

# **Ujian Akhir Semester**

## **Penjelasan Alur Program Persewaan Buku XYZ**



**Thowus Ajmalifanka**  
**A11.2019.11677**

**PROGRAM STUDI TEKNIK INFORMATIKA**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS DIAN NUSWANTORO**  
**SEMARANG**  
**2021**

Source Code Github:

Program:

[UAS\\_PBO/perpus at main · ifanthowus/UAS\\_PBO \(github.com\)](https://github.com/ifanthowus/UAS_PBO)

SQL:

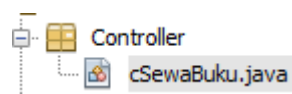
[UAS\\_PBO/SQL at main · ifanthowus/UAS\\_PBO \(github.com\)](https://github.com/ifanthowus/UAS_PBO)

Program ini dibuat menggunakan:

- IDE : Netbeans IDE 12.4
- Versi JDK : JDK vers 15.0.2
- Database : MYSQL Connector

Source Packages:

### 1. Controller



Di dalam package Controller terdapat file cSewaBuku.java

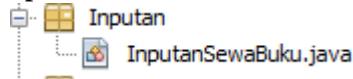
```
8  import Inputan.InputanSewaBuku; /*Memanggil file InputanSewaBuku.java dalam Package Inputan*/
9  import Model.TabelSewaBuku; /*Memanggil file TabelSewaBuku.java dalam Package Model*/
10 import Model.mTable; /* Memanggil file mTable.java dalam Package Model*/
11 import gui.maingui; /*Memanggil maingui.java dalam Package */
12 import java.util.List; /*Untuk memanggil package list pada java*/
13 import Interfaces.InterfacesSewaBuku; /*Memanggil file InterfacesSewaBuku.java dalam Package Interfaces*/
```

Pada line 8 sampai 13 dalam file cSewaBuku.java digunakan untuk pemanggilan file dalam packages yang akan dipanggil.

```
19 public class cSewaBuku {
20
21     public cSewaBuku(maingui _tampil){
22         this._tampilan = _tampil;
23         InterfacesSewaBuku = new InputanSewaBuku();
24     }
25
26     public void IsiTable(){
27         list_sewaBuku = InterfacesSewaBuku.GetAll();
28         TabelSewaBuku tabelSewaBuku = new TabelSewaBuku(list_sewaBuku);
29         _tampilan.GetTabelData().setModel(tabelSewaBuku);
30     }
31
32     maingui _tampilan;
33     InterfacesSewaBuku InterfacesSewaBuku;
34     List<mTable> list_sewaBuku;
35
36 }
37
```

Pada line 19 sampai 36 pada file cSewaBuku.java digunakan untuk menampilkan isi table dari database.

## 2. Inputan



Di dalam package Inputan terdapat file InputanSewaBuku.java

```
8 import Koneksi.Koneksi; /*Untuk memanggil Koneksi.java dalam Package Koneksi*/
9 import Model.mTable; /*Untk memanggil mTable.java dalam Package Model*/
10 import java.sql.Connection; /*Untuk memanggil package sql connection pada java*/
11 import java.sql.PreparedStatement; /*Untuk memanggil package sql PreparedStatement pada java*/
12 import java.sql.ResultSet; /*Untuk memanggil package sql ResultSet pada java*/
13 import java.sql.SQLException; /*Untuk memanggil package sql SQLException pada java */
14 import java.sql.Statement; /*Untuk memanggil package sql Statement pada java */
15 import java.text.ParseException; /*Untuk memanggil package text ParseException pada java */
16 import java.text.SimpleDateFormat; /*Untuk memanggil package text SimpleDateFormat pada java */
17 import java.time.Duration; /*Untuk memanggil package time Duration pada java */
18 import java.time.LocalDate; /*Untuk memanggil package time LocalDate pada java */
19 import java.time.format.DateTimeFormatter; /*Untuk memanggil package time format DateTimeFormatter pada java */
20 import java.util.ArrayList; /*Untuk memanggil package ArrayList pada java */
21 import java.util.Calendar; /*Untuk memanggil package Calendar pada java */
22 import java.util.Date; /*Untuk memanggil package Date pada java */
23 import java.util.List; /*Untuk memanggil package List pada java */
24 import java.util.logging.Level; /*Untuk memanggil package logging level pada java */
25 import java.util.logging.Logger; /*Untuk memanggil package logging loggerr pada java */
26 import Interfaces.InterfacesSewaBuku; /*Untuk memanggil InterfaceSewaBuku.java dalam package Interfaces*/
```

Pada line 8 sampai 26 dalam file InputanSewaBuku.java digunakan untuk pemanggilan file dalam packages yang akan dipanggil.

```
31 public class InputanSewaBuku implements InterfacesSewaBuku{
32     Connection con = Koneksi.GetConnection();
```

Pada line 31 sampai 32 dalam file InputanSewaBuku.java digunakan untuk mengkonesikan projek ke database.

```
36 public List<mTable> GetAll() {
37     SimpleDateFormat format_waktu = new SimpleDateFormat("dd MMMM yyyy");
    List<mTable> list_sewaBuku = null ;
```

Pada line 36 sampai 37 digunakan untuk pembuatan format waktu yang sesuai dengan tanggal pada hari penginputan.

```

38 |         try{
39 |             list_sewaBuku = new ArrayList<mTable>();
40 |             Statement st = con.createStatement();
41 |             String query = "SELECT * FROM sewabuku";
42 |             ResultSet rs = st.executeQuery(query);

```

Pada line 39 sampai 42 digunakan untuk pengambilan list sewa buku pada database.

```

44 |             mTable newObj = new mTable();
45 |             newObj.id = rs.getInt("id");
46 |             newObj.judul = rs.getString("judul");
47 |

```

Pada line 44 sampai 46 digunakan untuk menginputkan judul buku yang bertipe string.

```

48 |             newObj.tanggal_pinjam = rs.getString("tanggal_pinjam");
49 |             String _tgl_pinjam = newObj.tanggal_pinjam;
50 |             Date date_pinjam = new SimpleDateFormat("yyyy-MM-dddd").parse(_tgl_pinjam);
51 |             newObj.tanggal_pinjam = format_waktu.format(date_pinjam);

```

Pada line 48 sampai 51 digunakan untuk mengidentifikasi tanggal pinjam yang sesuai dengan tanggal penginputan.

```

53 |             newObj.tanggal_harus_kembali = rs.getString("tanggal_harus_kembali");
54 |             String _tgl_harusKembali = newObj.tanggal_harus_kembali;
55 |             Date date_harus_kembali = new SimpleDateFormat("yyyy-MM-dddd").parse(_tgl_harusKembali);
56 |             newObj.tanggal_harus_kembali = format_waktu.format(date_harus_kembali);
57 |
58 |             newObj.tanggal_kembali = rs.getString("tanggal_kembali");
59 |             if(newObj.tanggal_kembali != null){
60 |                 String _tgl_kembali = newObj.tanggal_kembali;
61 |                 Date date_kembali = new SimpleDateFormat("yyyy-MM-dddd").parse(_tgl_kembali);
62 |                 newObj.tanggal_kembali = format_waktu.format(date_kembali);
63 |             }else{
64 |                 newObj.tanggal_kembali = "";
65 |             }

```

Pada line 53 sampai 56 digunakan untuk mengidentifikasi tanggal harus kembali sesuai dengan aturan yaitu 7 hari setelah peminjaman.

```

68         newObj.denda = rs.getInt("denda");
69         newObj.biaya_sewa = rs.getInt("biaya_sewa");
70         list_sewaBuku.add(newObj);

```

Pada line 68 sampai 70 digunakan untuk penghitungan denda jika terjadi keterlambatan pengembalian dan perhitungan hasil biaya sewa.

```

73         catch(SQLException e){
74             System.out.println("Error");
75         } catch (ParseException ex) {
76             Logger.getLogger(DAOSewaBuku.class.getName()).log(Level.SEVERE, null, ex);
77         }
78         return list_sewaBuku;
79     }

```

Pada line 73 sampai 79 digunakan untuk mengidentifikasi semua hasil inputan kedalam sql apakah terjadi kesalahan penginputan atau tidak, jika terjadi kesalahan penginputan maka terjadi error.

```

82 String query_InsertData = "INSERT INTO sewabuku (judul,tanggal_pinjam,tanggal_harus_kembali, biaya_sewa) VALUES (
83 public void InserNewData(String _judul,String _tanggal_pinjam, String _tanggal_harus_kembali){
84     PreparedStatement statement = null;
85     int biaya_sewa = 5000;
86     try{
87         statement = con.prepareStatement(query_InsertData);
88         statement.setString(1, _judul);
89         statement.setString(2, _tanggal_pinjam);
90         statement.setString(3, _tanggal_harus_kembali);
91         statement.setInt(4, biaya_sewa);
92         statement.execute();
93     }
94     catch(SQLException e){
95         System.out.println("Gagal");
96     }
97     finally
98     {
99         try
100         {
101             statement.close();
102         } catch (SQLException ex)
103         {
104             System.out.println("Gagal");
105         }
106     }
107 }

```

Pada line 82 sampai 107 adalah statement yang sudah dibuat pada saat penginputan judul buku yang dipinjam, tanggal peminjaman, tanggal pengembalian, berserta biaya sewa ke dalam database.

```

109 String query_updateData = "UPDATE sewabuku SET judul = ? WHERE id = ?";
110 public void UpdateData(int _id,String _judul){
111     PreparedStatement statement = null;
112
113     try{
114         statement = con.prepareStatement(query_updateData);
115         statement.setString(1,_judul);
116         statement.setInt(2, _id);
117         statement.execute();
118     }
119     catch(SQLException e){
120         System.out.println("Gagal");
121     }
122     finally
123     {
124         try
125         {
126             statement.close();
127         } catch (SQLException ex)
128         {
129             System.out.println("Gagal");
130         }
131     }
132 }
133

```

Pada line 109 samapi 133 adalah statement yang sudah dibuat pada saat penginputan judul buku yang digunakan untuk update data pada database.

```

134 String query_deleteData = "DELETE FROM sewabuku where id = ?";
135 public void DeleteData(int id){
136     PreparedStatement statement = null;
137
138     try{
139         statement = con.prepareStatement(query_deleteData);
140         statement.setInt(1, id);
141         statement.execute();
142     }
143     catch(SQLException e){
144         System.out.println("Gagal");
145     }
146     finally
147     {
148         try
149         {
150             statement.close();
151         } catch (SQLException ex)
152         {
153             System.out.println("Gagal");
154         }
155     }
156 }

```

Pada line 134 sampai 156 adalah statement yang sudah dibuat untuk dapat menghapus data yang sudah dibuat.

```

159 public String GetTanggalHarusKembali(int id){
160     String tanggal_harusKembali = "";
161     try{
162         Statement st = con.createStatement();
163         String query = "SELECT tanggal_harus_kembali FROM sewabuku WHERE id = "+id;
164         System.out.println(query);
165         ResultSet rs = st.executeQuery(query);
166         while (rs.next()) {
167             tanggal_harusKembali = rs.getString("tanggal_harus_kembali");
168         }
169     }
170     catch(SQLException e){
171         System.out.println("Error");
172     }
173     return tanggal_harusKembali;
174 }

```

Pada line 159 samapi 174 adalah statement untuk memberi tahu tanggal pengembalian.

```

177 public void KembalikanBuku(int id) throws ParseException{
178     long denda = 2000;
179     String _harusKembali = GetTanggalHarusKembali(id);
180     Date harusKembali =new SimpleDateFormat("yyyy-MM-dddd").parse(_harusKembali);
181     Date kembali = new Date();
182
183
184     if(kembali.after(harusKembali)){
185
186         SimpleDateFormat formatter = new SimpleDateFormat("yyyy-MM-dd");
187         String str1 = formatter.format(harusKembali);
188         String str2 = formatter.format(kembali);
189
190         LocalDate d1 = LocalDate.parse(str1, DateTimeFormatter.ISO_LOCAL_DATE);
191         LocalDate d2 = LocalDate.parse(str2, DateTimeFormatter.ISO_LOCAL_DATE);
192
193         Duration diff = Duration.between(d1.atStartOfDay(), d2.atStartOfDay());
194
195         long diffDays = diff.toDays();
196         denda= denda * diffDays;
197     }else{
198         denda = 0;
199     }
200
201     UpdateTanggalKembaliBuku(denda,id);
202 }
203 }

```

Pada line 177 sampai 201 adalah perhitungan denda yang dikenakan jika terjadi keterlambatan pengembalian buku dengan perhitungan sebagai berikut:

Rumus:

Keterlambatan = tanggal\_kembali – tanggal\_harus\_kembali

Denda = keterlambatan \* 2000

Biaya sewa = 5000

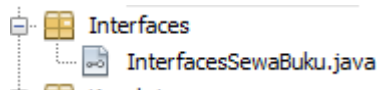
```

205 String query_updateKembaliBuku = "UPDATE sewabuku SET tanggal_kembali = ?,denda = ? WHERE id = ?";
206 public void UpdateTanggalKembaliBuku(long _denda,int id){
207     PreparedStatement statement = null;
208
209     try{
210         statement = con.prepareStatement(query_updateKembaliBuku);
211
212         SimpleDateFormat formatter = new SimpleDateFormat("yyyy-MM-dd");
213         Date dateNow = new Date();
214
215         statement.setString(1,formatter.format(dateNow));
216         statement.setLong(2, _denda);
217         statement.setInt(3, id);
218         statement.execute();
219     }
220     catch(SQLException e){
221         System.out.println("Gagal");
222     }
223     finally
224     {
225         try
226         {
227             statement.close();
228         } catch (SQLException ex)
229         {
230             System.out.println("Gagal");
231         }
232     }

```

Pada line 205 sampai 231 adalah statement yang dibuat untuk update tanggal pengembalian buku yang sudah dipinjam.

### 3. Interface



Di dalam package Interfaces terdapat file InterfacesSewaBuku.java

```

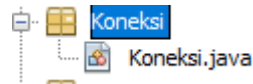
6 package Interfaces;
7
8 import Model.mTable; /*Untuk memanggil mTable.java dalam package Model*/
9 import java.util.List; /*Untuk memanggil package List pada java */
10
11 /**
12  *
13  * @author Firdana
14  */
15 public interface InterfacesSewaBuku {
16     public List<mTable> GetAll();
17 }
18

```

Pada line 6 sampai 17 digunakan sebagai penghubung interface ke dalam file InputanSewaBuku.java pada package Inputan.



#### 4. Koneksi



Pada package Koneksi terdapat file Koneksi.java

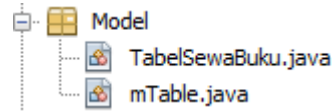
```
8 import com.mysql.cj.jdbc.MysqlDataSource; /*Untuk memanggil package mysql cj jdbc MysqlDataSource pada java */
9 import java.sql.Connection; /*Untuk memanggil package sql Connection pada java */
10 import java.sql.SQLException; /*Untuk memanggil package sql SQLException pada java */
```

Pada line 8 sampai 10 dalam file Koneksi.java digunakan untuk pemanggilan file sql dalam packages yang akan dipanggil

```
15 public class Koneksi {
16     static Connection con;
17     static String db_name = "sewabuku";
18     static String db_user = "root";
19     static String db_pass = "";
20
21     public static Connection GetConnection() {
22         if (con == null) {
23             MysqlDataSource data = new MysqlDataSource();
24             data.setDatabaseName(db_name);
25             data.setUser(db_user);
26             data.setPassword(db_pass);
27
28             try {
29                 con = data.getConnection();
30                 System.out.println("Database connect");
31             }
32             catch (SQLException e) {
33                 System.out.println("Database tidak connect");
34             }
35         }
36         return con;
37     }
38 }
```

Pada line 15 sampai 38 digunakan untuk menghubungkan antara program dan database MYSQL agar terkoneksi ke program.

## 5. Model



Pada package Model terdapat dua file yaitu TabelSewaBuku.Java dan mTable.java

```
8 import java.util.List; /*Untuk memanggil package List pada java */
9 import javax.swing.table.AbstractTableModel; /*Untuk memanggil swing table AbstractTableModel pada java*/
```

Pada line 8 sampai 9 dalam file TebleSewaBuku.java digunakan untuk pemanggilan file dalam package yang akan dipanggil.

```
14 public class TabelSewaBuku extends AbstractTableModel{
15     List<mTable> list_sewaBuku;
16
17     public TabelSewaBuku(List<mTable> list_sewaBuku){
18         this.list_sewaBuku = list_sewaBuku;
19     }
20
21     @Override
22     public int getRowCount() {
23         return this.list_sewaBuku.size();
24     }
25
26     @Override
27     public int getColumnCount() {
28         return 7;
29     }
30
31     public Object getValueAt(int rowIndex, int columnIndex) {
32         switch (columnIndex) {
33             case 0:
34                 return list_sewaBuku.get(rowIndex).id;
35             case 1:
36                 return list_sewaBuku.get(rowIndex).judul;
37             case 2:
38                 return list_sewaBuku.get(rowIndex).tanggal_pinjam;
39             case 3:
40                 return list_sewaBuku.get(rowIndex).tanggal_harus_kembali;
41             case 4:
42                 return list_sewaBuku.get(rowIndex).tanggal_kembali;
43             case 5:
44                 return list_sewaBuku.get(rowIndex).denda;
45             case 6:
46                 return list_sewaBuku.get(rowIndex).biaya_sewa;
47             default :
48                 return null;
49         }
50     }
51 }
```

```

53      @Override
54      public String getColumnName(int column) {
55          switch (column) {
56              case 0:
57                  return "ID";
58              case 1:
59                  return "Judul";
60              case 2:
61                  return "Tanggal Pinjam";
62              case 3:
63                  return "Tanggal Harus Kembali";
64              case 4:
65                  return "Tanggal Kembali";
66              case 5:
67                  return "Denda";
68              case 6:
69                  return "Biaya Sewa";
70              default :
71                  return null;
72          }
73      }
74  }
75

```

Pada line 14 sampai 75 dalam file TabelSewaBuku.java digunakan untuk membuat tampilan table dari table sql database ke UI.

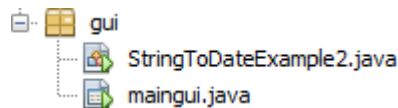
```

6      package Model;
7
8      /**
9       *
10      * @author Firdana
11      */
12      public class mTable {
13          public int id;
14          public String judul;
15          public String tanggal_pinjam;
16          public String tanggal_kembali;
17          public String tanggal_harus_kembali;
18          public int denda;
19          public int biaya_sewa;
20      }
21

```

Pada line 6 samapai 21 dalam file mTable.java digunakan untuk membuat variable table yang nantinya akan dijadikan list dan menumpang data dari database sql.

## 6. gui



Pada package gui terdapat dua file yaitu StringToDateExample.java dan maingui.java

```
3 import java.text.SimpleDateFormat; /*Untuk memanggil package text SimpleDataFormat pada java*/
4 import java.time.Duration; /*Untuk memanggil package time Duration pada java*/
5 import java.time.LocalDate; /*Untuk memanggil package time LocalDate*/
6 import java.time.Period; /*Untuk memanggil package time period*/
7 import java.time.format.DateTimeFormatter; /*Untuk memanggil package format DateTimeFormatter pada java*/
8 import java.util.Calendar; /*Untuk memanggil package Calendar pada java*/
9 import java.util.Date; /*Untuk memanggil package util Data pada java*/

10 public class StringToDateExample2 {
11     public static void main(String[] args) throws Exception {
12
13         Date harusKembali = new Date();
14         Calendar c = Calendar.getInstance();
15         c.setTime(harusKembali);
16         c.add(Calendar.DATE, 7);
17         Date kembali = c.getTime();
18
19
20         if(kembali.after(harusKembali)){
21
22
23         }
24         SimpleDateFormat formatter = new SimpleDateFormat("yyyy-MM-dd");
25         String str1 = formatter.format(harusKembali);
26         String str2 = formatter.format(kembali);
27
28         LocalDate d1 = LocalDate.parse(str1, DateTimeFormatter.ISO_LOCAL_DATE);
29         LocalDate d2 = LocalDate.parse(str2, DateTimeFormatter.ISO_LOCAL_DATE);
30
31         Duration diff = Duration.between(d1.atStartOfDay(), d2.atStartOfDay());
32
33         long diffDays = diff.toDays();
34         System.out.println("Difference between dates is : "+diffDays + "days");
35     }
36 }
```

Pada line 10 sampai 35 dalam file StringToDateExample.java digunakan untuk perhitungan tanggal Kembali dan harus Kembali.

## Persewaan Buku XYZ

Tanggal : tanggal
Jam : jam

Judul : 
Biaya :

Rp

Simpan
Kembalikan Buku
Edit
Delete

ID	Judul Buku	Tanggal Pinjam	Tanggal Harus Kembali	Tanggal Kembali	Denda	Biaya Sewa

Gambar berikut adalah desain gui dari program Persewaan Buku XYZ

### Program Berjalan

ID	Judul	Tanggal Pinjam	Tanggal Harus K...	Tanggal Kembali	Denda	Biaya Sewa
8	buku	12 July 2021	19 July 2021	12 July 2021	0	5000

