

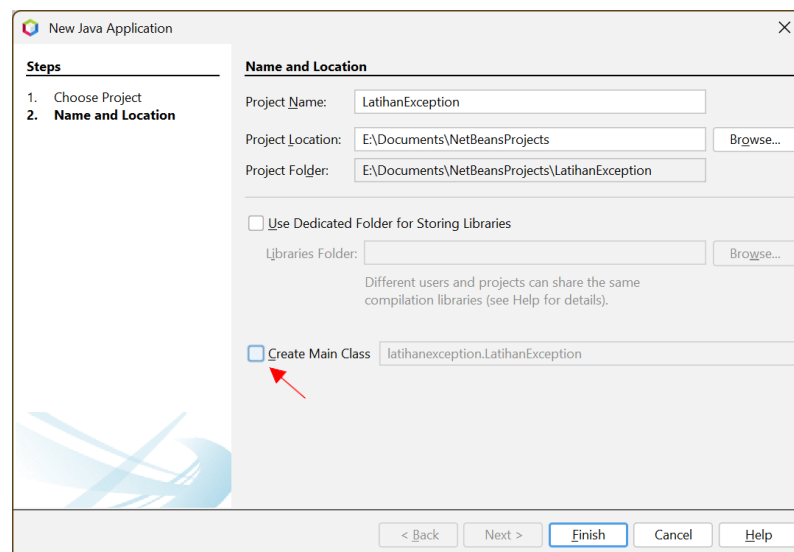
LATIHAN EXCEPTION HANDLING

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

```
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;
    }
}
```

```

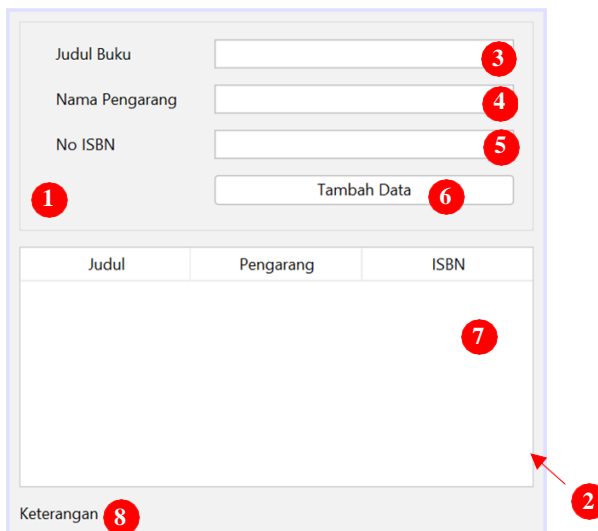
public String getJudul() {
    return judul;
}

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 di bawah ini:



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

```
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

JAWABAN GUIDED :

1. Buku.java

```

package perpustakaan;

/**
 *
 * @author GeordySiphoSamuelDamanik
 * 2311103112
 * S1SI-07-C
 */
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 Kosong Tulung isi dulu");
        }
        if(pengarang.isEmpty()){
            throw new ValidasiInputException ("Pengarang Kosong woyy");
        }
        if(!isbn.matches("\\d{13}")){
            throw new ValidasiInputException ("ISBN ISI 13 Brayy");
        }
        this.judul = judul;
        this.pengarang = pengarang;
        this.isbn = isbn;
    }

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

```

= Berikut adalah kode kelas buku yang mana dalam kelas ini merepresentasikan sebuah buku dengan tiga atribut privat yaitu judul, pengarang, dan isbn. Lalu pada konstruktor kelas ini memiliki validasi input yang

ketat untuk memastikan data buku yang dimasukkan memenuhi kriteria tertentu. Jadi jika judul atau pengarang dikosongkan, atau jika ISBN tidak tepat atau tidak terdiri dari 13 digit angka, maka konstruktor akan melemparkan sebuah `ValidasiInputException` dengan pesan kesalahan yang spesifik. Setelah melewati validasi, konstruktor akan menetapkan nilai-nilai tersebut ke variabel instance. Kemudian pada kelas ini juga menyediakan metode getter sederhana yaitu (`getJudul()`, `getPengarang()`, dan `getIsbn()`) untuk mengakses nilai-nilai atribut privat, yang memungkinkan pengambilan informasi buku tanpa mengubah nilainya secara langsung.

2. InputData.java

```
package perpustakaan;

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

/**
 *
 * @author GeordySiphoSamuelDamanik
 * 2311103112
 * S1SI-07-C
 */
public class InputData extends javax.swing.JFrame {

    /**
     * Creates new form InputData
     */
    public InputData() {
        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();
        txtJudul = new javax.swing.JTextField();
        txtPengarang = new javax.swing.JTextField();
        txtIsbn = new javax.swing.JTextField();
        btnSubmit = new javax.swing.JButton();
        jScrollPane1 = new javax.swing.JScrollPane();
        tableBuku = new javax.swing.JTable();
        lblMessage = new javax.swing.JLabel();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jLabel1.setText("Judu Buku");

        jLabel2.setText("No.ISBN");
```

```
jLabel3.setText("Pengarang");

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

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(18, 18, 18)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel1)
                .addComponent(jLabel3)
                .addComponent(jLabel2))
            .addGap(61, 61, 61)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                .addComponent(btnSubmit, javax.swing.GroupLayout.DEFAULT_SIZE, 223, Short.MAX_VALUE)
                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
                    .addComponent(txtIsbn, javax.swing.GroupLayout.DEFAULT_SIZE, 223, Short.MAX_VALUE)
                    .addComponent(txtPengarang)
                    .addComponent(txtJudul)))
            .addContainerGap(28, Short.MAX_VALUE))
        );
jPanel1Layout.setVerticalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addContainerGap()
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel1)
                .addComponent(txtJudul, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 16, Short.MAX_VALUE)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel3)
                .addComponent(txtPengarang, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(18, 18, 18)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel2)
                .addComponent(txtIsbn, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        );
```

[illegible]


```

        .addGap(53, 53, 53))
    );

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

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

        Buku buku = new Buku (judul, pengarang, isbn);

        DefaultTableModel data =(DefaultTableModel)tableBuku.getModel();
        data.addRow(new Object[] {buku.getJudul(), buku.getPengarang(), buku.getIsbn()});

        txtJudul.setText("");
        txtPengarang.setText("");
        txtIsbn.setText("");
        lblMessage.setText("Data Berhasil Ditambahkan");
        lblMessage.setForeground(Color.green);
    } catch (ValidasiInputException e) {
        lblMessage.setText("Error : "+e.getMessage());
        lblMessage.setForeground(Color.red);
    }
}

/**
 * @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.html
     */
    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(InputData.class.getName()).log(java.util.logging.Level.SEVERE,
        null, ex);
    } catch (InstantiationException ex) {

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

```

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

java.util.logging.Logger.getLogger(InputData.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 InputData().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton btnSubmit;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JPanel jPanel1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JLabel lblMessage;
private javax.swing.JTable tableBuku;
private javax.swing.JTextField txtIsbn;
private javax.swing.JTextField txtJudul;
private javax.swing.JTextField txtPengarang;
// End of variables declaration
}
```

= Berikut adalah kode untuk menginput data buku yang dibuat menggunakan Swing. Pada kelas InputData disini adalah sebuah form yang memungkinkan kita sebagai pengguna untuk memasukkan informasi buku seperti judul, pengarang, dan ISBN melalui tiga text field yang disediakan. Ketika tombol "Tambah Data" diklik, aplikasi akan melakukan validasi input menggunakan kelas Buku yang telah dibuat sebelumnya. Jika input valid, data buku akan ditambahkan ke sebuah tabel (tableBuku) yang menampilkan daftar buku yang telah diinput. Proses validasi mencakup pengecekan apakah judul dan pengarang tidak kosong, serta ISBN harus terdiri dari 13 digit angka. Jika terjadi kesalahan validasi, sebuah pesan error akan ditampilkan dalam label lblMessage dengan warna merah. Setelah kita berhasil menambahkan data, input field akan dikosongkan kembali dan label akan menampilkan pesan sukses dengan warna hijau.

```
package perpustakaan;

/**
 *
 * @author GeordySiphoSamuelDamanik
 * 2311103112
 * S1SI-07-C
 */
public class ValidasiInputException extends Exception {
    public ValidasiInputException(String message){
        super (message);
    }
}
```

= Kode ini mendefinisikan kelas `ValidasiInputException`, kelas ini ialah turunan dari kelas `Exception` yang dirancang untuk menangani kesalahan validasi input secara spesifik dalam konteks aplikasi perpustakaan. Konstruktor dari kelas ini menerima sebuah pesan kesalahan sebagai parameter dan menggunakan metode `super()` untuk meneruskan pesan tersebut ke konstruktor kelas induk `Exception`. Jadi kelas `ValidasiInputException` memberikan kemampuan kepada pengembang untuk melempar pengecualian dengan pesan yang jelas dan informatif saat terjadi kesalahan validasi, seperti input yang kosong atau tidak sesuai dengan format, dalam aplikasi manajemen data buku. Dalam kode sebelumnya, pengecualian ini diterapkan untuk memvalidasi input judul, pengarang, dan ISBN pada kelas `Buku`, sehingga memberikan mekanisme kontrol kesalahan yang lebih terstruktur dan mudah untuk dipahami.

HASIL RUNNING :



Judu Buku: Kulawa Majakuza

Pengarang: Geordy

No.ISBN: 12345

Tambah Data

Judul	Pengarang	ISBN
-------	-----------	------

Error : ISBN ISI 13 Brayy

= Berikut adalah hasil running yang mana akan tampil sebuah design tampilan Frame seperti atas. Ada judul buku yang saya isikan Kulawa Majakuza, untuk pengarang Geordy, dan No.ISBN diisikan 12345, namun saat ingin menekan button Tambah Data maka akan gagal dikarenakan ISBN tidak diisikan sebanyak 13 angka, dan tampilan gagalnya akan tampil dipojok kiri bawah dengan warna merah yang menandakan bahwa gagal dan keterangannya yaitu “Error : ISBN ISI 13 Brayy”.



Judu Buku:

Pengarang:

No.ISBN:

Tambah Data

Judul	Pengarang	ISBN
Kulawa Majakuza	Geordy	1234567890123

Data Berhasil Ditambahkan

= Namun ketika kita isikan No.ISBN sebanyak 13 angka maka data akan berhasil ditambahkan seperti pada tampilan diatas, dan untuk keterangan berhasilnya disertai dengan warna Hijau yang berarti Berhasil dan keterangannya adalah “Data Berhasil Ditambahkan”.

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)

JAWABAN UNGUIDED :

1. Tambahan Inputan Tahun Terbit, Harga Buku, dan Kategori Buku pada kelas Buku.

```
package perpustakaan;

/**
 *
 * @author GeordySiphoSamuelDamanik
 * 2311103112
 * S1SI-07-C
 */
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 Kosong Tulung isi dulu");
        }
        if(pengarang.isEmpty()){
            throw new ValidasiInputException ("Pengarang Kosong woyy");
        }
        if(!isbn.matches("\\d{13}")){
            throw new ValidasiInputException ("ISBN ISI 13 Brayy");
        }

        if(!tahunTerbit.matches("\\d{4}")){
            throw new ValidasiInputException("Tahun Terbit harus 4 digit angka");
        }

        if(hargaBuku <= 0){
            throw new ValidasiInputException("Harga Buku harus positif");
        }

        if(kategori == null || kategori.isEmpty()){
            throw new ValidasiInputException("Kategori Buku harus dipilih");
        }

        this.judul = judul;
        this.pengarang = pengarang;
        this.isbn = isbn;
```

```
this.tahunTerbit = tahunTerbit;
    this.hargaBuku = hargaBuku;
    this.kategori = kategori;
}

public String getTahunTerbit() {
    return tahunTerbit;
}

public double getHargaBuku() {
    return hargaBuku;
}

public String getKategori() {
    return kategori;
}

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

= Berikut adalah kode kelas buku yang mana didalamnya terdapat penambahan 3 atribut baru yaitu tahunTerbit, hargaBuku, dan kategori. Atribut tahunTerbit bertipe String yang mengharuskan pengguna untuk memasukkan tahun terbit dengan format empat digit angka (\\d{4}). Jika input tidak sesuai dengan format ini, maka akan dilemparkan `ValidasiInputException`. Atribut hargaBuku ditambahkan dengan tipe data double, yang memerlukan validasi agar harga buku harus lebih besar dari nol. Jika nilai harga buku tidak valid (negatif atau nol), maka exception akan dilemparkan. Atribut kategori ditambahkan sebagai String yang harus diisi. Jika kategori tidak dipilih (kosong atau null), maka akan ada pengecekan dan exception yang dilemparkan. Juga kita tambahkan Validasi untuk tahunTerbit memastikan input adalah empat digit angka, hargaBuku harus lebih besar dari nol, dan kategori harus terisi. Dan juga terdapat penambahan konstruktor tahunTerbit, hargaBuku, dan kategori untuk pengecekan tambahan sesuai dengan validasi yang telah disebutkan.

2. Tambahkan Inputan pada Tahun Terbit, Harga Buku, dan Kategori Buku pada kelas InputData.

```
package perpustakaan;

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

/**
 *
 * @author GeordySiphoSamuelDamanik
 * 2311103112
 * SISI-07-C
 */
public class InputData extends javax.swing.JFrame {

    /**
     * Creates new form InputData
     */
    public InputData() {
        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();
        txtJudul = new javax.swing.JTextField();
        txtPengarang = new javax.swing.JTextField();
        txtIsbn = new javax.swing.JTextField();
        jLabel4 = new javax.swing.JLabel();
        txtTahunTerbit = new javax.swing.JTextField();
        jLabel5 = new javax.swing.JLabel();
        txtHargabuku = new javax.swing.JTextField();
        jLabel6 = new javax.swing.JLabel();
        btnSubmit = new javax.swing.JButton();
        txtKategori = new javax.swing.JComboBox<>();
        jScrollPane1 = new javax.swing.JScrollPane();
        tableBuku = new javax.swing.JTable();
        lblMessage = new javax.swing.JLabel();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jLabel1.setText("Judu  Buku");

        jLabel2.setText("No.ISBN");

        jLabel3.setText("Pengarang");
```

```

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

jLabel4.setText("Tahun Terbit");

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

jLabel5.setText("Harga Buku");

jLabel6.setText("Kategori");

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

txtKategori.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "Isi Kategori...",
"Fiksi", "Novel", "komik", "Cerpén", "Pendidikan", "Relogi", "Dongeng", "Cinta" }));

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(18, 18, 18)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel1)
                .addComponent(jLabel3)
                .addComponent(jLabel2)
                .addComponent(jLabel4, javax.swing.GroupLayout.PREFERRED_SIZE, 83,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel5, javax.swing.GroupLayout.PREFERRED_SIZE, 74,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel6, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(35, 35, 35)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING,
false)
                .addComponent(txtIsbn, javax.swing.GroupLayout.Alignment.TRAILING)
                .addComponent(txtPengarang, javax.swing.GroupLayout.Alignment.TRAILING)
                .addComponent(txtJudul, javax.swing.GroupLayout.Alignment.TRAILING)
                .addComponent(txtTahunterbit)
                .addComponent(txtHargabuku)
                .addComponent(txtKategori, 0, 223, Short.MAX_VALUE))
            .addGap(32, 32, 32)
            .addComponent(btnSubmit, javax.swing.GroupLayout.PREFERRED_SIZE, 152,
javax.swing.GroupLayout.PREFERRED_SIZE)

```



```

.addContainerGap(51, Short.MAX_VALUE))
);
jPanel1Layout.setVerticalGroup(
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(jPanel1Layout.createSequentialGroup()
.addGap(15, 15, 15)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
.addComponent(jLabel1)
.addComponent(txtJudul, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addGap(18, 18, 18)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
.addComponent(jLabel3)
.addComponent(txtPengarang, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addGap(18, 18, 18)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
.addComponent(jLabel2)
.addComponent(txtIsbn, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addGap(15, 15, 15)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
.addComponent(jLabel4)
.addComponent(txtTahunTerbit, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addGap(18, 18, 18)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
.addComponent(jLabel5)
.addComponent(txtHargabuku, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addGap(18, 18, 18)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
.addComponent(jLabel6)
.addComponent(btnSubmit)
.addComponent(txtKategori, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addContainerGap(12, Short.MAX_VALUE))
);

tableBuku.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {

    },
    new String [] {
        "Judul", "Pengarang", "ISBN", "Tahun Terbit", "Harga Buku", "Kategori"
    }
) {
    Class[] types = new Class [] {
        java.lang.String.class, java.lang.String.class, java.lang.String.class, java.lang.String.class,
        java.lang.String.class, java.lang.String.class
    };
};

```

```

public Class getColumnClass(int columnIndex) {
    return types [columnIndex];
}
});
jScrollPane1.setViewportView(tableBuku);

lblMessage.setText("Keterangan");

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(31, 31, 31)
                .addComponent(lblMessage))
            .addGroup(layout.createSequentialGroup()
                .addGap(473, Short.MAX_VALUE)
                .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 556,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGroup(layout.createSequentialGroup()
                .addGap(14, 14, 14)
                .addComponent(lblMessage)))
        .addContainerGap())
    );
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 200,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 7,
Short.MAX_VALUE)
        .addComponent(lblMessage)
        .addContainerGap(14, true)))
    );

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

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

        double hargaBuku;
        try {
            hargaBuku = Double.parseDouble(txtHargabuku.getText());
        } catch (NumberFormatException e) {
            throw new ValidasiInputException("Harga Buku harus berupa angka");
        }
    }
}

```

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

    Buku buku = new Buku (judul, pengarang, isbn, tahunTerbit, hargaBuku, kategori);

    DefaultTableModel data =(DefaultTableModel)tableBuku.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);

    lblMessage.setText("Data Berhasil Ditambahkan");
    lblMessage.setForeground(Color.green);
} catch (ValidasiInputException e){
    lblMessage.setText("Error : "+e.getMessage());
    lblMessage.setForeground(Color.red);
}
}

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

private void txtTahunTerbitActionPerformed(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.html
    */
    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(InputData.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
    } catch (InstantiationException ex) {
        java.util.logging.Logger.getLogger(InputData.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
    }
}
```

```

    } catch (IllegalAccessException ex) {

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

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

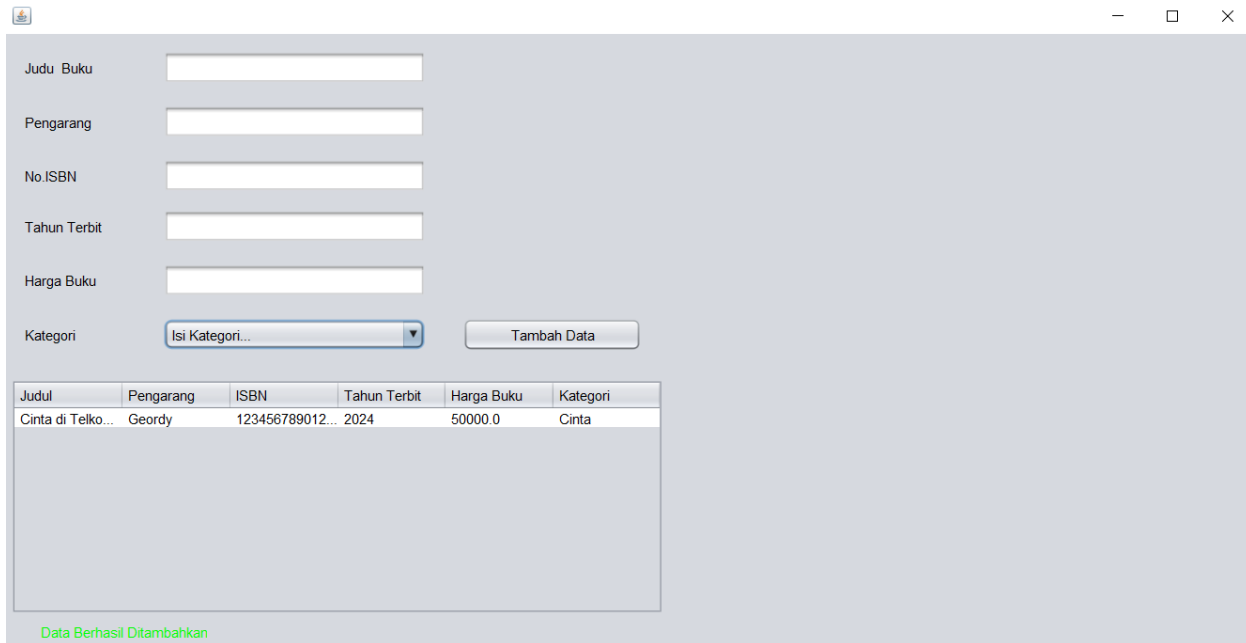
    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new InputData().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton btnSubmit;
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.JLabel jLabel6;
private javax.swing.JPanel jPanel1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JLabel lblMessage;
private javax.swing.JTable tableBuku;
private javax.swing.JTextField txtHargabuku;
private javax.swing.JTextField txtIsbn;
private javax.swing.JTextField txtJudul;
private javax.swing.JComboBox<String> txtKategori;
private javax.swing.JTextField txtPengarang;
private javax.swing.JTextField txtTahunTerbit;
// End of variables declaration
private javax.swing.JTextField txtTahunTerbit;
private javax.swing.JTextField txtHargaBuku;
private javax.swing.JComboBox<String> cbKategori;
}

```

= Kode yang telah dimodifikasi memberikan penambahan fitur yang signifikan pada aplikasi pencatatan buku. Selain data dasar seperti judul, pengarang, dan ISBN, kode baru ini memungkinkan pengguna untuk memasukkan tahun terbit, harga buku, dan kategori buku. Penambahan kolom-kolom baru pada tabel dan pembaruan pada metode `btnSubmitActionPerformed` mencerminkan perubahan ini. Perubahan yang paling menonjol adalah validasi input untuk harga buku. Kode sekarang memastikan bahwa input harga buku hanya berupa angka dengan menambahkan blok try-catch untuk menangani pengecualian `NumberFormatException` jika pengguna memasukkan karakter non-numerik. Secara keseluruhan, modifikasi ini membuat aplikasi menjadi lebih komprehensif dan informatif, memungkinkan pengguna untuk mengelola data buku dengan lebih detail.

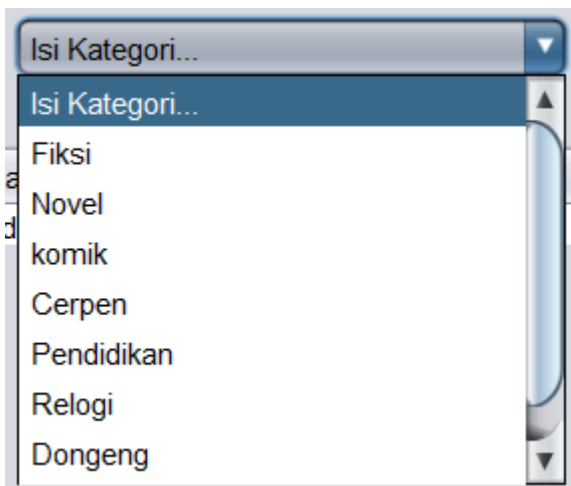
HASIL RUNNING :



The screenshot shows a web application interface for managing books. It features several input fields for book details: Judul Buku, Pengarang, No.ISBN, Tahun Terbit, and Harga Buku. There is also a dropdown menu for Kategori with a custom option 'Isi Kategori...'. A 'Tambah Data' button is located next to the category dropdown. Below the input fields is a table displaying the added data.

Judul	Pengarang	ISBN	Tahun Terbit	Harga Buku	Kategori
Cinta di Telko...	Geordy	123456789012...	2024	50000.0	Cinta

At the bottom left of the interface, a green status message reads: "Data Berhasil Ditambahkan".



= Berikut adalah tampilan design yang sudah dimodifikasi ada tambahan fitur Tahun Terbit, Harga Buku, dan Kategori. Pada Kategori itu sendiri kita bisa menambahkan berbagai kategori buku menggunakan Combo Box. Pada kategori saya setel di properties dan saya isikan kategorinya ada Fiksi, Novel, komik, Cerpen, Pendidikan, Relogi, dan Dongeng. Lalu saya mencoba untuk menambahkan data dengan Judul Cinta di Telkom Purwokerto, Pengarang Geordy, ISBN 1234567890123, Tahun terbit 2024, Harga Buku 50000, Kategori buku yaitu Cinta, dan ketika data berhasil ditambahkan akan muncul tampilan berwarna Hijau dipojok kiri bawah yaitu keterangan "Data Berhasil Ditambahkan".