



PEMROGRAMAN BERORIENTASI OBJEK

LAPORAN DOKUMENTASI

Paramadina Mulya Majid

130201080006

B1

- Tugas 11 -

PEMROGRAMAN BERORIENTASI OBJEK

PROSES PEMBUATAN APLIKASI

1. Pembuatan Database

Pada database ini terdapat table "*tabel_mahasiswa*" dengan field Nomhs, Nama dan Jurusan. Untuk jangkauan dari jurusan saya mengubahnya sedikit dari spesifikasi yang telah ditentukan sebelumnya yaitu 15 dikarenakan untuk jurusan minimal karakter yang digunakan yaitu 16 untuk "sistem informasi" dan 18 untuk "teknik informatika".

```
1 • create database mahasiswa;  
2 • use mahasiswa;  
3  
4 • create table tabel_mahasiswa(Nomhs varchar(5) primary key, Nama varchar(30), Jurusan varchar(25));
```

2. Pembuatan Class Koneksi Database agar database terhubung dengan aplikasi yang dibuat menggunakan netbeans

```
1  /*  
2      Nama      : Paramadina Mulya Majid  
3      Stambuk   : 13020180006  
4      Hari/Tgl  : Kamis, 11 Juni 2020  
5      Waktu    : 21.14 WITA  
6  */  
7  package Source;  
8  
9  import java.sql.Connection;  
10 import java.sql.DriverManager;  
11 import java.sql.SQLException;  
12 import java.util.logging.Level;  
13 import java.util.logging.Logger;  
14  
15 public class koneksiDatabase {  
16     private static Connection con;  
17     public static Connection getConnection(){  
18         if(con==null){  
19             try {  
20                 DriverManager.registerDriver(new com.mysql.jdbc.Driver());  
21                 con=DriverManager.getConnection("jdbc:mysql://localhost:3306/mahasiswa","root","");  
22             } catch (SQLException ex) {  
23                 Logger.getLogger(koneksiDatabase.class.getName()).log(Level.SEVERE, null, ex);  
24             }  
25         }  
26         return con;  
27     }  
28 }
```

3. Pembuatan Class ImplementMhs yang didalamnya berisi fungsi-fungsi seperti insert data pada table, update data, menghapus data dan menampilkan data.

```
1  /*
2     Nama      : Paramadina Mulya Majid
3     Stambuk   : 19020180006
4     Hari/Tgl  : Kamis, 11 Juni 2020
5     Waktu    : 21.21 WITA
6  */
7  package Source;
8
9  import java.sql.PreparedStatement;
10 import java.sql.ResultSet;
11 import java.sql.SQLException;
12 import java.sql.Statement;
13 import java.util.ArrayList;
14 import java.util.List;
15
16 public class ImplementMhs implements interfaceMhs{
17
18     @Override
19     public mahasiswa insert(mahasiswa o) throws SQLException {
20         PreparedStatement st = koneksiDatabase.getConnection().prepareStatement("insert into tabel_mahasiswa values(?,?,?)");
21         st.setString(1, o.getNomhs());
22         st.setString(2, o.getNama());
23         st.setString(3, o.getJurusan());
24         st.executeUpdate();
25         return o;
26     }
27
28     @Override
29     public void update(mahasiswa o, String stbAwal) throws SQLException {
30         PreparedStatement st = koneksiDatabase.getConnection().prepareStatement("update tabel_mahasiswa set Nama='"+o.getNama()+"',Jurusan='"+o.getJurusan()+"', Nomhs='"+o.getNomhs()+"' where Nomhs='"+stbAwal+"'");
31         // st.setString(1, o.getNomhs());
32         // st.setString(2, o.getNama());
33         // st.setString(3, o.getJurusan());
34         st.executeUpdate();
35     }
36
37     @Override
38     public void delete(String Nomhs) throws SQLException {
39         PreparedStatement st = koneksiDatabase.getConnection().prepareStatement("delete from tabel_mahasiswa where Nomhs=?");
40         st.executeUpdate();
41     }
42
43     @Override
44     public List getAll() throws SQLException {
45         Statement st = koneksiDatabase.getConnection().createStatement();
46         ResultSet rs = st.executeQuery("select * from tabel_mahasiswa");
47         List list = new ArrayList();
48         while (rs.next()) {
49             mahasiswa p = new mahasiswa();
50             p.setNomhs(rs.getString("Nomhs"));
51             p.setNama(rs.getString("Nama"));
52             p.setJurusan(rs.getString("Jurusan"));
53             list.add(p);
54         }
55         return list;
56     }
57
58     @Override
59     public List getSearch(String code) throws SQLException {
60         Statement st = koneksiDatabase.getConnection().createStatement();
61         ResultSet rs = st.executeQuery("select * from tabel_mahasiswa where Nama LIKE '"+code+"%'"");
62         List list = new ArrayList();
63         while (rs.next()) {
64             mahasiswa p = new mahasiswa();
65             p.setNomhs(rs.getString("Nomhs"));
66             p.setNama(rs.getString("Nama"));
67             p.setJurusan(rs.getString("Jurusan"));
68             list.add(p);
69         }
70         return list;
71     }
72 }
```

4. Pembuatan Class interfaceMhs

```
1  /*
2      Nama      : Paramadina Mulya Majid
3      Stambuk   : 13020180006
4      Hari/Tgl  : Kamis, 11 Juni 2020
5      Waktu    : 21.22 WITA
6  */
7  package Source;
8
9  import java.sql.SQLException;
10 import java.util.List;
11
12 public interface interfaceMhs {
13     mahasiswa insert(mahasiswa o) throws SQLException;
14     void update(mahasiswa o, String stbAwal) throws SQLException;
15     void delete(String Nomhs) throws SQLException;
16     List getAll() throws SQLException;
17     List getSearch(String code) throws SQLException;
18 }
```

5. Pembuatan Class mahasiswa

```
1  /*
2      Nama      : Paramadina Mulya Majid
3      Stambuk   : 13020180006
4      Hari/Tgl  : Kamis, 11 Juni 2020
5      Waktu    : 21.37 WITA
6  */
7  package Source;
8
9  import java.util.Objects;
10
11 public class mahasiswa {
12     String Nomhs;
13     String Nama;
14     String Jurusan;
15
16     public mahasiswa() { }
17     public mahasiswa(String Nomhs, String Nama, String Jurusan) {
18         this.Nomhs = Nomhs;
19         this.Nama = Nama;
20         this.Jurusan = Jurusan;
21     }
22     public String getJurusan() {
23         return Jurusan;
24     }
25     public void setJurusan(String Jurusan) {
26         this.Jurusan = Jurusan;
27     }
28     public String getNama() {
29         return Nama;
30     }
31     public void setNama(String Nama) {
32         this.Nama = Nama;
33     }
34     public String getNomhs() {
35         return Nomhs;
36     }
37     public void setNomhs(String Nomhs) {
38         this.Nomhs = Nomhs;
39     }
40     @Override
41     public String toString() {
42         return "Mahasiswa{" + "Nomhs=" + Nomhs + ", Nama=" + Nama + ", Jurusan=" + Jurusan + '}';
43     }
44 }
```

```

44  @Override
45  public int hashCode() {
46      int hash = 3;
47      hash = 89 * hash + Objects.hashCode(this.Nomhs);
48      hash = 89 * hash + Objects.hashCode(this>Nama);
49      hash = 89 * hash + Objects.hashCode(this.Jurusan);
50      return hash;
51  }
52  @Override
53  public boolean equals(Object obj) {
54      if (this == obj) {
55          return true;
56      }
57      if (obj == null) {
58          return false;
59      }
60      if (getClass() != obj.getClass()) {
61          return false;
62      }
63      final mahasiswa other = (mahasiswa) obj;
64      if (!Objects.equals(this.Nomhs, other.Nomhs)) {
65          return false;
66      }
67      if (!Objects.equals(this>Nama, other>Nama)) {
68          return false;
69      }
70      if (!Objects.equals(this.Jurusan, other.Jurusan)) {
71          return false;
72      }
73      return true;
74  }

```

6. Pembuatan Class dataSiswaFikom include dengan desain aplikasi menggunakan JFrame Form

a. Pembuatan Action pada textField No Mhs

```

256 private void txtNomhsActionPerformed(java.awt.event.ActionEvent evt) {
257     // TODO add your handling code here:
258     Simpan.setEnabled(true);
259     Edit.setEnabled(false);
260     Hapus.setEnabled(false);
261 }

```

b. Pembuatan Action pada textfield Nama

```
263 private void txtNamaActionPerformed(java.awt.event.ActionEvent evt) {  
264     // TODO add your handling code here:  
265     Simpan.setEnabled(true);  
266     Edit.setEnabled(false);  
267     Hapus.setEnabled(false);  
268 }
```

c. Pembuatan method pada ComboBox Jurusan

```
48 void daftarJurusan() {  
49     Object[] jur = {"Teknik Informatika", "Sistem Informasi"};  
50     for (int i = 0; i < jur.length; i++) {  
51         CbJurusan.addItem(jur[i].toString());  
52     }  
53     kJur = CbJurusan.getSelectedItem().toString();  
54 }
```

d. Pembuatan Action pada button Simpan

```
295 private void SimpanActionPerformed(java.awt.event.ActionEvent evt) {  
296  
297     try {  
298         if (txtNomhs.getText().equals("") && txtNama.getText().equals("") && CbJurusan.getSelectedItem().equals("")) {  
299             JOptionPane.showMessageDialog(this, "silahkan isi data yang akan ditambah");  
300         } else {  
301             mahasiswa p = new mahasiswa();  
302             p.setNomhs(txtNomhs.getText());  
303             p.setNama(txtNama.getText());  
304             p.setJurusan((String) CbJurusan.getSelectedItem());  
305             InterfaceMhs.insert(p);  
306             JOptionPane.showMessageDialog(this, "Data tersimpan");  
307             ambil_data(record = InterfaceMhs.getAll());  
308             bersih();  
309         }  
310     } catch (SQLException ex) {  
311         JOptionPane.showMessageDialog(this, ex);  
312     }  
313 }
```


e. Pembuatan Action pada button Edit

```
275 private void EditActionPerformed(java.awt.event.ActionEvent evt) {  
276     try {  
277         mahasiswa mhs = new mahasiswa();  
278         mhs.setNomhs(txtNomhs.getText().trim());  
279         mhs.setNama(txtNama.getText().trim());  
280         mhs.setJurusan(String.valueOf(CbJurusan.getSelectedItem()).trim());  
281         InterfaceMhs.update(mhs, stbAwal);  
282         // this.statusAwal();  
283         JOptionPane.showMessageDialog(this, "data berhasil diubah");  
284         bersih();  
285         ambil_data(record = InterfaceMhs.getAll());  
286     } catch (SQLException ex) {  
287         // System.out.println("Tidak Masuk");  
288         System.out.println(""+ex.getMessage());  
289         Logger.getLogger(dataSiswaFikom.class.getName()).log(Level.SEVERE, null, ex);  
290     }  
291 }  
292  
293 }
```

f. Pembuatan Action pada button Hapus

```
315 private void HapusActionPerformed(java.awt.event.ActionEvent evt) {  
316     // TODO add your handling code here:  
317     try {  
318         String Nomhs = txtNomhs.getText();  
319         InterfaceMhs.delete(Nomhs);  
320         // this.statusAwal();  
321         ambil_data(record = InterfaceMhs.getAll());  
322         JOptionPane.showMessageDialog(this, "data berhasil dihapus");  
323     } catch (SQLException ex) {  
324         Logger.getLogger(dataSiswaFikom.class.getName()).log(Level.SEVERE, null, ex);  
325     }  
326 }
```

g. Pembuatan method pada button Cari

```
361 private void CariActionPerformed(java.awt.event.ActionEvent evt) {  
362     String cari = txtCari.getText().trim();  
363     try {  
364         if(cari==null){  
365             ambil_data(record = InterfaceMhs.getAll());  
366         }else{  
367             ambil_data(record = InterfaceMhs.getSearch(cari));  
368         }  
369     } catch (SQLException ex) {  
370         Logger.getLogger(dataSiswaFikom.class.getName()).log(Level.SEVERE, null, ex);  
371     }  
372 }
```

LUARAN APLIKASI

Aplikasi ini dibuat untuk melakukan penginputan data mahasiswa yang berada di fakultas ilmu komputer UMI. Data tersebut terdiri atas No Mahasiswa, Nama Mahasiswa dan Jurusan (Teknik Informatika dan Sistem Informasi). Pada aplikasi ini terdapat fungsi simpan, edit, hapus dan melakukan pencarian data berdasarkan nama yang diinputkan

The screenshot displays the application's main interface. At the top, a banner reads "DATA MAHASISWA FAKULTAS ILMU KOMPUTER UMI". Below this, there are input fields for "NO MHS", "NAMA", and "JURUSAN". To the right of these fields are buttons for "CARI" (Search), "SIMPAN" (Save), "EDIT", and "HAPUS" (Delete). Below the input fields is a table with columns "NOMHS", "NAMA", and "JURUSAN". The table is currently empty. The background features a colorful, abstract pattern.

This screenshot shows the same application interface as above, but with additional labels and arrows indicating the component structure. The labels on the left side are: "Form dataSiswaFikom", "Other Components", "[Frame]", "AbsoluteLayout", "btNomhs [JTextField]", "btNama [JTextField]", "CbJurusan [JComboBox]", "Edit [JButton]", "Simpn [JButton]", "btCari [JTextField]", "Cari [JButton]", "jScrollPane2 [JScrollPane]", "Hapus [JButton]", and "ider jLabel1 [JLabel]". Arrows point from these labels to the corresponding components in the application window. The table in the screenshot has a header row with "NIM", "Nama", and "Jurusan", and several empty rows below it.

Detail Components

1. Fungsi Pencarian Data

Pada fungsi ini terdapat sebuah textfield dengan variabel "*txtCari*" dan Button dengan variabel "*Cari*".



2. Kolom Inputan No Mahasiswa

Pada kolom ini terdapat sebuah textfield dengan variabel "*txtNomhs*" yang digunakan untuk menginput No Mahasiswa



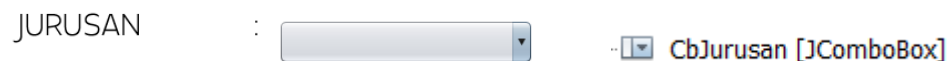
3. Kolom Inputan Nama

Pada kolom ini terdapat sebuah textfield dengan variabel "*txtNama*" yang digunakan untuk menginput Nama Mahasiswa



4. Pilihan Jurusan

Pada bagian ini terdapat sebuah comboBox dengan variabel "*Cbjurusan*" yang digunakan untuk memilih jurusan sesuai dengan pilihan yang ada yaitu "Teknik Informatika" dan "Sistem Informasi".



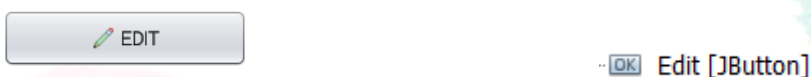
5. Tombol Simpan

Pada tombol ini terdapat sebuah Button dengan variabel "*Simpan*" yang digunakan ketika kita telah menginputkan data seperti No Mahasiswa, Nama dan Jurusan maka untuk menyimpan data tersebut kita menekan tombol Simpan. Ketika kita telah menekan tombol Simpan maka akan tampil pesan "Data Tersimpan"



6. Tombol Edit

Pada tombol ini terdapat sebuah Button dengan variabel "*Edit*" yang digunakan ketika kita ingin mengubah data yang telah diinputkan seperti No Mahasiswa, Nama dan Jurusan maka untuk mengubah data tersebut kita menekan tombol Edit.



7. Tombol Hapus

Pada tombol ini terdapat sebuah Button dengan variabel "*Hapus*" yang digunakan ketika kita ingin menghapus data yang telah diinputkan maka untuk menghapus data tersebut kita menekan tombol Hapus.



OK Hapus [JButton]

DESAIN INPUT

The screenshot shows a window titled "DATA MAHASISWA FAKULTAS ILMU KOMPUTER UMI". The form contains the following fields:

- NO MHS: 2018
- NAMA: Paramadina Mulya
- JURUSAN: Teknik Informatika (selected from a dropdown menu showing options: Teknik Informatika, Teknik Informatika, Sistem Informasi)

Below the form is a table with columns: NOMHS, NAMA, JURUSAN. The table is currently empty.

On the right side, there are buttons: "CARI", "SIMPAN", "EDIT", and "HAPUS".

DESAIN OUTPUT

The screenshot shows the same window as the input design, but with data entered into the form:

- NO MHS: 2018
- NAMA: Paramadina Mulya
- JURUSAN: Teknik Informatika

The table below the form now contains one row of data:

NOMHS	NAMA	JURUSAN
2018	Paramadina Mulya	Teknik Informatika

The buttons "CARI", "SIMPAN", "EDIT", and "HAPUS" are still present on the right side.

CONTOH PENERAPAN

1. Menyimpan Data

a. Menginputkan data terlebih dahulu

DATA MAHASISWA FAKULTAS ILMU KOMPUTER UMI

NO MHS : 2018

NAMA : Paramadina Mulya

JURUSAN : Teknik Informatika

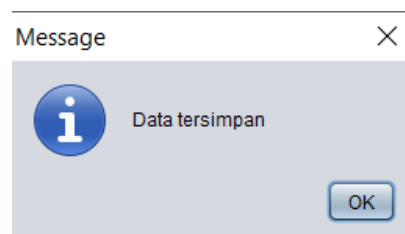
SIMPAN

NOMHS	NAMA	JURUSAN
-------	------	---------

EDIT

HAPUS

b. Menekan tombol simpan ketika telah menginputkan data, jika berhasil maka akan tampil seperti gambar dibawah ini lalu klik ok.



c. Data tersebut telah tersimpan

DATA MAHASISWA FAKULTAS ILMU KOMPUTER UMI

NO MHS :

NAMA :

JURUSAN :

SIMPAN

NOMHS	NAMA	JURUSAN
2018	Paramadina Mulya	Teknik Informatika

EDIT

HAPUS

2. Mengedit Data

a. Klik data yang ingin diubah

DATA MAHASISWA FAKULTAS ILMU KOMPUTER UMI

NO MHS : 2018
NAMA : Paramadina Mulya
JURUSAN : Teknik Informatika

NOMHS	NAMA	JURUSAN
2018	Paramadina Mulya	Teknik Informatika
2019	Majid	Sistem Informasi

Buttons: CARI, SIMPAN, EDIT, HAPUS

b. Ubahlah sesuai yang kita diinginkan, pada percobaan ini saya mengubah No Mahasiswa menjadi "2017" dan Nama menjadi "Dina"

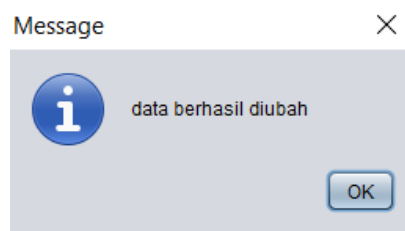
DATA MAHASISWA FAKULTAS ILMU KOMPUTER UMI

NO MHS : 2017
NAMA : Dina
JURUSAN : Teknik Informatika

NOMHS	NAMA	JURUSAN
2018	Paramadina Mulya	Teknik Informatika
2019	Majid	Sistem Informasi

Buttons: CARI, SIMPAN, EDIT, HAPUS

c. Tekan tombol edit. Jika berhasil akan muncul tampilan seperti gambar dibawah ini kemudian klik ok dan data tersebut telah berubah.

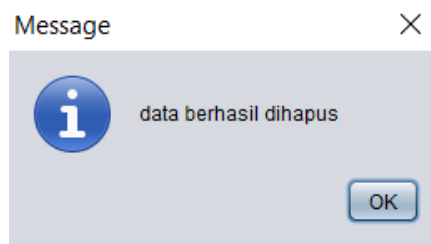


NOMHS	NAMA	JURUSAN
2017	Dina	Teknik Informatika
2019	Majid	Sistem Informasi

3. Menghapus Data

a. Klik data yang ingin dihapus

b. Tekan tombol hapus. Jika berhasil akan muncul tampilan seperti gambar dibawah ini kemudian klik ok dan data tersebut telah berhasil terhapus.



NOMHS	NAMA	JURUSAN
2017	Dina	Teknik Informatika

4. Mencari Data

a. Cari data berdasarkan nama yang ingin dicari.

NOMHS	NAMA	JURUSAN
2017	Dina	Teknik Informatika
2018	Mulya	Sistem Informasi
2019	Diana	Teknik Informatika

Paramadina

b. Tekan tombol Cari dan aplikasi akan menampilkan hasil dari nama sesuai dengan pencarian yang telah diinputkan.

NOMHS	NAMA	JURUSAN
2017	Dina	Teknik Informatika
2019	Diana	Teknik Informatika

Paramadina