawah ini:

No.	Component	Variable Name
1	JPanel	
2	JScrollPane	
3	TextField	txtJudul
4	TextField	txtPengarang
5	TextField	txtIsbn
6	Button	btnSubmit
7	JTable	tblBuku
8	JLabel	lblKeterangan

6. Klik 2x pada tombol Tambah Data, tambahkan kode berikut :

```

private void btnSubmitActionPerformed(java.awt.event.ActionEvent evt) {
    String judul = txtJudul.getText();
    String pengarang = txtPengarang.getText();
    String isbn = txtIsbn.getText();
}

```

```

// Membuat objek Buku baru
Buku buku = new Buku(judul, pengarang, isbn);

// Menambah data buku ke tabel
DefaultTableModel data = (DefaultTableModel)
tblBuku.getModel();
data.addRow(new Object[] {buku.getJudul(), buku.getPengarang(),
buku.getIsbn()});

// Reset form input
txtJudul.setText("");
txtPengarang.setText("");
txtIsbn.setText("");
}

```

7. Jalankan project lalu coba tambahkan data, maka akan muncul data pada tabel. Namun ketika form kosong lalu klik tambah data, maka akan ada data kosong yang masuk pada tabel. Hal ini bisa diantisipasi dengan membuat *user defined exception*.
8. Buat sebuah class baru dengan nama **ValidasiInputException**
ValidasiInputException.java

```

public class ValidasiInputException extends
Exception{public ValidasiInputException(String
message) {
    super(message);
}
}

```

9. Ubah kode pada Class buku
Buku.java

```

package Perpustakaan;

/**
 * 2311103115
 * @author Erick Evan Nicholas
 * S1SI-07C
 */
public class Buku {
    private String judul;
    private String pengarang;
    private String isbn;

    public Buku(String judul, String pengarang, String isbn)throws ValidasiInputException{
        if (judul.isEmpty()) {
            throw new ValidasiInputException("Judul buku tidak boleh kosong!");
        }
    }
}

```

```

if (pengarang.isEmpty()) {
    throw new ValidasiInputException("Pengarang tidak boleh kosong!");
}
if (!isbn.matches("\\d{13}")) {
    throw new ValidasiInputException("ISBN harus 13 digit angka!");
}
this.judul = judul;
this.pengarang = pengarang;
this.isbn = isbn;
}
public String getJudul() {
    return judul;
}
public String getPengarang() {
    return pengarang;
}
public String getIsbn() {
    return isbn;
}
}

```

10. Klik 2x pada tombol Tambah Data, ubah kode berikut:

```

private void btnSubmitActionPerformed(java.awt.event.ActionEvent evt) {
    try{
        String judul = txtJudul.getText();
        String pengarang = txtPengarang.getText();
        String isbn = txtIsbn.getText();

        // Membuat objek Buku baru
        Buku buku = new Buku(judul, pengarang, isbn);

        // Menambah data buku ke tabel
        DefaultTableModel data = (DefaultTableModel) tblBuku.getModel();
        data.addRow(new Object[]{buku.getJudul(), buku.getPengarang(),
            buku.getIsbn()});

        // Reset form input
        txtJudul.setText("");
        txtPengarang.setText("");
    }
}

```

```

txtIsbn.setText("");

lblKeterangan.setForeground(Color.GREEN);

lblKeterangan.setText("Buku berhasil ditambahkan!");
} catch (ValidasiInputException e) {

    lblKeterangan.setForeground(Color.RED);

    lblKeterangan.setText("Error: " + e.getMessage());

}
}

```

11. Jalankan project, coba tambahkan data baru. Ketika form kosong maka akan muncul peringatan pada bagian bawah aplikasi

Judul Buku	[_____]
Nama Pengarang	[_____]
No ISBN	[_____]
[Tambah Data]	

Tabel Buku:		
+	+	+
	Judul	Pengarang
+	+	+
+	+	+

Keterangan: [Status akan muncul di sini]

Penjelasan : Aplikasi Manajemen Buku Perpustakaan ini adalah sebuah program sederhana yang dirancang untuk membantu pustakawan atau pengelola perpustakaan dalam mencatat dan mengelola data buku. Singkatnya, ini adalah aplikasi sederhana namun fungsional untuk mengelola data buku dengan fokus pada kemudahan penggunaan dan akurasi data.

C. Unguided

Tambahkan inputan berikut pada aplikasi:

1. Tahun Terbit: Validasi untuk angka empat digit.
2. Harga Buku: Validasi untuk angka positif.
3. Kategori Buku: Validasi untuk input tidak kosong. (dalam bentuk Dropdown)

Jawab :

Buku.java:

```
package Perpustakaan;

/**
 * 2311103115
 * @author Erick Evan Nicholas
 * S1SI-07C
 */
public class Buku {
    private String judul;
    private String pengarang;
    private String isbn;
    private String tahunTerbit;
    private double hargaBuku;
    private String kategori;

    public Buku(String judul, String pengarang, String isbn, String tahunTerbit, double
    hargaBuku, String kategori) throws ValidasiInputException {
        if (judul.isEmpty()) {
            throw new ValidasiInputException("Judul buku tidak boleh kosong!");
        }
        if (pengarang.isEmpty()) {
            throw new ValidasiInputException("Pengarang tidak boleh kosong!");
        }
        if (!isbn.matches("\\d{13}")) {
            throw new ValidasiInputException("ISBN harus 13 digit angka!");
        }
        if (!tahunTerbit.matches("\\d{4}")) {
            throw new ValidasiInputException("Tahun terbit harus 4 digit angka!");
        }
        if (hargaBuku <= 0) {
            throw new ValidasiInputException("Harga buku harus lebih dari 0!");
        }
        if (kategori == null || kategori.isEmpty()) {
            throw new ValidasiInputException("Kategori buku tidak boleh kosong!");
        }
        this.judul = judul;
        this.pengarang = pengarang;
        this.isbn = isbn;
        this.tahunTerbit = tahunTerbit;
        this.hargaBuku = hargaBuku;
        this.kategori = kategori;
    }
}
```

```

    public String getJudul() {
        return judul;
    }
    public String getPengarang() {
        return pengarang;
    }
    public String getIsbn() {
        return isbn;
    }
    public String getTahunTerbit() {
        return tahunTerbit;
    }

    public double getHargaBuku() {
        return hargaBuku;
    }

    public String getKategori() {
        return kategori;
    }
}

```

InputBuku.java

```

package Perpustakaan;
import javax.swing.table.DefaultTableModel;
import java.awt.Color;

/**
 * 2311103115
 * @author Erick Evan Nicholas
 * S1SI-07C
 */
public class InputBuku extends javax.swing.JFrame {

    /**
     * Creates new form InputBuku
     */
    public InputBuku() {
        initComponents();
        txtKategori.addItem("Novel");
        txtKategori.addItem("Buku Teks");
        txtKategori.addItem("Komik");
        txtKategori.addItem("Majalah");
        txtKategori.addItem("Lainnya");
    }
    private void initComponents() {
        txtTahunTerbit = new javax.swing.JTextField();
        txtHargaBuku = new javax.swing.JTextField();

```



```

        txtKategori = new javax.swing.JComboBox<>();
        jLabel5 = new javax.swing.JLabel();
        jLabel6 = new javax.swing.JLabel();
        jLabel7 = new javax.swing.JLabel();
        jLabel5.setText("Tahun Terbit");
        jLabel6.setText("Harga Buku");
        jLabel7.setText("Kategori");
        tblBuku.setModel(new javax.swing.table.DefaultTableModel(
            new Object [][] {
                { "", "", "", "", "", "" }
            },
            new String [] {
                "Judul", "Pengarang", "ISBN", "Tahun Terbit", "Harga", "Kategori"
            }
        ));
    }

    /**
     * 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();
        txtJudul = new javax.swing.JTextField();
        txtPengarang = new javax.swing.JTextField();
        txtIsbn = new javax.swing.JTextField();
        btnSubmit = new javax.swing.JButton();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jLabel4 = new javax.swing.JLabel();
        jScrollPane1 = new javax.swing.JScrollPane();
        tblBuku = new javax.swing.JTable();
        lblKeterangan = new javax.swing.JLabel();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

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

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

```

```

});

jLabel2.setText("Judul Buku");

jLabel3.setText("Nama Pengarang");

jLabel4.setText("No Isbn");

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

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup()
        .addGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    .addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    .addComponent(jLabel4, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
    .addComponent(txtJudul)
    .addComponent(txtPengarang)
    .addComponent(txtIsbn)
    .addComponent(btnSubmit, javax.swing.GroupLayout.DEFAULT_SIZE, 185,
Short.MAX_VALUE))
    .addGap())
);
jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel1Layout.createSequentialGroup()
        .addGap()
        .addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(txtJudul, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jLabel2))

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

```

```

.addGroup(jPanel1Layout.createParallelGroup(GroupLayout.Alignment.BASELINE)
    .addComponent(txtPengarang,
        javax.swing.GroupLayout.PREFERRED_SIZE,
        javax.swing.GroupLayout.DEFAULT_SIZE,
        javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jLabel3))

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

.addGroup(jPanel1Layout.createParallelGroup(GroupLayout.Alignment.BASELINE)
    .addComponent(txtIsbn, javax.swing.GroupLayout.PREFERRED_SIZE,
        javax.swing.GroupLayout.DEFAULT_SIZE,
        javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jLabel4))
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
    javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    .addComponent(btnSubmit))
);

tblBuku.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {"", null, null},
        {null, null, null},
        {null, null, null},
        {null, null, null}
    },
    new String [] {
        "Judul", "Pengarang", "ISBN"
    }
));
jScrollPane1.setViewportViewView(tblBuku);

lblKeterangan.setText("Keterangan");

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .add(layout.createParallelGroup(GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
                javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT_SIZE,
                388, Short.MAX_VALUE)
            .addComponent(lblKeterangan, javax.swing.GroupLayout.DEFAULT_SIZE,
                388, Short.MAX_VALUE)
        )
    )
);

```

```

        .addComponent(lblKeterangan)
        .addGap(0, 0, Short.MAX_VALUE)))
        .addContainerGap()
    );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
                javax.swing.GroupLayout.DEFAULT_SIZE,
                javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(21, 21, 21)
            .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE,
                100, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
                javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(lblKeterangan)
            .addContainerGap()
        );

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

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

private void btnSubmitActionPerformed(java.awt.event.ActionEvent evt) {
    double hargaBuku;
    try{
        hargaBuku = Double.parseDouble(txtHargaBuku.getText());
    } catch (NumberFormatException e) {
        throw new ValidasiInputException("Harga buku harus berupa angka!");
    }

    String kategori = (String) txtKategori.getSelectedItem();

    // Membuat objek Buku baru
    Buku buku = new Buku(judul, pengarang, isbn, tahunTerbit, hargaBuku, kategori);

    // Menambah data buku ke tabel
    DefaultTableModel data = (DefaultTableModel) tblBuku.getModel();
    data.addRow(new Object[] {
        buku.getJudul(),
        buku.getPengarang(),
        buku.getIsbn(),
        buku.getTahunTerbit(),
        buku.getHargaBuku(),
        buku.getKategori()
    });
}

```

```

        txtJudul.setText("");
        txtPengarang.setText("");
        txtIsbn.setText("");
        txtTahunTerbit.setText("");
        txtHargaBuku.setText("");
        txtKategori.setSelectedIndex(0);

        lblKeterangan.setForeground(Color.GREEN);
        lblKeterangan.setText("Buku berhasil ditambahkan!");
    } catch (ValidasiInputException e) {
        lblKeterangan.setForeground(Color.RED);
        lblKeterangan.setText("Error: " + e.getMessage());
    }
}

```

ValidasiInputException.java

```

package Perpustakaan;

/**
 * 2311103115
 * @author Erick Evan Nicholas
 * S1SI-07C
 */
public class ValidasiInputException extends
    Exception{public ValidasiInputException(String message) {
        super(message);
    }
}

```

Running :

```

+-----+
| Judul Buku      [_____] |
| Nama Pengarang  [_____] |
| No ISBN         [_____] |
| Tahun Terbit    [_____] |
| Harga Buku      [_____] |
| Kategori        [Dropdown ▼] |
|               |
|               [Tambah Data] |
+-----+
| Tabel Buku: |
| +-----+ |
| | Judul | Pengarang | ISBN | Thn | |
| |       |          |      |    | |
| |       |          |      |    | |
| |       |          |      |    | |
| |       |          |      |    | |
| +-----+ |
| |       |          |      |    | |
| |       |          |      |    | |
| |       |          |      |    | |
| +-----+ |
+-----+
| Keterangan: [Status akan muncul di sini] |
+-----+

```

Tampilan awal akan seperti di atas.

Setelah menginputkan Buku :

```

+-----+
| Tabel Buku: |
| +-----+ |
| | Laskar | Andrea | 1234567890 | 2005 | |
| | Pelangi | Hirata | 123 | 85000 | |
| |       |          |      |      | |
| |       |          |      |      | |
| +-----+ |
+-----+
| Keterangan: Buku berhasil ditambahkan! |
+-----+

```

Apabila melakukan kesalahan saat input :

```

+-----+
| Keterangan: Error: Tahun terbit harus |
|               4 digit angka! |
+-----+

```

Penjelasan : Aplikasi ini merupakan alat bantu digital yang efisien untuk manajemen katalog perpustakaan, menggabungkan kemudahan penggunaan dengan kontrol kualitas data yang tinggi.