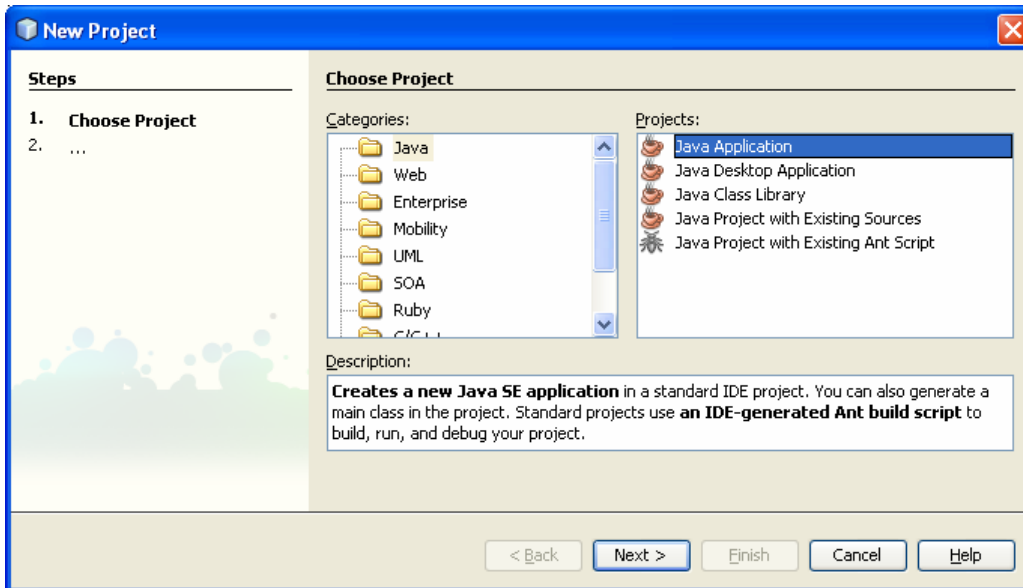


# Menghubungkan Database Access Menggunakan Netbeans

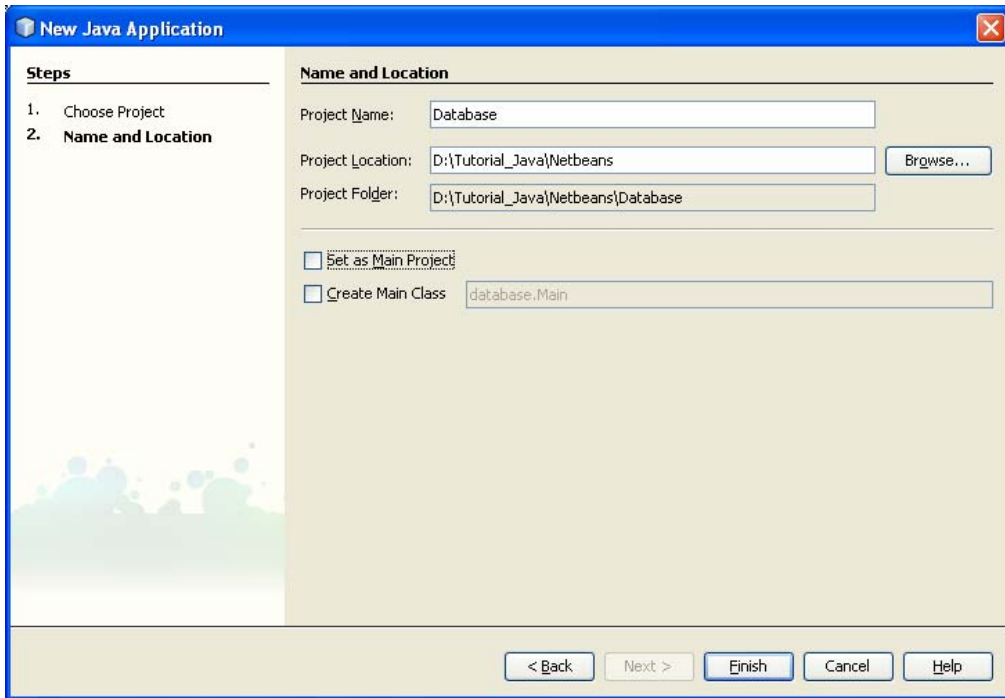
Oleh : Yuliana Setiowati  
(yuliana@eepis-its.edu)

Membuat Projek Baru di Netbeans

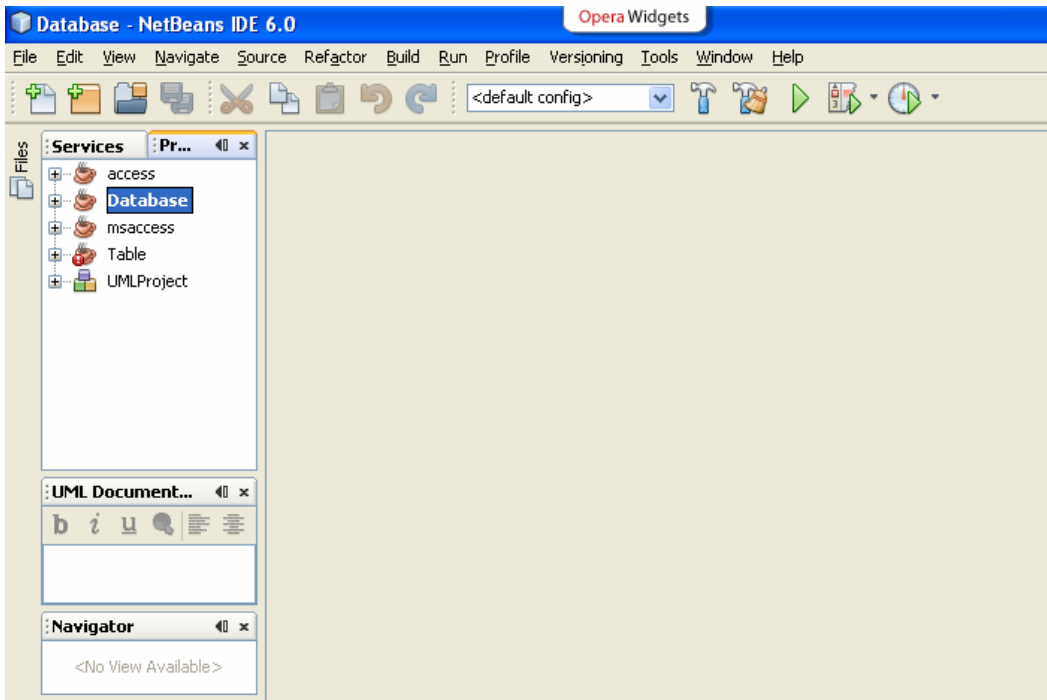
1. Buatlah projek baru di Netbeans pilih File → New Project. Pilih Categories : Java dengan Projects : Java Application ditunjukkan pada gambar 1. Selanjutnya klik Next.  
Isi nama project dengan Database dan pilih lokasi (folder) project yang akan Anda kerjakan (gambar 2). Hilangkan tanda (V) pada check box Set as Main Project dan Create Main Class. Selanjutnya klik finish, hasil ditunjukkan pada gambar 3. Setelah proses ini selesai akan terbentuk direktori dengan nama Database dengan folder nbproject, src, dan test (gambar 4).



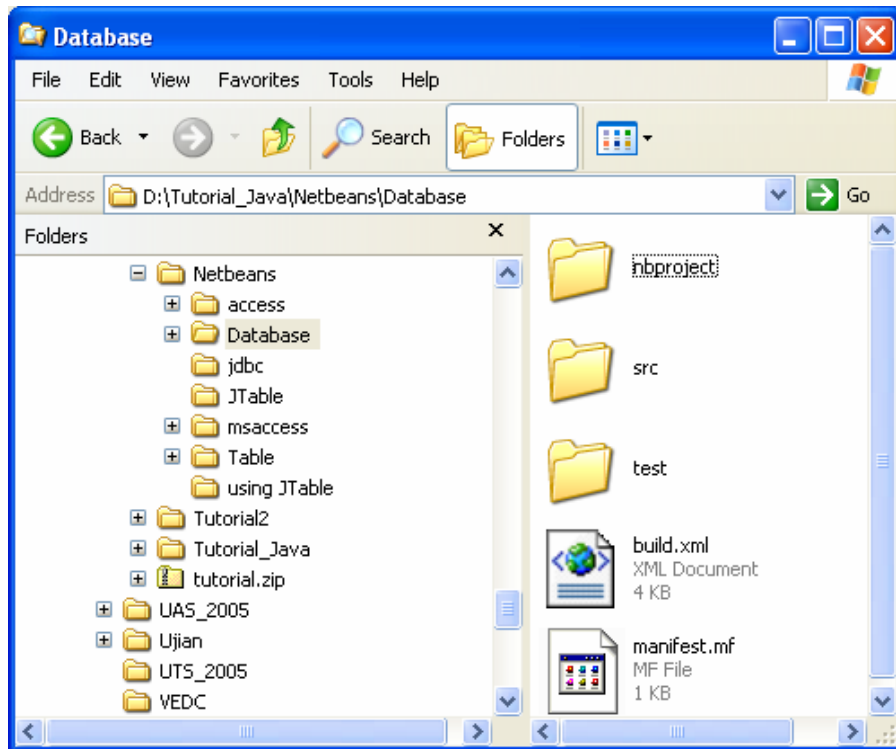
Gambar 1



Gambar 2



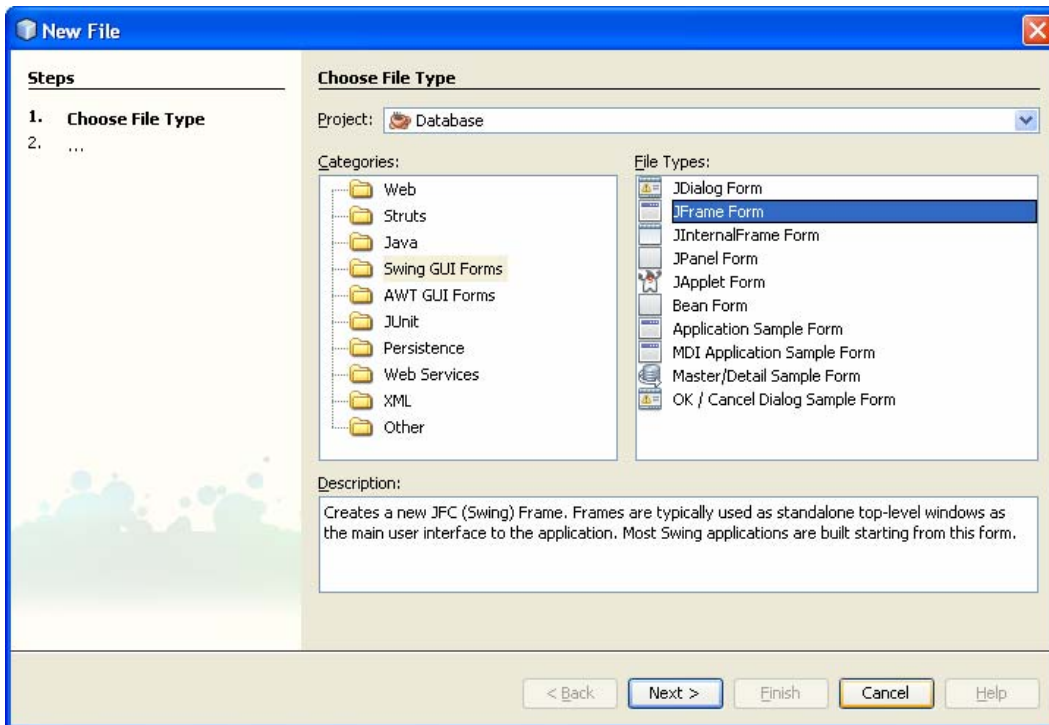
Gambar 3.



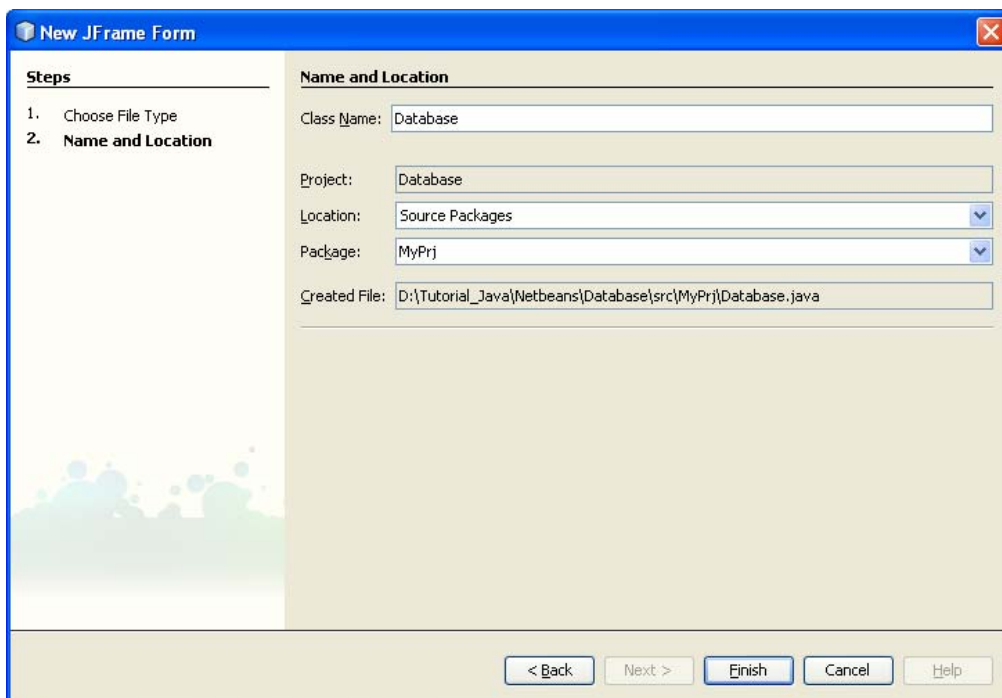
Gambar 4

Membuat form di Netbeans

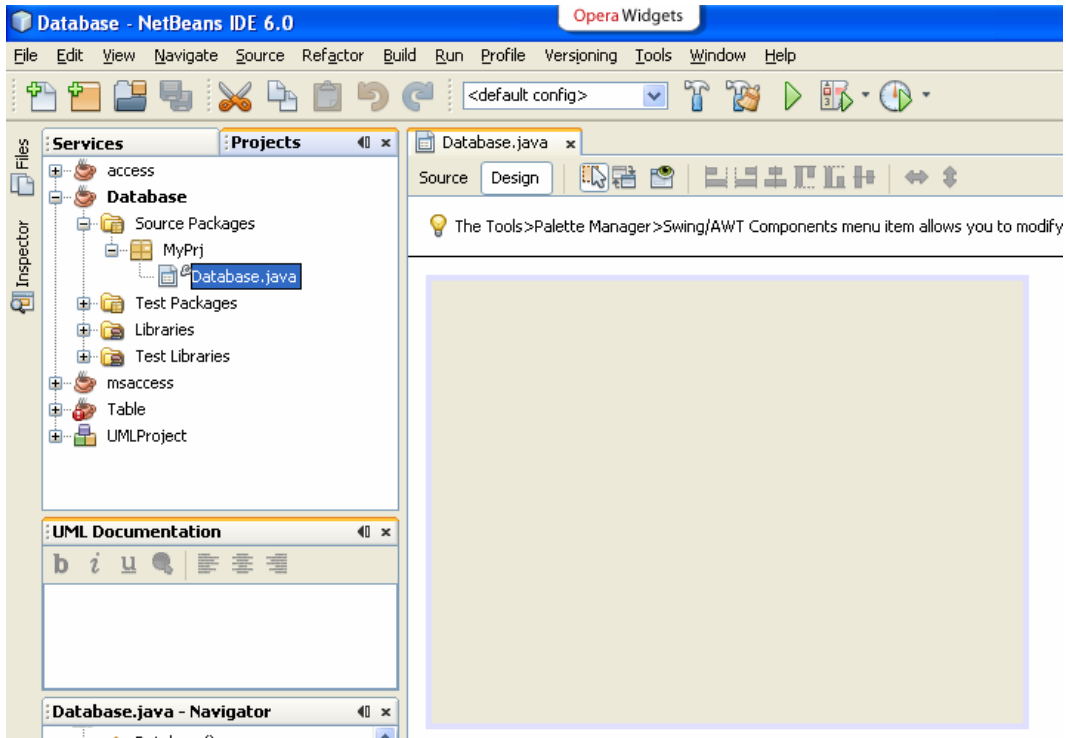
2. Membuat form di Netbeans pilih File→New File dengan Categories : Swing GUI Forms dengan tipe file : JFrame Form selanjutnya tekan next (gambar 5). Buat nama class Database, untuk package boleh diisi atau tidak selanjutnya tekan finish (gambar 6). Jika diisi akan membentuk direktori pada direktori src. Misal kita buat dengan MyPrj maka akan dibuat direktori MyPrj pada direktori src (gambar 7).



Gambar 5



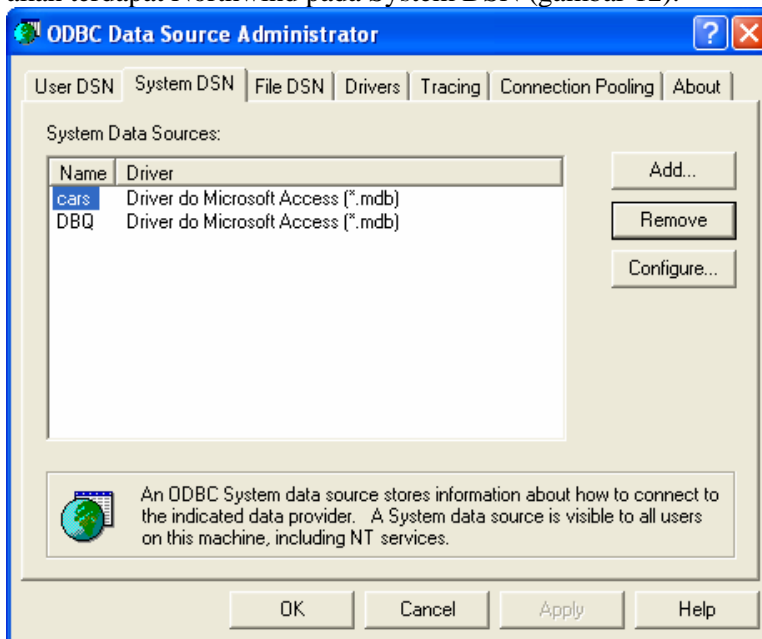
Gambar 6



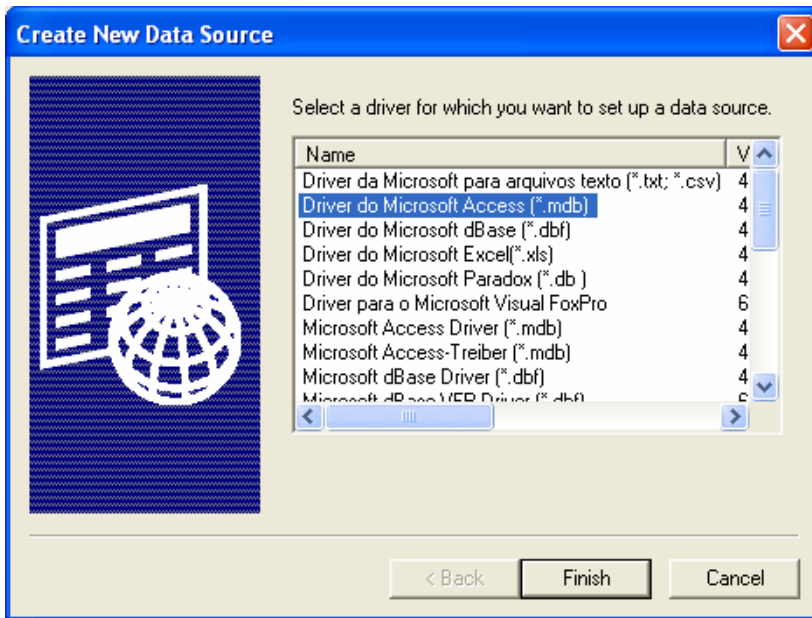
Gambar 7

Mempersiapkan database NorthWind

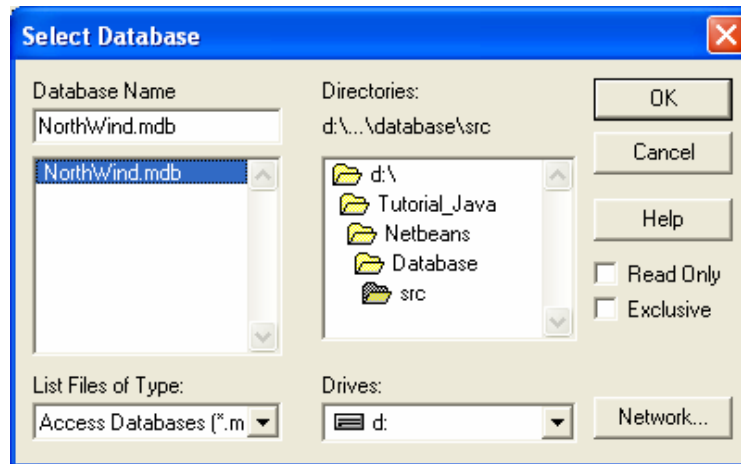
3. Simpan database Northwind pada direktori src. Buat koneksi ODBC. Pilih system DSN (gambar 8). Pilih button Add, maka akan muncul form create new data source pilih driver Microsoft Access, selanjutnya tekan finish (gambar 9). Pada form ODBC Microsoft Access Setup tentukan nama data source dan letak database (gambar 10 dan 11). Setelah proses selesai maka akan terdapat Northwind pada System DSN (gambar 12).



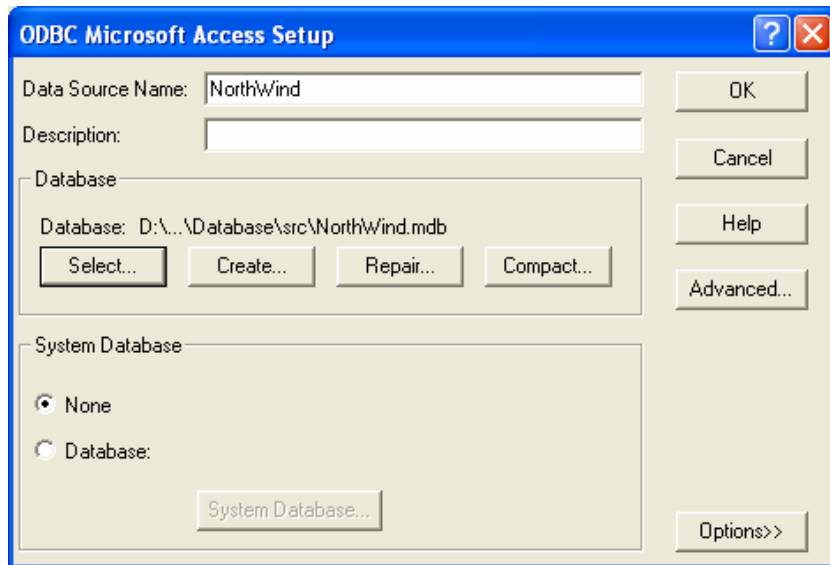
Gambar 8



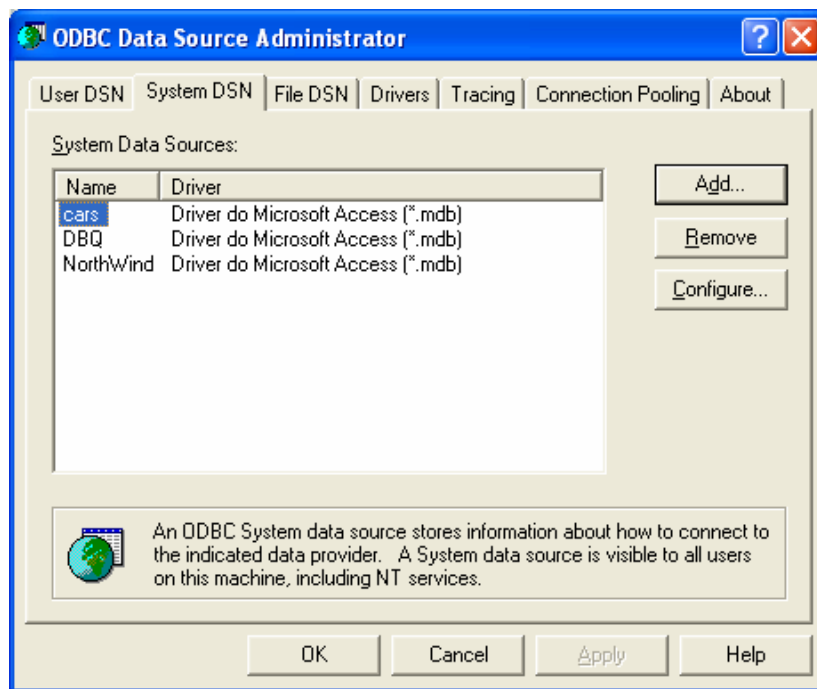
Gambar 9



Gambar 10



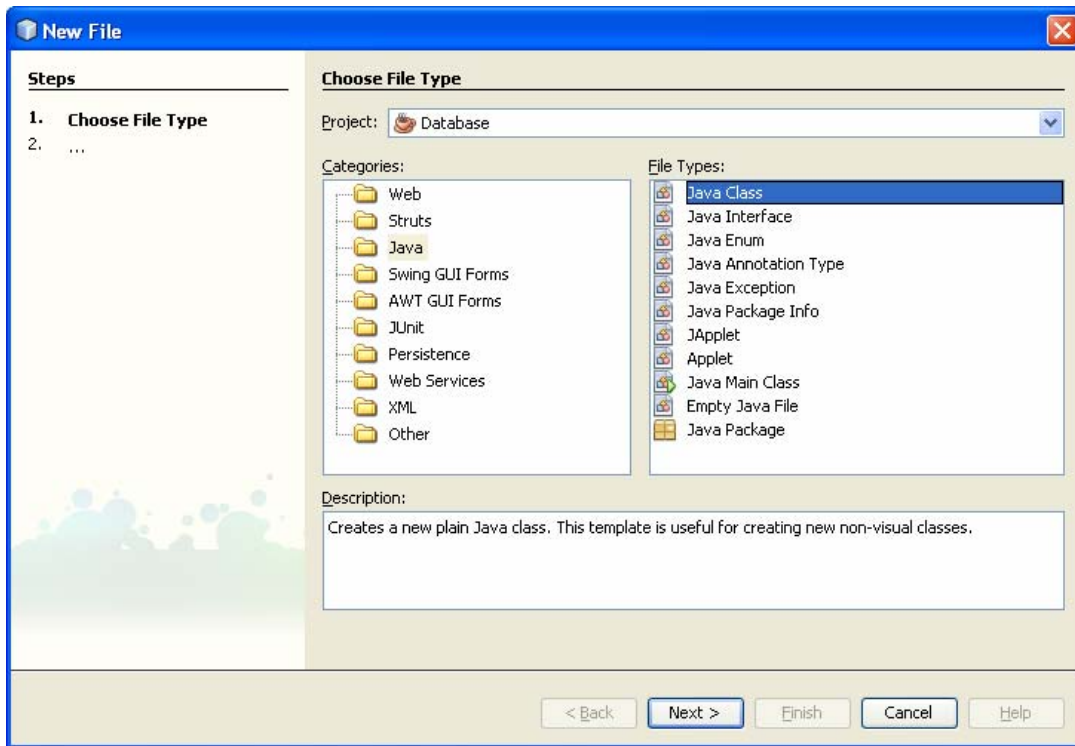
Gambar 11



Gambar 12

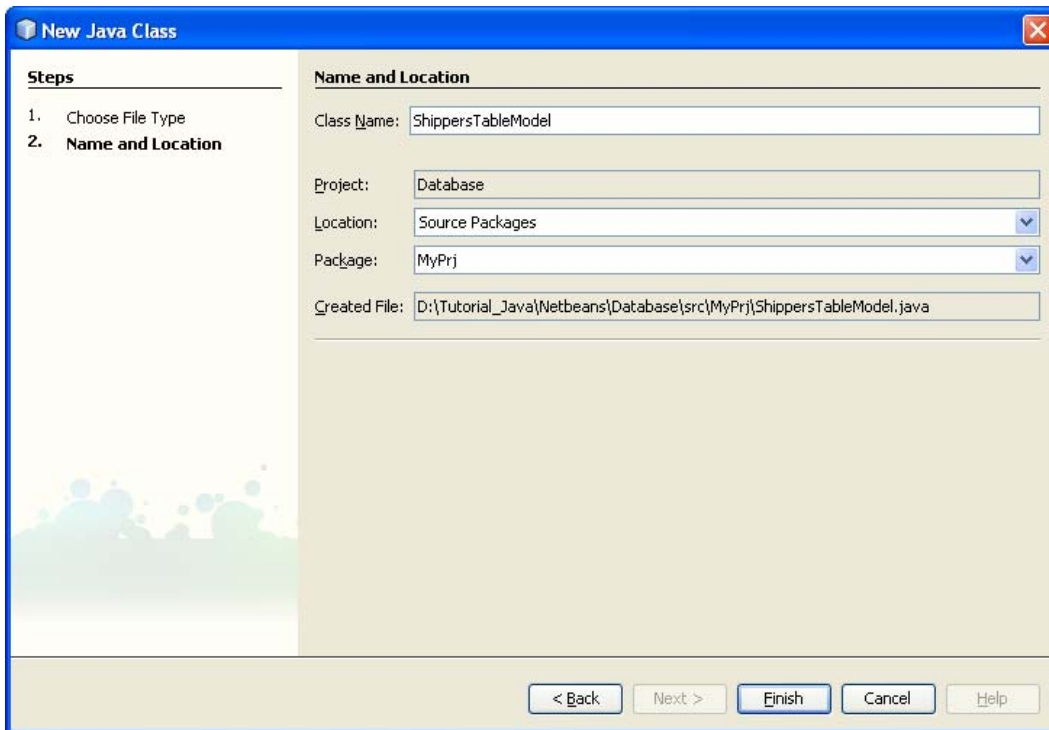
Membuat file class : ShipperTableModel

4. Pilih File → New File, pilih kategori : Java dengan tipe file java class, selanjutnya tekan next (gambar 13). Buat nama class ShipperTableModel pada project yang sudah Anda buat selanjutnya tekan finish (gambar 14). Tambahkan extends AbstractTableModel pada class ShipperTableModel (gambar 15). Maka akan terlihat tulisan AbstractTableModel tercetak miring menunjukkan ada kesalahan. Kesalahan ini disebabkan kita belum melakukan import. Caranya klik kanan pada source editor pilih **fix import**, maka secara otomatis akan mengimport javax.swing.table.AbstractTableModel.

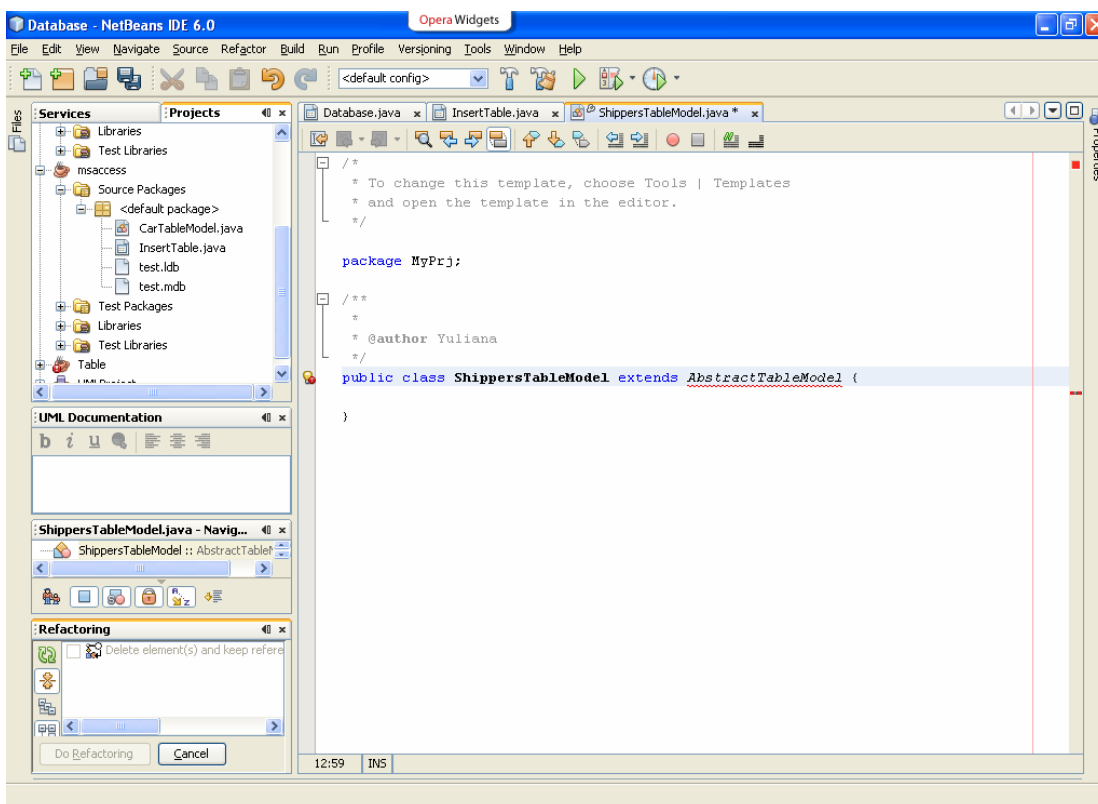


Gambar 13





Gambar 14



Gambar 15

## Class ShippersTableModel

```
package MyPrj;

import java.sql.ResultSet;
import java.util.ArrayList;
import javax.swing.table.AbstractTableModel;

public class ShippersTableModel extends AbstractTableModel{
    private String[] columnNames={"ShipperID","CompanyName","Phone"};
    private Object[][] data = new Object[50][3];

    public ShippersTableModel (ResultSet rs) {
        int brs =0;
        try{
            while(rs.next()){

                data[brs][0] = rs.getString("ShipperID") ;
                data[brs][1] = rs.getString("CompanyName") ;
                data[brs][2] = rs.getString("Phone") ;

                brs++ ;
            }
        } catch(Exception e){
            System.out.println("Exception in CarTableModel");
        }
    }

    public Object getValueAt(int rowindex, int columnindex) {
        return data[rowindex][columnindex];
    }

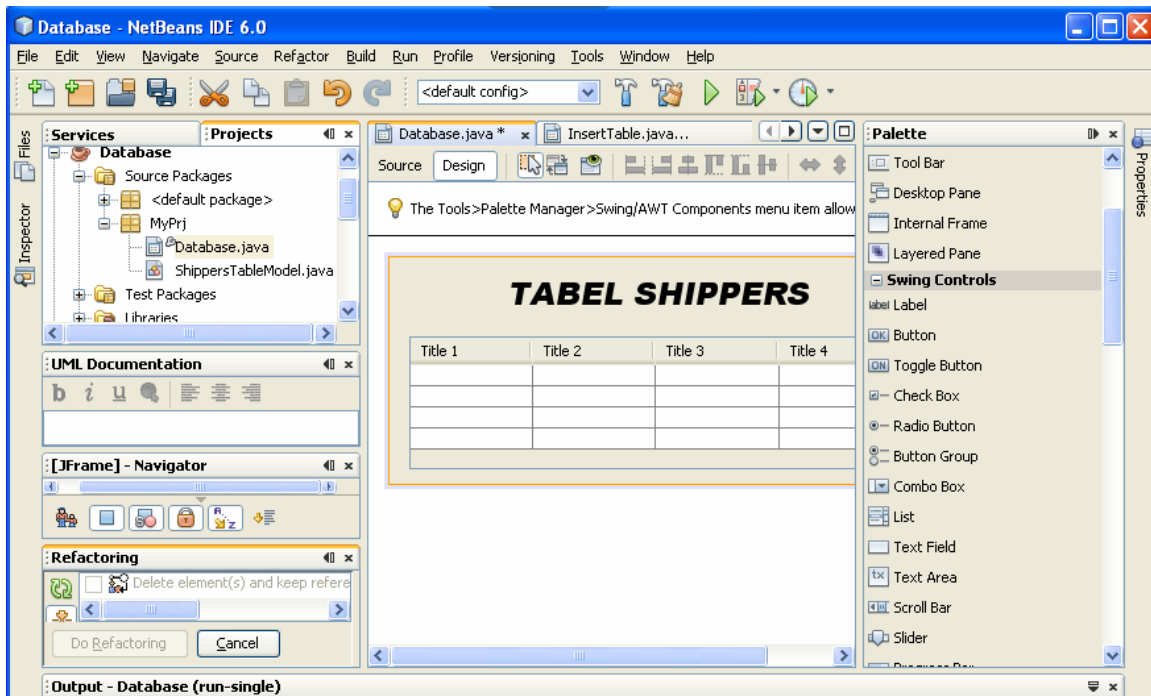
    public int getRowCount() {
        return data.length ;
    }

    public int getColumnCount() {
        return columnNames.length ;
    }

    public String getColumnName(int param) {
        return columnNames[param];
    }
}
```

Membuat file Database.java

5. Buat form seperti gambar 16. Form terdiri dari dua komponen yaitu JLabel dan JTable. Untuk mengubah label dan nama variabel, arahkan kursor pada komponen tersebut dan klik kanan. Untuk mengubah font pada label klik kanan pilih propertis, pada propertis pilih font.



Gambar 16

Berikut ini tambahkan pada file Database. File dengan tulisan tebal merupakan bawaan dari Netbeans. Pada konstruktor terdapat beberapa fungsi yang mempunyai kegunaan sbb :

Koneksi(): untuk menghubungkan jdbc dengan odbc.

initComponents(): inialisasi komponen pada aplikasi

getResultFromShippers() : untuk mendapatkan data pada tabel Shippers

shippersTable.setModel(new ShippersTableModel(rs)) : untuk mengisi shippersTable

Output program ditunjukkan pada gambar 17.

```
package MyPrj;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class Database extends javax.swing.JFrame {
    private Connection con ;
    private Statement st;

    /** Creates new form Database */
    public Database() {
        initComponents();
        Koneksi();
        ResultSet rs = getResultFromShippers();
        shippersTable.setModel(new ShippersTableModel(rs));
    }

    public void Koneksi(){
        String dbname = "jdbc:odbc:NorthWind";
        try{
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
            con = DriverManager.getConnection(dbname, "", "");
        }
    }
}
```

```

    }
    catch(ClassNotFoundException ex) {
        System.err.println("Driver Error");
        ex.printStackTrace();
        System.exit(1);
    }
}

    catch(SQLException e){
        System.out.println("Tidak berhasil koneksi");
    }
}

public ResultSet getResultFromShippers(){

    ResultSet rs=null;

    try {
        st=con.createStatement();
        rs=st.executeQuery("SELECT * FROM Shippers");
    }
    catch(SQLException ex){
        ex.printStackTrace();
    }
    return(rs);
}

}

public void dbClose(){
    try {
        con.close();
    }
    catch(SQLException sqlx){
        System.err.println("Error :Koneksi Database tidak Bisa diputus");
    }
}
}
}

```



The screenshot shows a window with a blue title bar and standard Windows-style window controls (minimize, maximize, close). The window content has a light gray background with the title "TABEL SHIPPERS" in bold black text. Below the title is a table with three columns: "ShipperID", "CompanyName", and "Phone". The table contains five rows of data.

ShipperID	CompanyName	Phone
1	TIKI	511111
2	United Package	522222
3	Federal Shipping	533333
4	Federal Express	(031) 555-1111
5	POS	(031) 555-1111

Gambar 17

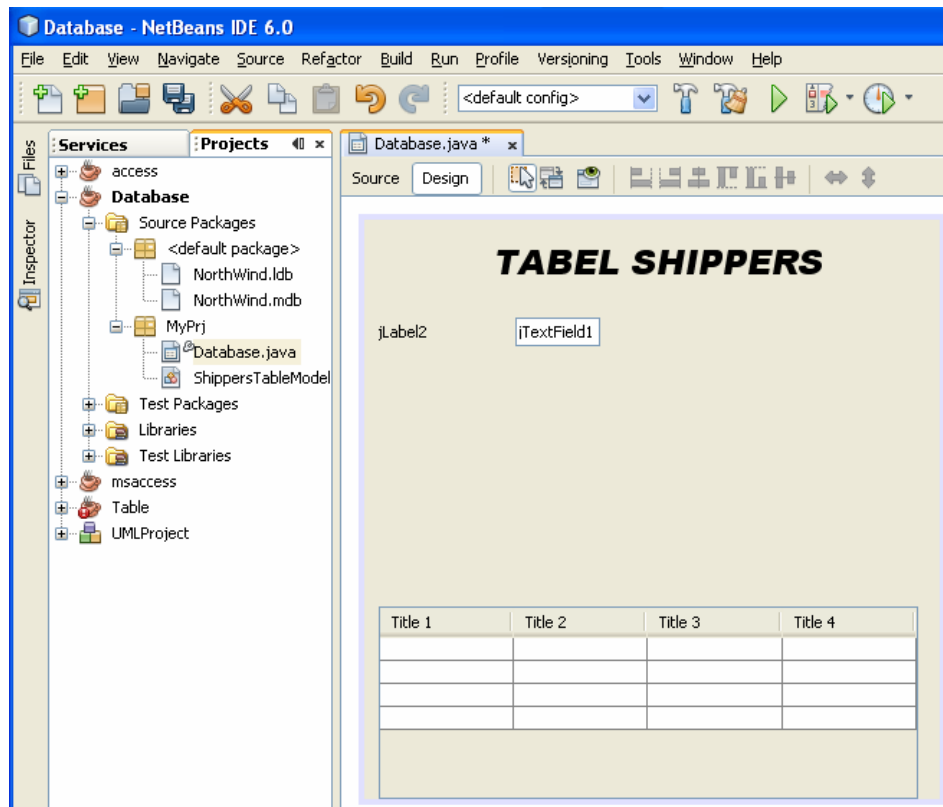
Kita kembangkan lagi aplikasi yang sudah kita buat. Pengembangan dari aplikasi adalah dapat menyisipkan data baru ke table Shippers.

6. Ambil JPanel dari pallete (dengan nama jPanel1) ke dalam aplikasi. Letakkan JLabel dan JTextField ke dalam JPanel (gambar 18). Pada panel lakukan klik kanan pilih setLayout GridLayout (gambar 19). Klik Inspector , klik tanda + pada jPanel1 maka terdapat GridLayout, kemudian lakukan klik kanan pilih properties. Isi kolom dengan 2 dan baris dengan 3 (gambar 20). Tambahkan JLabel dan JTextField masing-masing 2 dan ubah

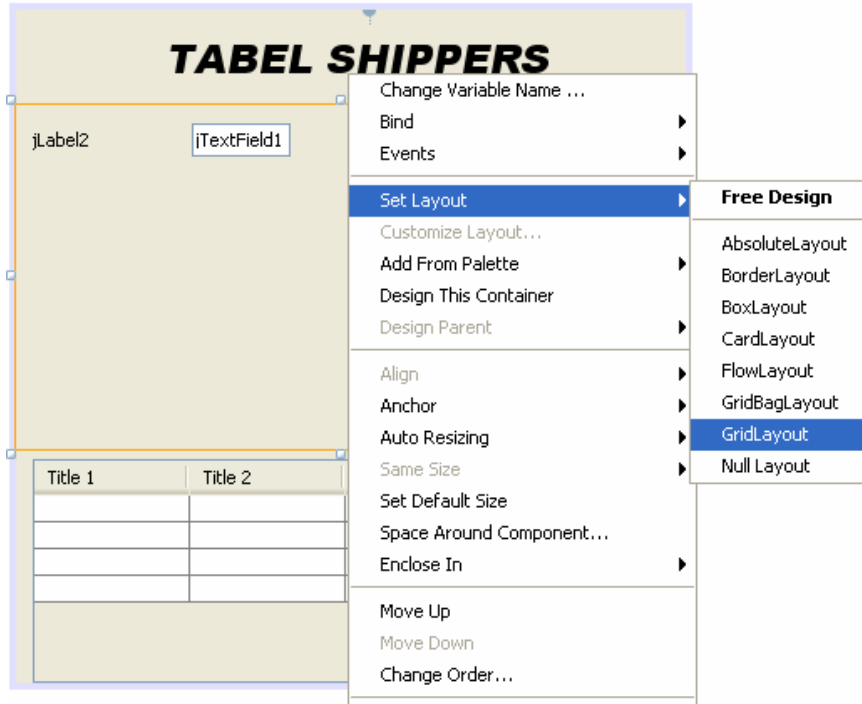
namanya seperti pada tabel 1. Tambahkan pula JButton ubah text dengan send dan nama button dengan sendButton (gambar 21 dan 22).

Label	JTextField
Text : ShipperID Name : shipperIDLabel	Text : (kosong) Name : shipperIDField
Text : CompanyName Name : companyNameLabel	Name : companyNameField
Text : Phone Name : phoneLabel	Name : phoneField

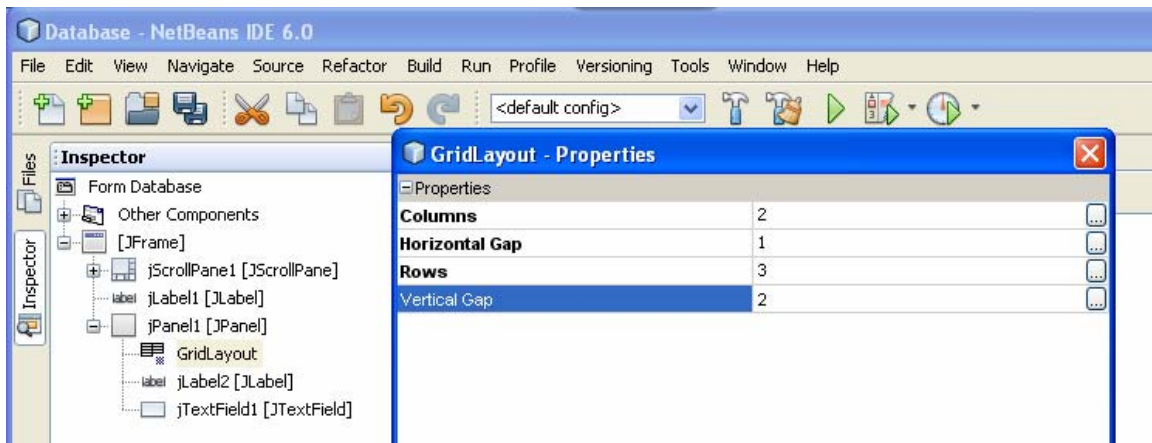
Tabel 1



Gambar 18

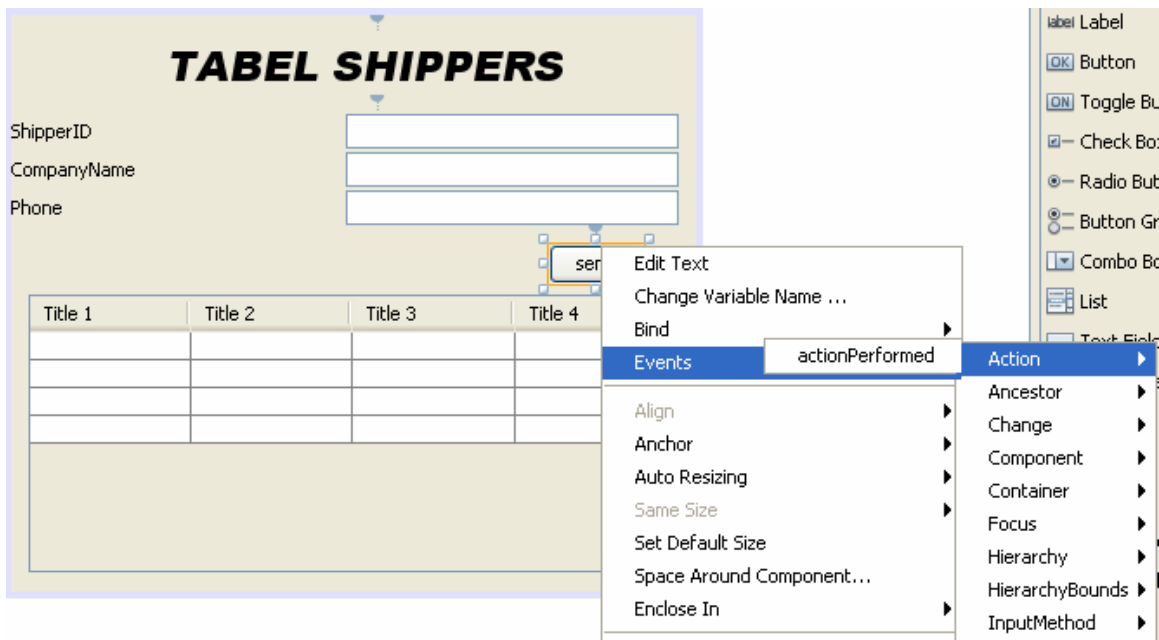


Gambar 19



Gambar 20





Gambar 23

7. Isi fungsi `sendButtonActionPerformed` dengan program seperti yang ada di bawah ini. Tambahkan pula fungsi `quotate`.

```
private void sendButtonActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String id=shipperIDField.getText();
    String name=companyNameField.getText();
    String ph=phoneField.getText();
    String insertStr="";

    try{
        st = con.createStatement();
        insertStr="insert into Shippers (ShipperID, CompanyName, Phone)values("
                +quotate(id)+", "
                +quotate(name)+", "
                +quotate(ph)
                +" )";

        int done=st.executeUpdate(insertStr);

        //Penting : untuk menampilkan pada shippersTable
        ResultSet rs = getResultFromShippers();
        shippersTable.setModel(new ShippersTableModel2(rs));

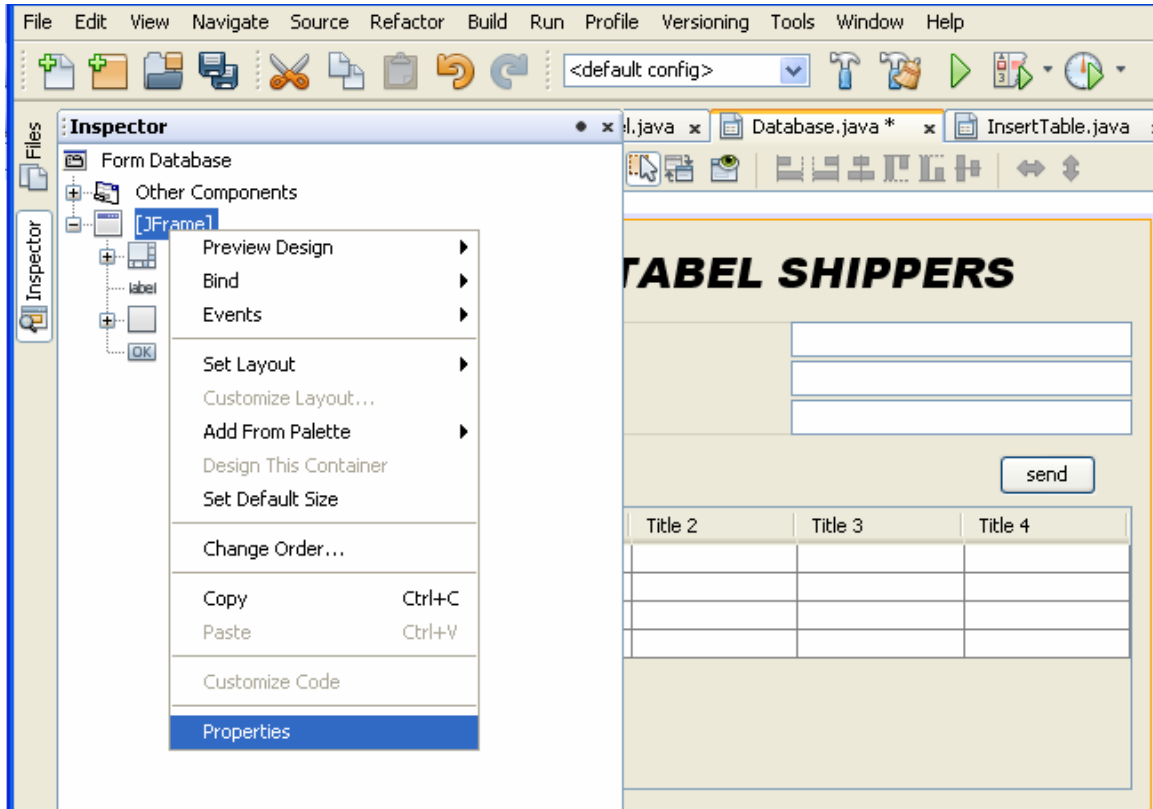
    }
    catch(Exception e){
        e.printStackTrace();
    }
}

public String quotate(String content){
    return "'" +content+"'";
}
}
```

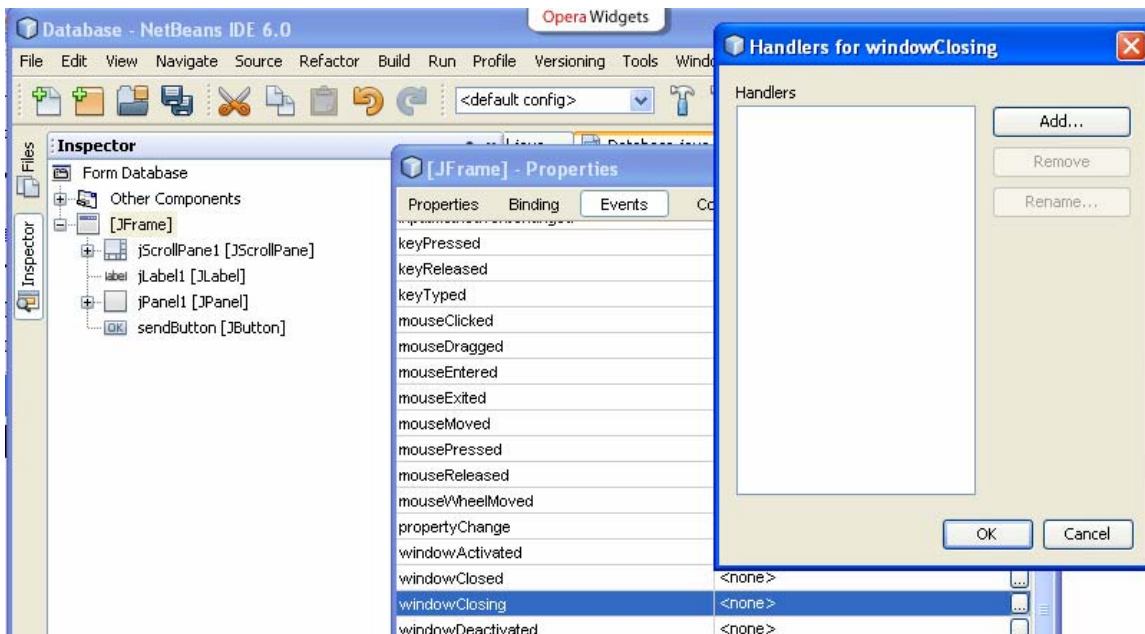
Membuat event `windowClosing` pada form



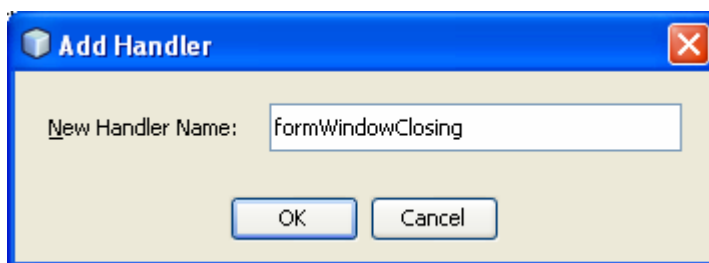
8. Pilih Inspector, klik pada JFrame klik kanan pilih propertis (gambar 24). Pilih events untuk window closing (gambar 25) dan beri nama dengan formWindowClosing (gambar 26). Hasil seperti gambar 27. Tambahkan fungsi dbClose yang sudah Anda kerjakan sebelumnya. Lakukan pengisian pada ShipperID, CompanyName dan Phone (gambar 28) , hasil ditunjukkan pada gambar 29.



Gambar 24



Gambar 25



Gambar 26

```
private void formWindowClosing(java.awt.event.WindowEvent evt) {
    // TODO add your handling code here:
    dbClose();
}
```

Gambar 27

**TABEL SHIPPERS**

ShipperID:

CompanyName:

Phone:

ShipperID	CompanyName	Phone
1	TIKI	511111
2	United Package	522222
3	Federal Shipping	533333
4	Federal Express	(031) 555-1111
5	POS	(031) 555-1111

Gambar 28

**TABEL SHIPPERS**

ShipperID:

CompanyName:

Phone:

ShipperID	CompanyName	Phone
1	TIKI	511111
2	United Package	522222
3	Federal Shipping	533333
4	Federal Express	(031) 555-1111
5	POS	(031) 555-1111
6	Test	3111111

Gambar 29