

Nama : Ade Reiki Karuna

NIM : A12.2020.06455

Mata Kuliah : PBO

Tugas Praktik 10

(FBarang.java)

Source Code :

```
import java.awt.*;
import java.awt.event.*;
import java.io.*;
import javax.swing.*;
import java.sql.*;
import javax.swing.table.DefaultTableModel;

public class FBarang extends JFrame implements ActionListener,
MouseListener
{
    private JPanel pBrg      = new JPanel();
    private JLabel lKode      = new JLabel ("Kode          "),
              lNama          = new JLabel ("Nama Barang "),
              lHrgBeli       = new JLabel ("Harga Beli  "),
              lHrgJual        = new JLabel ("Harga Jual   "),
              lJumlah         = new JLabel ("Jumlah       "),
              lCreate         = new JLabel ("Pembuat      ");

    //mengambil input dari baris teks
    private JTextField fKode    = new JTextField (),
              fNama           = new JTextField (),
              fHrgBeli        = new JTextField (),
              fHrgJual         = new JTextField (),
              fJumlah          = new JTextField ();

    //menambah tombol
    private JButton btnAdd      = new JButton (),
              btnCari          = new JButton (),
              btnKoreksi        = new JButton (),
              btnHapus          = new JButton (),
              btnSelesai        = new JButton ();

    private ResultSet rs;
    String[] header = {"Kode", "Nama", "Harga Beli", "Harga Jual",
"Jumlah"};
    Integer[] lebaklm = {75,200,100,100,100};
```

```

DefaultTableModel tabMode01;
JTable tabel = new JTable();
JScrollPane skrTabel = new JScrollPane();

public FBarang() {
    setPreferredSize(new Dimension(800,500));
    setTitle("Data Barang");
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    new BorderLayout();
    JDesktopPane pBrg = new JDesktopPane();
    add(pBrg, BorderLayout.CENTER);

    /* Mengatur letak objek Label Di container*/
    lKode.setBounds (20, 20, 100, 25);
    lNama.setBounds (20, 55, 100, 25);
    lHrgBeli.setBounds (20, 90, 100, 25);
    lHrgJual.setBounds (20, 125, 100, 25);
    lJumlah.setBounds (20, 155, 100, 25);

    lCreate.setBounds (600, 20, 250, 25);
    lCreate.setText("Nama : Ade Reiki Karuna 06455");
    lCreate.setFont(new Font("Verdana", Font.PLAIN, 15));
    lCreate.setForeground(new Color(220, 10, 10));

    /* Mengatur letak objek Text Di Container */
    fKode.setBounds (115, 20, 100, 25);
    fNama.setBounds (115, 55, 205, 25);
    fHrgBeli.setBounds (115, 90, 92, 25);
    fHrgJual.setBounds (115, 125, 92, 25);
    fJumlah.setBounds (115, 155,100,25);

    /* Mengatur letak objek Button di Container */
    btnAdd.setBounds (260, 350, 85, 25);
    btnAdd.setLabel("Add");
    btnAdd.setBackground(Color.green);
    btnAdd.setForeground(Color.getHSBColor(250,0,255));
    btnCari.setBounds (360, 350, 85, 25);
    btnCari.setLabel("Cari");
    btnKoreksi.setBounds (460, 350, 85, 25);
    btnKoreksi.setLabel("Koreksi");
    btnHapus.setBounds (560, 350, 85, 25);
    btnHapus.setLabel("Hapus");

```

```

btnSelesai.setBounds (675, 350, 85, 25);
btnSelesai.setLabel("Selesai");
btnSelesai.setToolTipText("Mengakhiri Program");

/* Objek Button di Non Aktifkan dan di aktifkan */
btnAdd.setEnabled(true);
btnSelesai.setEnabled(true);

// Mengatur objek untuk dapat berinteraksi
btnAdd.addActionListener (this);
btnCari.addActionListener (this);
btnKoreksi.addActionListener (this);
btnHapus.addActionListener (this);
btnSelesai.addActionListener (this);
tabel.addMouseListener(this);

tabMode01 = new DefaultTableModel(null,header);
tabel.setModel(tabMode01);

skrTabel.getViewport().add(tabel);
tabel.setEnabled(true);
//mengatur koordinat tabel
skrTabel.setBounds(20,190,750,150);

tabel.setAutoResizeMode(JTable.AUTO_RESIZE_OFF);

tabel.getColumnModel().getColumn(0).setPreferredWidth(120);
tabel.getColumnModel().getColumn(1).setPreferredWidth(350);
tabel.getColumnModel().getColumn(2).setPreferredWidth(150);
tabel.getColumnModel().getColumn(3).setPreferredWidth(150);

/* Mengatur Perataan teks di Text */
fHrgBeli.setHorizontalAlignment (JTextField.RIGHT);
fJumlah.setHorizontalAlignment (JTextField.RIGHT);

// Meletakkan seluruh kontrol pada objek panel */
pBrg.add (skrTabel);
pBrg.add (lKode);
pBrg.add (fKode);
pBrg.add (lNama);
pBrg.add (fNama);
pBrg.add (lHrgBeli);
pBrg.add (fHrgBeli);
pBrg.add (lHrgJual);

```

```

        pBrg.add (fHrgJual);
        pBrg.add (lJumlah);
        pBrg.add (fJumlah);
        pBrg.add (lCreate);
        pBrg.add (btnAdd);
        pBrg.add (btnKoreksi);
        pBrg.add (btnHapus);
        pBrg.add (btnCari);
        pBrg.add (btnSelesai);

        /* Menambahkan objek panel (pBrg) ke container frame */
        getContentPane().add
(pBrg).setBackground(Color.getHSBColor(110,110,110));

        /* Menampilkan frame ke layar monitor */
        pack();
        setLocationRelativeTo(null);
        tampiltabel();
        setVisible (true);
    }

    public static void main(String[] args) {
        new FBarang().setVisible(true);
    }

    /* Fungsi jika user melakukan action penekanan tombol Button */
    public void actionPerformed (ActionEvent ae) {
        Object obj = ae.getSource();

        if (obj == btnAdd)
        {
            if (btnAdd.getText()=="Simpan")
            {
                String xnip = fKode.getText();
                if (xnip.length() != 0) Add();
                else
                    JOptionPane.showMessageDialog(this, "Kode kosong
!");

                btnAdd.setText("Add");
                Kosong();
                tampiltabel();
                settombol(1, 1, 0, 0, 1);
            }
            else

```

```

        {
            Kosong();
            settombol(1, 0, 0, 0, 1);
            btnAdd.setText("Simpan");
            btnAdd.setForeground(Color.getHSBColor(100,10,0));
            btnAdd.setBackground(Color.getHSBColor(200,100,100))
        }
    }

    if (obj == btnCari)
    {
        String xcari =
JOptionPane.showInputDialog(this,"Masukkan kode !");
        if (xcari != null)
        {
            fKode.setText(xcari);
            int xx = Cari();
            if (xx == 1) settombol(1, 1, 1, 1, 1);
        }
        else
            JOptionPane.showMessageDialog(this, "kode kosong
!");
    }

    if (obj == btnKoreksi)
    {
        if (btnKoreksi.getText()=="Simpan")
        {
            if (fKode.getText() != "")
                Koreksi();
            btnKoreksi.setText("Koreksi");
            Kosong();
            tampiltabel();
            settombol(1, 1, 0, 0, 1);
        }
        else
        {
            btnKoreksi.setText("Simpan");
            settombol(0, 0, 1, 0, 1);
            fNama.requestFocus();
        }
    }

    if (obj == btnHapus)

```

```

        {
            int stt = JOptionPane.showConfirmDialog(this, "Yakin
dihapus ?");
            System.out.print(stt);

            if (stt == 0)
            {
                Hapus();
                Kosong();
                tampiltabel();
            }
        }

        if (obj == btnSelesai)
        {
            //System.exit(0);
            this.dispose();
        }
    }

    private void tampil()
    {
        try
        {
            String vkode = String.valueOf(tabel.getValueAt
                (tabel.getSelectedRow(), 0));
            String vnama = String.valueOf(tabel.getValueAt
                (tabel.getSelectedRow(), 1));
            String vhrghbeli = String.valueOf(tabel.getValueAt
                (tabel.getSelectedRow(), 1));
            String vhrghjual = String.valueOf(tabel.getValueAt
                (tabel.getSelectedRow(), 1));
            String vjumlah = String.valueOf(tabel.getValueAt
                (tabel.getSelectedRow(), 1));

            fKode.setText(vkode);
            fNama.setText(vnama);
            fHrgBeli.setText(vhrghbeli);
            fHrgJual.setText(vhrghjual);
            fJumlah.setText(vjumlah);
            // yg lain silahkan tambahkan..... !
        }
        catch (Exception e)
        {

```

```

        JOptionPane.showMessageDialog(this, "Ada Kesalahan!!!");
    }
}

// pendefinisian method dari interface mouselistener.....
public void mouseClicked(MouseEvent me)
{
    try
    {
        String vkode = String.valueOf(tabel.getValueAt(
            tabel.getSelectedRow(),0));
        String vnama = String.valueOf(tabel.getValueAt(
            tabel.getSelectedRow(),1));
        String vhrgbeli = String.valueOf(tabel.getValueAt(
            tabel.getSelectedRow(),1));
        String vhrgjual = String.valueOf(tabel.getValueAt(
            tabel.getSelectedRow(),1));
        String vjumlah = String.valueOf(tabel.getValueAt(
            tabel.getSelectedRow(),1));

        fKode.setText(vkode);
        fNama.setText(vnama);
        fHrgBeli.setText(vhrgbeli);
        fHrgJual.setText(vhrgjual);
        fJumlah.setText(vjumlah);
        // yg lain silahkan tambahkan..... !
    }
    catch (Exception e)
    {
        JOptionPane.showMessageDialog(this, "Ada Kesalahan
!!!");
    }
}

public void mouseExited(MouseEvent me){}
public void mouseEntered(MouseEvent me){}
public void mousePressed(MouseEvent me){}
public void mouseReleased(MouseEvent me){}

// Fungsi untuk menambahkan data ke tabel barang
void Add()
{
    try
    {
        KoneksiDBMS CnPenjualan = new KoneksiDBMS();
    }
}

```

```

        Connection con = CnPenjualan.BukaCn();
        Statement stat = con.createStatement();

        String strsqlcr = "SELECT * FROM barang WHERE kode =" + fKode.getText () + "'";
        ResultSet rs = stat.executeQuery(strsqlcr);

        if(rs.next())
        {
            JOptionPane.showMessageDialog(this, "Kode Sudah Ada!");
            return;
        }

        String strsql = "insert into barang(kode, nama) values " +
            " ('"+fKode.getText() +
            "','"+fNama.getText()+"')";
        int stsproses = stat.executeUpdate(strsql);

        // field yg lain silahkan tambahkan..... !

        if (stsproses == 1)
            JOptionPane.showMessageDialog(this, "Sukses Di Tambahkan!!!");
        con.close();
    }
    catch(SQLException e)
    {
        JOptionPane.showMessageDialog(this, "Penambahan Gagal!!!");
        System.err.println("Kesalahan perintah SQL : " + e.getMessage());
    }
}

// Fungsi untuk mencari Kode_Barang ke tabel barang
int Cari(){
    int x = 0;
    try {

        KoneksiDBMS CnPenjualan = new KoneksiDBMS();
        Connection con = CnPenjualan.BukaCn();
        Statement stat = con.createStatement();
    }
}

```



```

        String strsql = "SELECT * FROM barang WHERE kode='" +
fKode.getText () + "'";
        ResultSet rs = stat.executeQuery(strsql);

        if(rs.next())
        {
            // Jika kode Ditemukan Di Tabel barang
            fKode.setText (rs.getString ("kode"));
            fNama.setText (rs.getString ("nama"));
            fHrgBeli.setText (rs.getString ("hrgbeli"));
            fHrgJual.setText (rs.getString ("hrgjual"));
            fJumlah.setText (rs.getString (""));
            // yg lain silahkan tambahkan..... !
            x = 1;
        }
        else
        {
            // Jika kode tidak ditemukan
            JOptionPane.showMessageDialog(this, "Data tidak
ditemukan !");
            fKode.requestFocus ();
        }
        rs.close();
        con.close();
    }
    catch(SQLException e)
    {
    }
    return x;
}

// Fungsi untuk menambahkan data ke tabel barang
void Koreksi()
{
    try
    {
        KoneksiDBMS CnPenjualan = new KoneksiDBMS();
        Connection con = CnPenjualan.BukaCn();
        Statement stat = con.createStatement();

        String strsql = "update barang set nama='" +
fNama.getText()+
        "',hrgbeli='" + fHrgBeli.getText()+
        "' where kode='" + fKode.getText()+"' ";
    }
}

```

```

        int stsproses = stat.executeUpdate(strsql);

        if (stsproses == 1)
            JOptionPane.showMessageDialog(this, "Sukses
DiEdit!");
        con.close();
    }
    catch(SQLException e)
    {
        JOptionPane.showMessageDialog(this, "Koreksi Gagal !");
        System.err.println("Kesalahan perintah SQL : " +
e.getMessage());
    }
}

// Fungsi untuk menambahkan data ke tabel barang
void Hapus()
{
    try
    {
        KoneksiDBMS CnPenjualan = new KoneksiDBMS();
        Connection con = CnPenjualan.BukaCn();
        Statement stat = con.createStatement();

        String strsql = "delete from barang where kode='" +
fKode.getText()+"' ";
        int stsproses = stat.executeUpdate(strsql);

        if (stsproses == 1)
            JOptionPane.showMessageDialog(this, "Data Terhapus
!");
        con.close();
    }
    catch(SQLException e)
    {
        JOptionPane.showMessageDialog(this, "Penghapusan
Gagal!!!");
        System.err.println("Kesalahan perintah SQL : " +
e.getMessage());
    }
}

void tampiltabel()
{
    try {

```

```

        bersihtabel();
        KoneksiDBMS CnPenjualan = new KoneksiDBMS();
        Connection con = CnPenjualan.BukaCn();
        Statement stat = con.createStatement();

        String strsql = "select * from barang";
        rs = stat.executeQuery(strsql);
        ResultSetMetaData meta = rs.getMetaData();

        rs.beforeFirst();
        while(rs.next()) {
            String Kode = rs.getString("kode");
            String Nama = rs.getString("nama");
            String hargaBeli = rs.getString("hrgbeli");
            String hargaJual = rs.getString("hrgjual");
            String Jumlah = rs.getString("jumlah");
            String[] data = {Kode, Nama, hargaBeli, hargaJual,
Jumlah};

            tabMode01.addRow(data);
        }
        stat.close();
        rs.close();
        con.close();

    } catch (SQLException se) {
        System.err.println("Kesalahan perintah SQL : " +
se.getMessage());
    }
}

void bersihtabel()
{
    int brs = tabMode01.getRowCount();
    for(int i=0;i<brs;i++)
    {
        tabMode01.removeRow(0);
    }
}

// Fungsi untuk mengkosongkan Objek masukan
void Kosong ()
{
    fKode.setText("");
    fNama.setText("");
    fHrgBeli.setText("");
}

```

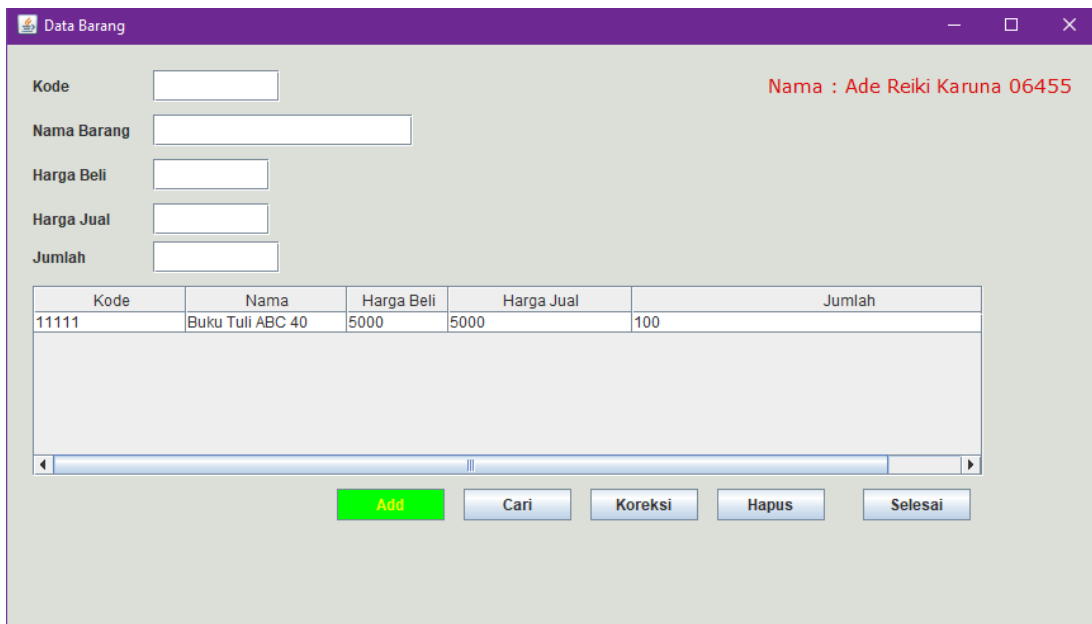
```

        //fHrgJual.setText ("");
        fHrgJual.setText("");
        fJumlah.setText ("");
        fKode.requestFocus ();
    }

    void settombol(int a, int b, int c, int d, int e)
    {
        btnAdd.setEnabled(a>=1 ? true: false);
        btnCari.setEnabled(b>=1 ? true: false);
        btnKoreksi.setEnabled(c>=1 ? true: false);
        btnHapus.setEnabled(d>=1 ? true: false);
        btnSelesai.setEnabled(e>=1 ? true: false);
    }
}

```

Output Program :



Data Barang

Nama : Ade Reiki Karuna 06455

Kode

Nama Barang

Harga Beli

Harga Jual

Jumlah

Kode	Nama	Harga Beli	Harga Jual	Jumlah
11111	Buku Tuli ABC 40	5000	5000	100