

Nama : St. Hatijah H. Ilyas

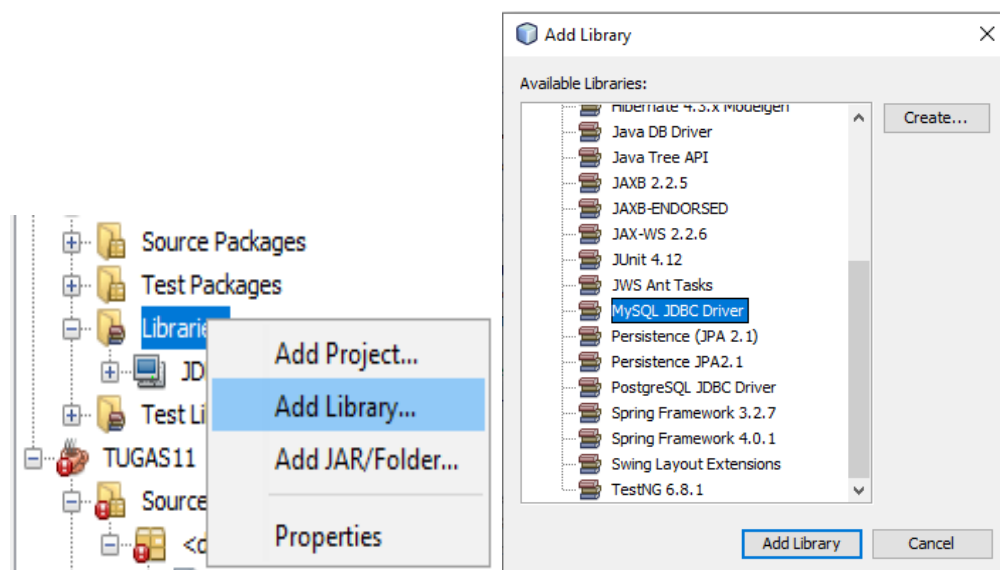
Stambuk : 13020180061

Kelas : B1

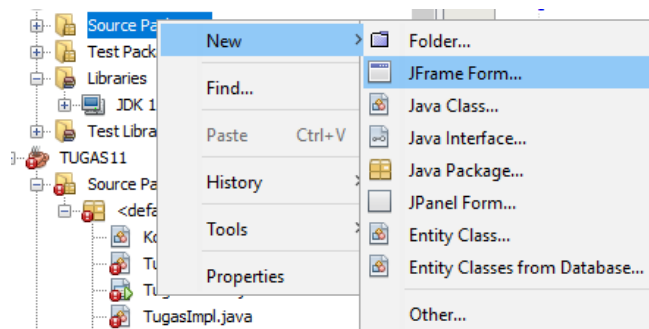
1. Membuat database di MySQL Workbench

```
1 • create database Mahasiswaaa;  
2 • use Mahasiswaaa;  
3  
4 • create table table_mahasiswaa ( `Nomhs` varchar(5) primary key,  
5   `Nama` varchar(30),  
6   `Jurusan` varchar(15));|
```

2. Buka Netbeans, buat nama Project sesuai selera. Kemudian klik kanan pada *libraries*, pilih *Add Library* dan pilih *MySQL JDBC Driver*, yang berfungsi untuk menghubungkan Database ke Netbeans.



3. Kemudian buat Form Class dengan klik kanan pada *Source Packages*, pilih *New* kemudian pilih *JFrame Form*.



4. Buat desain seperti ini

The design preview shows a window titled "Design Preview [TugasModel]". It contains two main sections: "INPUT DATA" and "DATA MAHASISWA".

**INPUT DATA**

Fields: Stambuk, Jurusan, Nama, Email.

Buttons: ADD, RESET.

**DATA MAHASISWA**

Search field: Cari Berdasarkan Nama.

Table with 4 columns: Title 1, Title 2, Title 3, Title 4.

Buttons: EDIT, DELETE.

5. Untuk mengubah *Title 1*, *Title 2*, *Title 3* dan *Title 4* menggunakan listing dibawah ini

```
28 public class TugasModel extends javax.swing.JFrame {
29     DefaultTableModel model;
30     /**
31      * Creates new form TugasModel
32      */
33     public TugasModel() {
34         initComponents();
35         String []judul={"Stambuk","Nama","Jurusan","Email"};
36         model = new DefaultTableModel(judul,0);
37         jTable1.setModel(model);
38         tampilkan();
39     }
```

Sehingga menjadi:

The image shows two side-by-side design previews of the "TugasModel" window, illustrating the change in the table headers.

**Left Preview:** The table headers are "Title 1", "Title 2", "Title 3", and "Title 4". These headers are highlighted with a red rectangle.

**Right Preview:** The table headers are "Stambuk", "Nama", "Jurusan", and "Email". These headers are highlighted with a red rectangle.

Listing dibawah merupakan method *tampilkan* yang berfungsi untuk menampilkan hasil inputan yang kita masukkan ke dalam table yang ada di form. Pada kotak merah yang telah ditandai merupakan *Driver JDBC* untuk meng-koneksikan database, dengan *localhost 3306*, nama database yang telah dibuat yaitu *Mahasiswaaa*, nama user *root* dan *password* (dikosongkan saja).

```

424 private void tampilkan() {
425     int row = jTable1.getRowCount();
426     for(int a=0; a<row; a++){
427         model.removeRow(0);
428     }
429     try {
430         Connection cn = DriverManager.getConnection("jdbc:mysql://localhost:3306/Mahasiswaaa","root","");
431         ResultSet rs = cn.createStatement().executeQuery("select * from table_mahasiswaaa");
432         while(rs.next()){
433             String data[]={rs.getString(1),rs.getString(2),rs.getString(3),rs.getString(4)};
434             model.addRow(data);
435         }
436     } catch (SQLException ex) {
437         Logger.getLogger(TugasModel.class.getName()).log(Level.SEVERE, null, ex);
438     }
439 }

```

- Untuk mengaktifkan *Action Performed* pada tombol Button *ADD* yang berfungsi untuk menambahkan data dan dimasukkan kedalam table, dengan menggunakan listing dibawah

```

291 private void AddActionPerformed(java.awt.event.ActionEvent evt) {
292     try {
293         Connection cn = DriverManager.getConnection("jdbc:mysql://localhost:3306/Mahasiswaaa","root","");
294         cn.createStatement().executeUpdate("insert into table_mahasiswaaa values"+"('"+txtStb.getText()+"','"+
295             + "'"+txtNama.getText()+"','"+txtJurusan.getText()+"','"+txtEmail.getText()+"')");
296         tampilkan();
297         reset();
298     } catch (SQLException ex) {
299         Logger.getLogger(TugasModel.class.getName()).log(Level.SEVERE, null, ex);
300     }
301 }

```

- Untuk mengaktifkan *Action Performed* pada tombol Button *RESET* menggunakan listing dibawah

```

326 private void ResetActionPerformed(java.awt.event.ActionEvent evt) {
327     // TODO add your handling code here:
328     reset();
329     tampilkan();
330 }

```

Pada method *Reset* menggunakan listing dibawah ini yang berfungsi untuk merefresh/ingin menghapus inputan yang diketikkan.

```

441 private void reset() {
442     txtStb.setText("");
443     txtNama.setText("");
444     txtJurusan.setText("");
445     txtEmail.setText("");
446     txtCari.setText("");
447 }
448 }

```

8. Untuk mengaktifkan *Action Performed* pada tombol Button *EDIT* yang berfungsi untuk mengganti/update data yang telah dimasukkan kedalam table sebelumnya, dengan menggunakan listing dibawah:

```
302 private void EditActionPerformed(java.awt.event.ActionEvent evt) {  
303     try {  
304         // TODO add your handling code here:  
305         Connection cn = DriverManager.getConnection("jdbc:mysql://localhost:3306/Mahasiswaaa","root","");  
306         cn.createStatement().executeUpdate("update table_mahasiswaa set Nama='"+txtNama.getText()+"',"  
307             + "Jurusan='"+txtJurusan.getText()+"',Email='"+txtEmail.getText()+"' "  
308             + "where Stambuk='"+txtStb.getText()+"'");  
309         tampilkan();  
310         reset();  
311     } catch (SQLException ex) {  
312         Logger.getLogger(TugasModel.class.getName()).log(Level.SEVERE, null, ex);  
313     }  
314 }
```

9. Untuk mengaktifkan *Action Performed* pada tombol Button *DELETE* yang berfungsi untuk menghapus data yang telah dimasukkan ke dalam table sebelumnya, dengan menggunakan listing berikut:

```
331 private void DeleteActionPerformed(java.awt.event.ActionEvent evt) {  
332     try {  
333         // TODO add your handling code here:  
334         Connection cn = DriverManager.getConnection("jdbc:mysql://localhost:3306/Mahasiswaaa","root","");  
335         cn.createStatement().executeUpdate("delete from table_mahasiswaa where Stambuk='"+txtStb.getText()+"'");  
336         tampilkan();  
337         reset();  
338     } catch (SQLException ex) {  
339         Logger.getLogger(TugasModel.class.getName()).log(Level.SEVERE, null, ex);  
340     }  
341 }
```

10. Untuk mengaktifkan *Action Performed* pada tombol Button *CARI* yang berfungsi untuk mencari data yang telah dimasukkan sebelumnya berdasarkan nama. Jadi kita hanya bisa mencari data berdasarkan nama yang diketikkan, dengan menggunakan listing berikut:

```
343 private void CARIActionPerformed(java.awt.event.ActionEvent evt) {  
344     // TODO add your handling code here:  
345     int row = jTable1.getRowCount();  
346     for(int a=0; a<row; a++){  
347         model.removeRow(0);  
348     }  
349     try {  
350         Connection cn = DriverManager.getConnection("jdbc:mysql://localhost:3306/Mahasiswaaa","root","");  
351         ResultSet rs = cn.createStatement().executeQuery("select * from table_mahasiswaa where "  
352             + "Nama='"+txtCari.getText()+"'");  
353         while(rs.next()){  
354             String data[]={rs.getString(1),rs.getString(2),rs.getString(3),rs.getString(4)};  
355             model.addRow(data);  
356         }  
357     } catch (SQLException ex) {  
358         Logger.getLogger(TugasModel.class.getName()).log(Level.SEVERE, null, ex);  
359     }  
360 }
```

## OUTPUT TAMPILAN AWAL

The screenshot shows a window titled "INPUT DATA" with four input fields: "Stambuk", "Jurusan", "Nama", and "Email". Below these fields are two buttons: "ADD" and "RESET". Below the input section is a section titled "DATA MAHASISWA" which includes a search bar with a "Cari" button and a table with four columns: "Stambuk", "Nama", "Jurusan", and "Email". The table is currently empty. At the bottom of the window are "EDIT" and "DELETE" buttons.

**INPUT DATA**

Stambuk  Jurusan

Nama  Email

**DATA MAHASISWA**

Cari Berdasarkan Nama

Stambuk	Nama	Jurusan	Email
---------	------	---------	-------

## OUTPUT TAMPILAN ADD

The left screenshot shows the "INPUT DATA" section with the following values: Stambuk: 068, Jurusan: Teknik, Nama: Hatijah, Email: hatijahh@gmail.com. These fields are enclosed in a red rectangular box. The "ADD" button is highlighted. The "DATA MAHASISWA" table is empty.

The right screenshot shows the same interface after the "ADD" action. The "DATA MAHASISWA" table now contains one row with the following data: Stambuk: 068, Nama: Hatijah, Jurusan: Teknik, Email: hatijahh@gmail.com. This row is highlighted with a red rectangular box.

**INPUT DATA**

Stambuk 068 Jurusan Teknik

Nama Hatijah Email hatijahh@gmail.com

**DATA MAHASISWA**

Cari Berdasarkan Nama

Stambuk	Nama	Jurusan	Email
---------	------	---------	-------

## OUTPUT TAMPILAN RESET

The image shows two side-by-side screenshots of a web application interface. Both screenshots display the 'INPUT DATA' section at the top, which contains four input fields: 'Stambuk' (003), 'Jurusan' (Ekonomi), 'Nama' (Ulfa), and 'Email' (Uppa12@gmail.com). A red rectangular box highlights these input fields. Below the input fields are two buttons: 'ADD' and 'RESET'. The 'RESET' button is highlighted with a blue border. Below the input section is the 'DATA MAHASISWA' section, which includes a search bar with the text 'Cari Berdasarkan Nama' and a 'Cari' button. Below the search bar is a table with the following data:

Stambuk	Nama	Jurusan	Email
068	Hatijah	Teknik	hatijahh@gmail.com

At the bottom of the 'DATA MAHASISWA' section are two buttons: 'EDIT' and 'DELETE'.

## OUTPUT TAMPILAN EDIT

The image shows two side-by-side screenshots of a web application interface. Both screenshots display the 'INPUT DATA' section at the top, which contains four input fields: 'Stambuk', 'Jurusan', 'Nama', and 'Email'. A red rectangular box highlights these input fields. Below the input fields are two buttons: 'ADD' and 'RESET'. The 'RESET' button is highlighted with a blue border. Below the input section is the 'DATA MAHASISWA' section, which includes a search bar with the text 'Cari Berdasarkan Nama' and a 'Cari' button. Below the search bar is a table with the following data:

Stambuk	Nama	Jurusan	Email
006	Dina	Ekonomi	Dinaatae@gmail.com
068	Hatijah	Teknik	hatijahh@gmail.com

At the bottom of the 'DATA MAHASISWA' section are two buttons: 'EDIT' and 'DELETE'.

## OUTPUT TAMPILAN DELETE

### INPUT DATA

Stambuk

Jurusan

Nama

Email

ADD

RESET

### DATA MAHASISWA

Cari

Cari Berdasarkan Nama

Stambuk	Nama	Jurusan	Email
006	Dina	Hukum	Dinaatae@gmail.com
068	Hatijah	Teknik	hatijahh@gmail.com

EDIT

DELETE

### INPUT DATA

Stambuk

Jurusan

Nama

Email

ADD

RESET

### DATA MAHASISWA

Cari

Cari Berdasarkan Nama

Stambuk	Nama	Jurusan	Email
006	Dina	Hukum	Dinaatae@gmail.com

EDIT

DELETE

## OUTPUT TAMPILAN CARI

### INPUT DATA

Stambuk

Jurusan

Nama

Email

ADD

RESET

### DATA MAHASISWA

Cari

Cari Berdasarkan Nama

Stambuk	Nama	Jurusan	Email
003	Uifa	Perpajakan	Uppaaa@gmail.com
006	Dina	Hukum	Dinaatae@gmail.com
061	Hatijah	Teknik	Hatijahh@gmail.com
068	Tifa	Industri	Tippaa@gmail.com

EDIT

DELETE

### INPUT DATA

Stambuk

Jurusan

Nama

Email

ADD

RESET

### DATA MAHASISWA

Cari

Cari Berdasarkan Nama

Stambuk	Nama	Jurusan	Email
003	Uifa	Perpajakan	Uppaaa@gmail.com

EDIT

DELETE