

p

PRAKTIKUM CRUD JAVA

TIARA KUSUMA DEWI

DAFTAR ISI

Persiapan	2
Membuat Database.....	2
Membuat Table	3
Membuat Project di NetBeans	3
Koneksi ke Database	5
Membuat Form Siswa	6
Menampilkan Data ke Dalam Tabel	9
Membuat Fungsi Simpan	10
Persiapan Untuk Membuat Fungsi Update	11
Membuat Fungsi Reset.....	12
Membuat Fungsi Update	13
Membuat Fungsi Delete	13

Persiapan

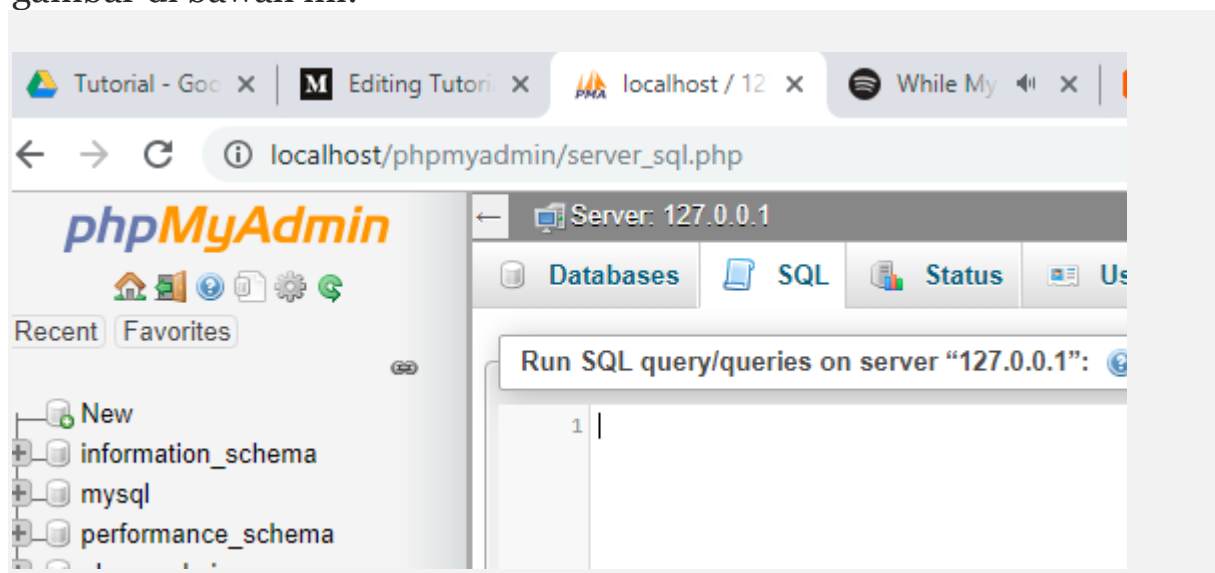
Tutorial kali ini kita akan buat aplikasi menggunakan Java dan MySQL dengan studi kasus input siswa.

Yang diperlukan:

1. Netbeans IDE
2. JDK ≥ 8
3. XAMPP

Membuat Database

Sebelum membuat database pastikan terlebih dahulu XAMPP kalian telah dibuka dan telah menjalankan server Apache dan MySql. Setelah itu kalian buka Localhost/phpmyadmin pada browser kalian seperti gambar di bawah ini:



Lalu klik pada tab **SQL**, kemudian masukkan syntaks

```
CREATE DATABASE tutorialjava;
```

syntax di atas berfungsi untuk membuat database baru dengan nama tutorialjava, **nama database tersebut dapat kalian ubah sesuai kebutuhan.**

Membuat Table

Setelah membuat database saatnya membuat table, pertama kalian harus select database yg telah kalian buat terlebih dahulu. setelah itu kalian tuliskan syntaknya (caranya sama seperti membuat database) hanya saja syntaknya yang berbeda, disini kita akan membuat sebuah table dengan nama **students** dengan isinya **nis, nama, jurusan, jk,**

alamat seperti dibawah ini:

```
CREATE TABLE students(  
  nis int PRIMARY KEY,  
  nama varchar(100),  
  jurusan varchar(100),  
  jk varchar(16),  
  alamat text  
);
```

Membuat Project di NetBeans

Pertama kalian harus membuka aplikasi Netbeans nya, lalu klik file -> new project. Setelah itu pilih kaegori Java dan pilih tipe project Java-nya Java Aplication lalu klik next:

Steps

1. Choose Project
2. ...

Choose Project

Filter:

Categories:

- Java
- JavaFX
- HTML5/JavaScript
- Maven
- PHP
- NetBeans Modules
- Samples

Projects:

- Java Application
- Java Frontend Application
- Java Class Library
- Java Project with Existing Sources
- Java Modular Project
- Java Free-Form Project

Description:

Creates a new Java SE application in a standard IDE project. You can also generate a main class in the project. Standard projects use **an IDE-generated Ant build script** to build, run, and debug your project.

< Back

Next >

Finish

Cancel

Help

Name and Location

Project Name: Java CRUD

Project Location: C:\Users\ariqh\Documents\NetBeansProjects

Browse...

Project Folder: C:\Users\ariqh\Documents\NetBeansProjects\Java CRUD

☐ Use Dedicated Folder for Storing Libraries

Libraries Folder:

Browse...

Different users and projects can share the same compilation libraries (see Help for details).

☐ Create Main Class java.crud.JavaCRUD

< Back

Next >

Finish

Cancel

Help

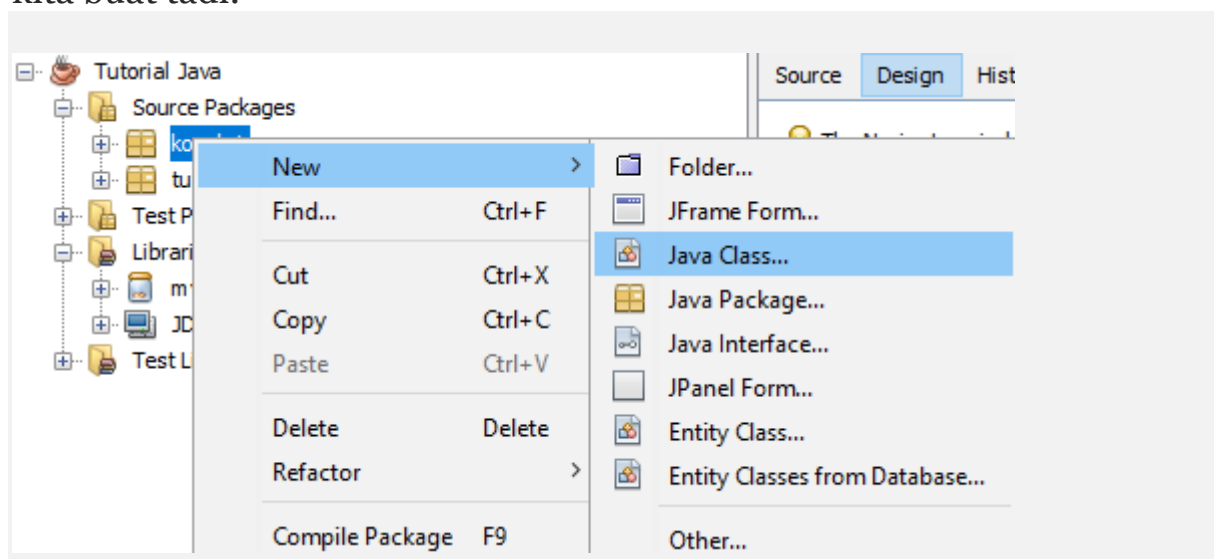
Beri nama dan **jangan ceklis pada Create Main Class**. Karena kita belum memerlukan sebuah Main Class. Setelah itu klik tombol finish, sekarang kita telah membuat project Java menggunakan Netbeans IDE.

Koneksi ke Database

Setelah membuat project, saatnya membuat koneksi ke database db_crud yang telah kita buat sebelumnya. Jadi yang pertama kali kita lakukan adalah menambah driver MySql ke dalam project yang kita buat. Caranya klik kanan bagian **Libraries -> Add Library, pilih MySql JDBC Driver lalu klik add library**.

Kemudian sekarang saatnya membuat sebuah Class yang digunakan untuk melakukan koneksi ke database MySql. Tapi sebelum kita membuat sebuah Class, pastikan kita membuat Package dulu, caranya klik kanan bagian Source Project yang telah kita buat lalu pilih **New -> Java Package** dan beri nama **“koneksi”**.

Setelah itu buat sebuah Class dengan mengklik kanan package yg telah kita buat tadi:



Beri nama **Koneksi** **Koneksi.java**

```
package koneksi;

import java.sql.Connection;
import java.sql.DriverManager;
import javax.swing.JOptionPane;

public class Koneksi {
    Connection koneksi;

    public static Connection Koneksi() {
        try {
            Class.forName("com.mysql.jdbc.Driver");
            Connection koneksi =
DriverManager.getConnection("jdbc:mysql://localhost/tutorialja
va", "root", "");

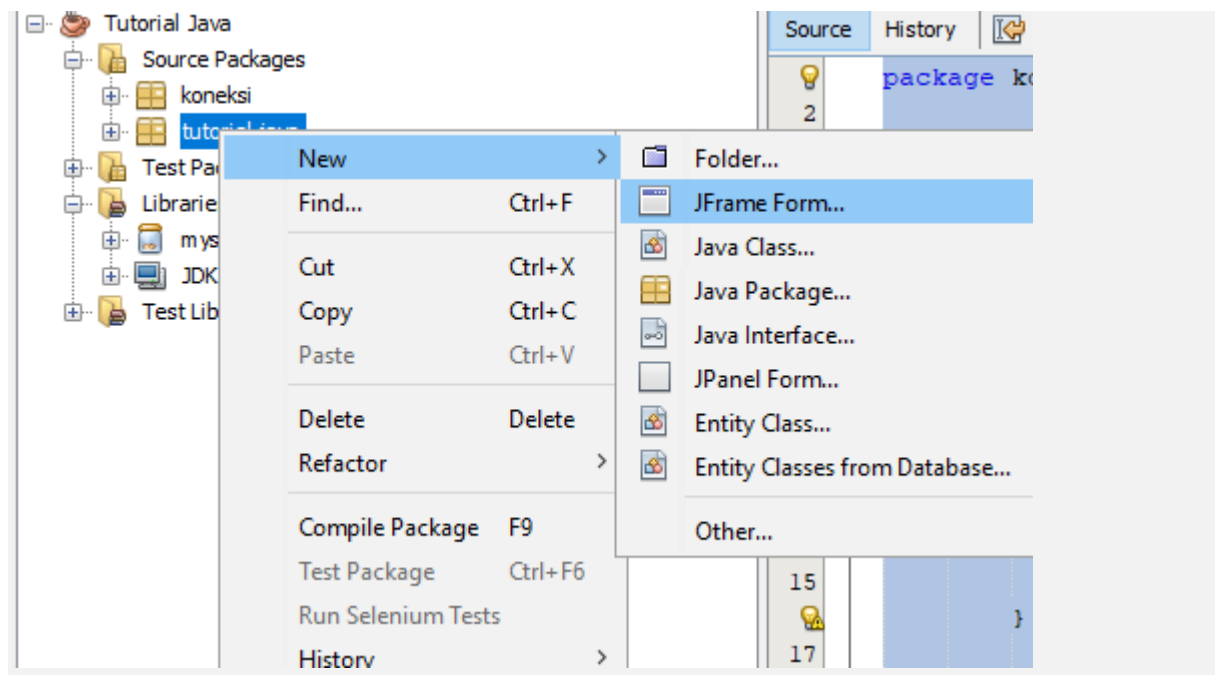
            return koneksi;
        } catch (Exception e) {
            JOptionPane.showMessageDialog(null, e);

            return null;
        }
    }
}
```

Pada line 13 **tutorialjava** adalah nama database yang tadi kita buat, sesuaikan dengan nama database yang kalian buat.

Membuat Form Siswa

yang pertama kalian lakukan adalah membuat package baru dengan nama view, dan klik kanan pada package tersebut new -> JFrame Form dan beri nama **FormSiswa**:



Sekarang saatnya membuat tampilan dari form nya, kita akan menambahkan sebuah label, textfield, textarea, button, comboBox dan table:

Form Siswa

NIS

Nama

Jurusan

Rekayasa Perangkat Lunak

JK

Laki-laki

Alamat

Simpan

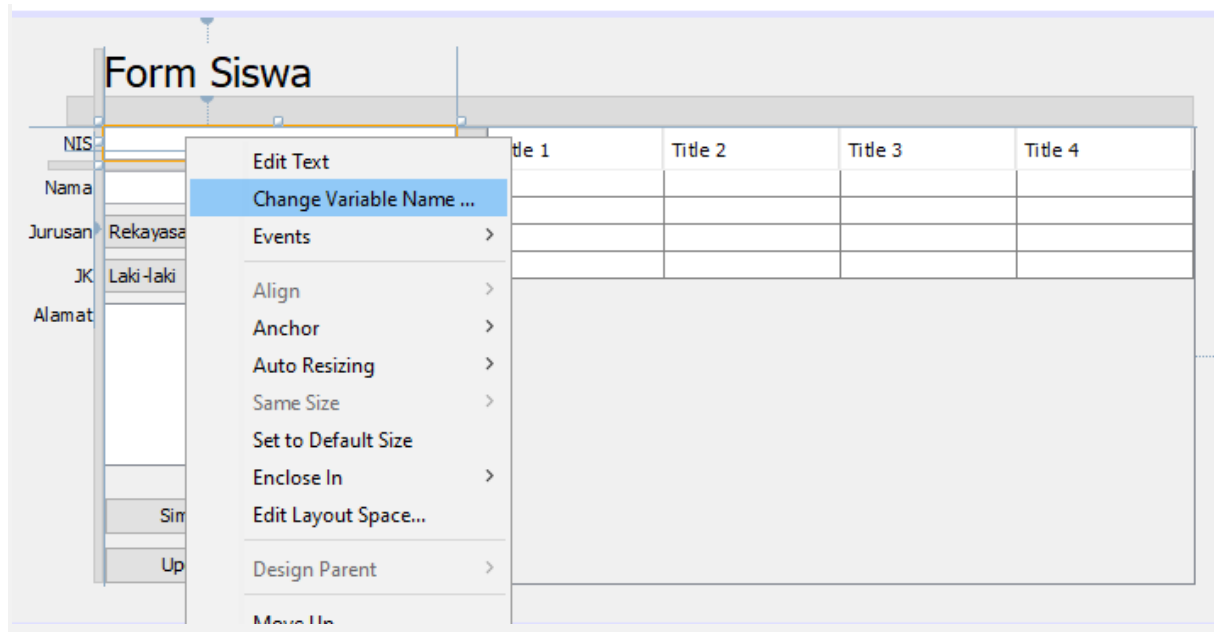
Hapus

Update

Reset

Title 1	Title 2	Title 3	Title 4

Dan setelah itu kalian ganti nama varibelnya dengan cara klik kanan lalu **change variable name**:



Ubah nama variable nya seperti gambar di bawah ini:

Kemudian klik kanan pada **inputJurusan** dan **inputJK** -> klik properties -> klik pada tombol dengan “titik tiga” disebelah kanan model kemudian masukkan jurusan yang akan di-input user.

FormSiswa.java

Klik pada tab **source** dibawah tab file FormSiswa.java

masukkan syntax berikut dibawah **package tutorial.java;**

```
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.Statement;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;
```

Kemudian deklarasikan variable seperti dibawah ini

```
public class FormSiswa extends javax.swing.JFrame {
    public Statement st;
    public ResultSet rs;
    public DefaultTableModel tabModel;
    Connection cn = koneksi.Koneksi.Koneksi();.....
```

Menampilkan Data ke Dalam Tabel

```
public void judul() {
    Object[] judul = {
        "NIS", "Nama", "Jurusan", "JK", "Alamat"
    };
    tabModel = new DefaultTableModel(null, judul);
    tableSiswa.setModel(tabModel);
}

public void tampilData(String where) {
    try {
        st = cn.createStatement();
        tabModel.getDataVector().removeAllElements();
        tabModel.fireTableDataChanged();
        rs = st.executeQuery("SELECT * FROM students " + where);

        while (rs.next()) {
            Object[] data = {
                rs.getString("nis"),
                rs.getString("nama"),
                rs.getString("jurusan"),
                rs.getString("jk"),
                rs.getString("alamat"),
            };

            tabModel.addRow(data);
        }
    }
}
```

```

    }
} catch (Exception e) {
    e.printStackTrace();
}
}

```

Deklarasikan class **tampilData()** & **judul()** ke dalam class FormSiswa. Contohnya seperti ini:

```

public FormSiswa() {
    initComponents();
    judul();
    tampilData("");

    buttonHapus.setEnabled(false);
    buttonUpdate.setEnabled(false);
}

```

Membuat Fungsi Simpan

Pertama double click pada button **buttonSimpan**, lalu masukan kode seperti bawah ini:

```

private void
buttonSimpanActionPerformed(java.awt.event.ActionEvent evt) {
    try {
        st = cn.createStatement();
        st.executeUpdate("INSERT INTO students VALUES('" +
inputNis.getText() + "','" +
        + inputNama.getText() + "','" +
        + inputJurusan.getSelectedItem() + "','" +
        + inputJK.getSelectedItem() + "','" +
        + inputAlamat.getText() + "')");
        tampilData("");
        JOptionPane.showMessageDialog(null, "Simpan Berhasil");
        inputNis.setText("");
        inputNama.setText("");
        inputAlamat.setText("");
        inputJurusan.setSelectedItem("");
    } catch (Exception e) {
        e.printStackTrace();
    }
}

```

Silahkan run project kalian untuk mencoba fungsi simpan **Shift + F6**.
Jika berhasil akan muncul window message “Simpan Berhasil”.

Persiapan Untuk Membuat Fungsi Update

Pada bagian ini kita akan membuat aplikasi kita pada saat klik datanya, maka data tersebut akan berpindah ke bagian textfield

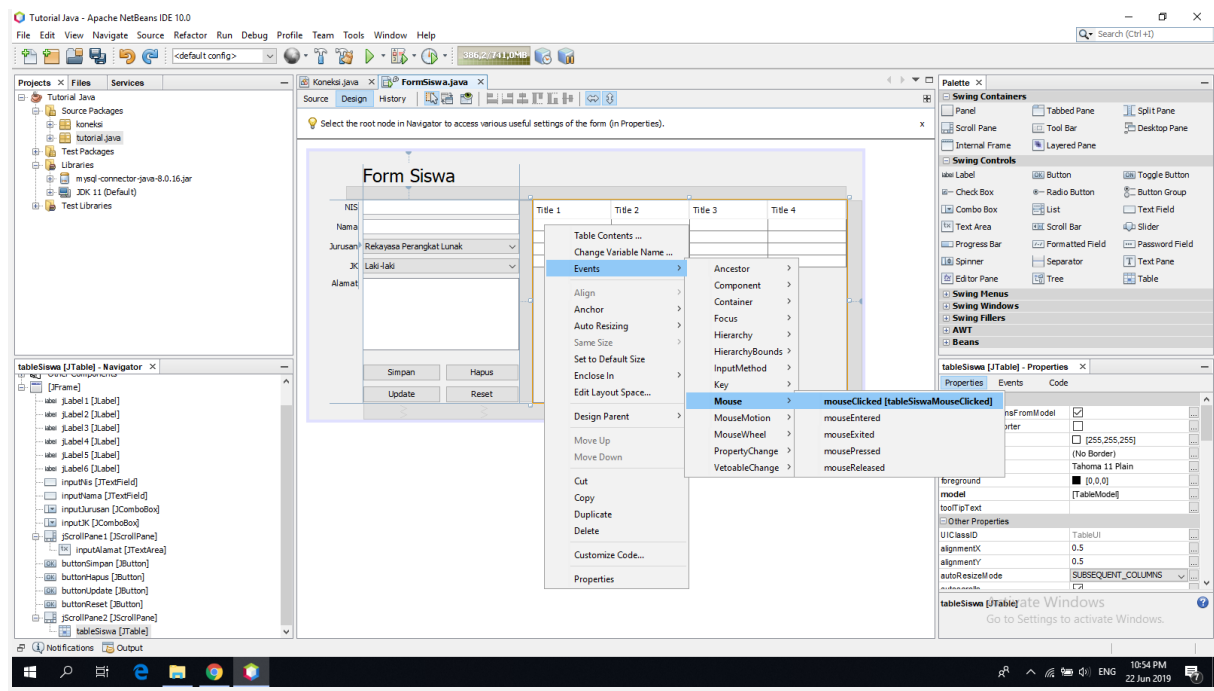
Form Siswa

NIS: 11705978
Nama: Ariq Heritsa
Jurusan: Rekayasa Perangkat Lunak
JK: Laki-laki
Alamat: Jl. Jalan Ke Kota Bogor

Simpan Hapus
Update Reset

NIS	Nama	Jurusan	JK	Alamat
11705978	Ariq Heritsa	rangkat Lunak	Laki-laki	Jl. Jalan Ke...

klik kanan pada table -> Events -> Mouse -> MouseClicked



Masukkan kode berikut

```
private void tableSiswaMouseClicked(java.awt.event.MouseEvent
evt) {

inputNis.setText(tableSiswa.getValueAt(tableSiswa.getSelectedR
ow(), 0).toString());

inputNama.setText(tableSiswa.getValueAt(tableSiswa.getSelected
Row(), 1).toString());

inputJurusan.setSelectedItem(tableSiswa.getValueAt(tableSiswa.
getSelectedRow(), 2).toString());

inputJK.setSelectedItem(tableSiswa.getValueAt(tableSiswa.getSe
lectedRow(), 3).toString());

inputAlamat.setText(tableSiswa.getValueAt(tableSiswa.getSelect
edRow(), 4).toString());
    buttonSimpan.setEnabled(false);
    buttonUpdate.setEnabled(true);
    buttonHapus.setEnabled(true);
}
```

Membuat Fungsi Reset

Fungsi ini adalah untuk mengosongkan semua textfield

```
public void reset() {
    inputNis.setText("");
    inputNama.setText("");
    inputJurusan.setSelectedItem("");
    inputJK.setSelectedItem("");
    inputAlamat.setText("");
}
```

Panggil fungsi tersebut pada **buttonReset**

```
private void buttonResetActionPerformed(java.awt.event.ActionEvent evt) {
    reset();
}
```

Membuat Fungsi Update

```
private void
buttonUpdateActionPerformed(java.awt.event.ActionEvent evt) {
    try {
        st = cn.createStatement();
        st.executeUpdate("UPDATE students set "
            + "nis='" + inputNis.getText() + "', "
            + "nama='" + inputNama.getText() + "', "
            + "jurusan='" + inputJurusan.getSelectedItem() + "', "
            + "jk='" + inputJK.getSelectedItem() + "', "
            + "alamat='" + inputAlamat.getText() + "'");
        tampilData("");
        JOptionPane.showMessageDialog(null, "Update Berhasil");
        reset();
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

Membuat Fungsi Delete

```
private void buttonHapusActionPerformed(java.awt.event.ActionEvent evt) {
```

```
try {
    int jawab;

    if ((jawab = JOptionPane.showConfirmDialog(null, "Ingin menghapus
data?", "konfirmasi", JOptionPane.YES_NO_OPTION)) == 0) {
        st = cn.createStatement();
        st.executeUpdate("DELETE FROM students WHERE nis='"
            + tabModel.getValueAt(tableSiswa.getSelectedRow(), 0) + "'");
        tampilData("");
        reset();
    }
} catch (Exception e) {
    e.printStackTrace();
}
}
```