

Mata Kuliah : PBO – TI – S1

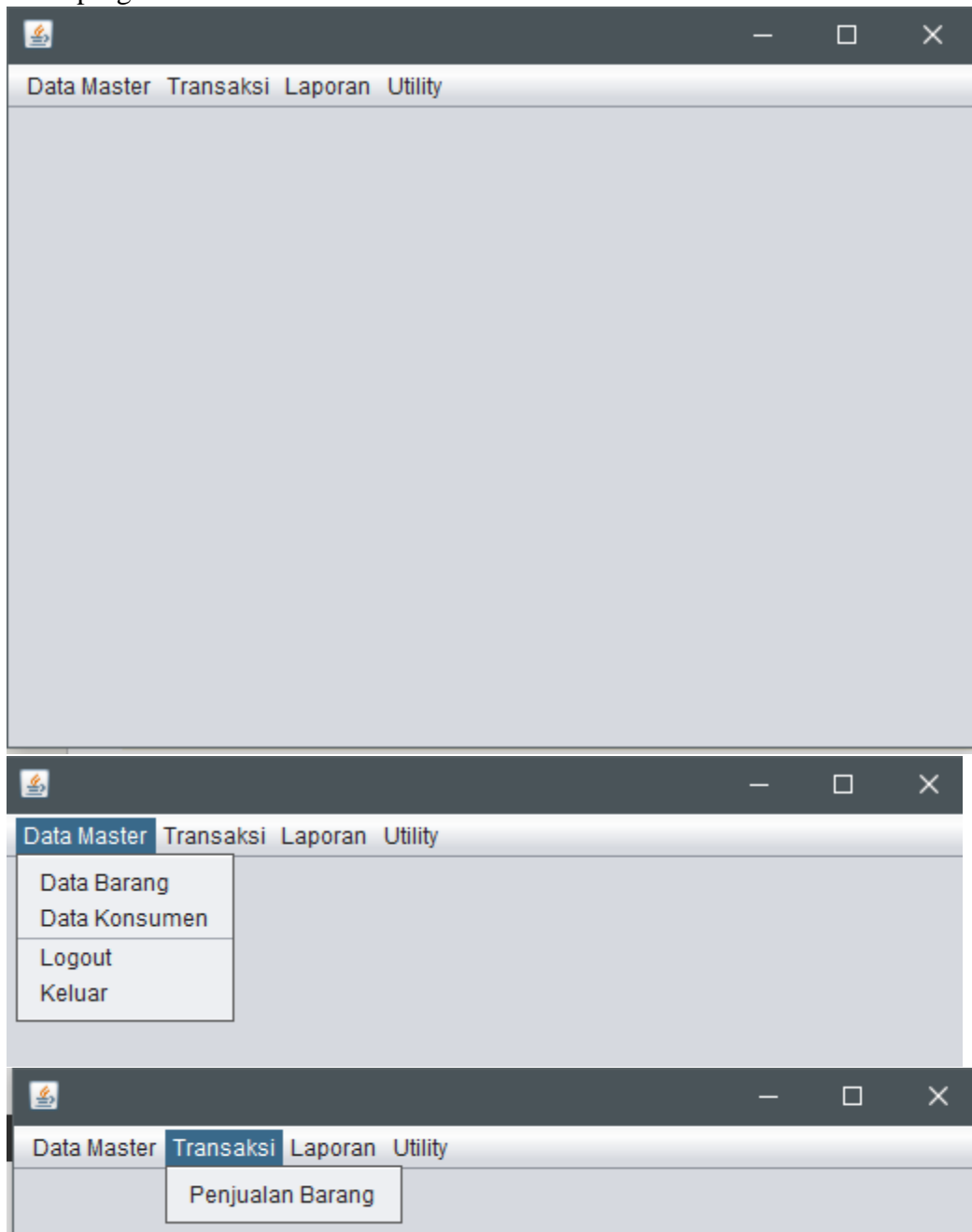
Pertemuan : 12 A – B

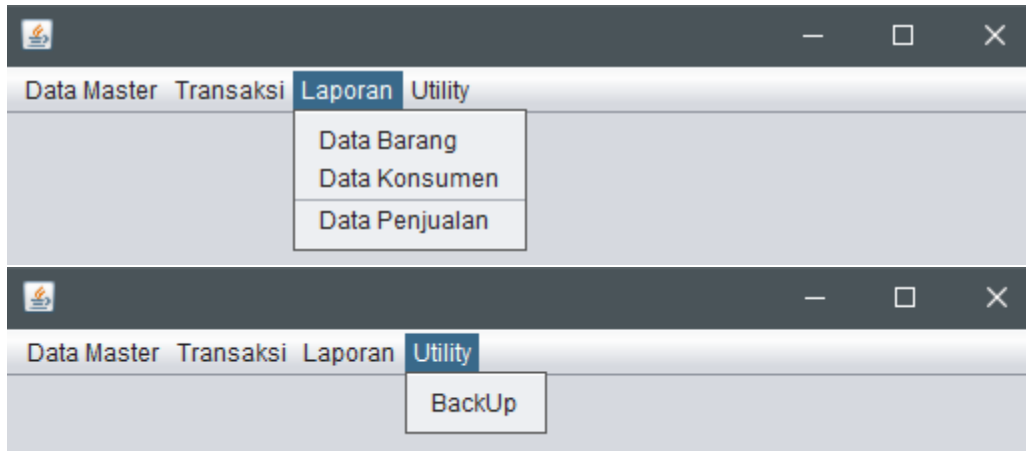
NIM : A11.2022.14532

Nama : Najma Aura Dias Prameswari

1. Tugas 1 : membuat form menu utama

Hasil program :





Code program :

```
package pjppenjualan;
import java.io.File;
import java.sql.*;
import javax.swing.*;
import javax.swing.JOptionPane;
import net.sf.jasperreports.engine.JasperFillManager;
import net.sf.jasperreports.engine.JasperPrint;
import net.sf.jasperreports.engine.JasperReport;
import net.sf.jasperreports.engine.JasperCompileManager;
import net.sf.jasperreports.view.JasperViewer;

public class frmmenu extends javax.swing.JFrame {
    Connection Con;
    public String user_login;

    /**
     * Creates new form frmmenu
     */
    public frmmenu() {
        initComponents();
        open_db();
    }

    public frmmenu(String user_login) {
        initComponents();
        open_db();
    }
}
```

```

private void open_db()
{
    try{
        KoneksiMysql kon= new KoneksiMysql("penjualan");
        // KoneksiMysql kon = new KoneksiMysql ("192.168.10.1");
        Con = kon.getConnection();
        //System.out.println("Berhasil ");
    }catch (Exception e) {
        System.out.println("Error : "+e);
    }
}

private void mnBarangActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    new frmBarang().setVisible(true);
}

private void mnKonsumenActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    new frmKonsumen().setVisible(true);
}

private void mnLogoutActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    int psn=JOptionPane.showConfirmDialog(this, "Logout System ?", "Pesan",JOptionPane.OK_CANCEL_OPTION, JOptionPane.QUESTION_MESSAGE);
    if (psn==0) {
        dispose();
        new frmLogin().setVisible(true);
    }
}

private void mnKeluarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    int psn=JOptionPane.showConfirmDialog(this, "Yakin keluar", "Pesan",JOptionPane.OK_CANCEL_OPTION, JOptionPane.QUESTION_MESSAGE);
    if (psn==0) System.exit(0);
}

private void mnJualActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    new frmTransaksi().setVisible(true);
}

private void mnLapBarangActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    // report rpt = new report("lbarang.jrxml", Con);
    // JasperViewer.viewReport("lap_brg.jrxml",Con);
    // new frmLaporanBarang().setVisible(true);
    try {
        Class.forName("com.mysql.jdbc.Driver");
        Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/penjualan", "root", "");
        String reportPath = "C:/Users/HP/Documents/PRAKTIK-PBO-JAVA-APPLICATION/pjpPenjualan/src/pjppenjualan/Lap_brg.jrxml"; //

        // Memverifikasi path absolut
        File reportFile = new File(reportPath);
        System.out.println("Looking for report at: " + reportFile.getAbsolutePath());
        if (!reportFile.exists()) {
            System.out.println("File not found at the specified path!");
        } else {
            JasperReport jr = JasperCompileManager.compileReport(reportPath);
            JasperPrint jp = JasperFillManager.fillReport(jr, null, con);
            JasperViewer.viewReport(jp);
        }
        con.close();
    } catch (Exception ex) {
        ex.printStackTrace();
    }
}

```

```

private void mnLapJualActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    new LapPenjualan().setVisible(true);
    // report rpt=new report("lap_penjualan.jrxml", Con);
}

/*
 * Set the Nimbus look and feel */
Look and feel setting code (optional)

/* Create and display the form */
java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
        new frmmenu().setVisible(true);
    }
});
}

// Variables declaration - do not modify
private javax.swing.JCheckBoxMenuItem jCheckBoxMenuItem1;
private javax.swing.JCheckBoxMenuItem jCheckBoxMenuItem2;
private javax.swing.JCheckBoxMenuItem jCheckBoxMenuItem3;
private javax.swing.JCheckBoxMenuItem jCheckBoxMenuItem4;
private javax.swing.JMenu jMenu5;
private javax.swing.JMenu jMenu6;
private javax.swing.JMenuBar jMenuBar1;
private javax.swing.JMenuBar jMenuBar3;
private javax.swing.JMenuItem jMenuItem1;
private javax.swing.JMenuItem jMenuItem2;
private javax.swing.JMenuItem jMenuItem3;
private javax.swing.JSeparator jSeparator1;
private javax.swing.JSeparator jSeparator2;
private javax.swing.JMenuItem mnBarang;
private javax.swing.JMenu mnData;
private javax.swing.JMenuItem mnJual;
private javax.swing.JMenuItem mnKeluar;
private javax.swing.JMenuItem mnKonsumen;
private javax.swing.JMenuItem mnLapBarang;
private javax.swing.JMenuItem mnLapJual;
private javax.swing.JMenuItem mnLapKonsumen;
private javax.swing.JMenu mnLaporan;
private javax.swing.JMenuItem mnLogout;
private javax.swing.JMenu mnTransaksi;
private javax.swing.JMenu mnUtility;
// End of variables declaration
}

```

2. Tugas 2 : membuat form transaksi
Hasil program :

Data Master Transaksi Laporan Utility

No Jual: 7
Tgl. Jual: 2024/07/10
Kode Konsumen: K08
Nama Konsumen: Lia

Kode Barang: K07
Nama Barang: Wortel
Harga Satuan: 7000.0
Jumlah: 10
Total: 70000.0

Hapus Item
Pilih Barang
K07

Kode Barang	Nama Barang	Harga Barang	Jumlah	Total
K07	Wortel	7000.0	10	70000.0

Nota Penjualan
No : 7
Tanggal : 2024-7-10

Kode	Nama Barang	Harga	Jml	Total
K07	Wortel	7000.0	10	70000.0

Total: 70000.0
Bayar: 80000
Kembali: 10000.0

Tambah Simpan Batal Cetak Keluar

Code program :

```
package pjppenjualan;
import java.awt.print.PrinterException;
import java.sql.*;
import java.sql.Date.*;
import java.text.MessageFormat;
import javax.swing.*;
import javax.swing.table.*;
import java.util.Calendar;
import javax.swing.JOptionPane;
```

```

public class frmTransaksi extends javax.swing.JFrame {

    Connection Con;
    ResultSet RsBrg;
    ResultSet RsKons;
    Statement stm;
    double total = 0;
    String tanggal,tgl1;
    Boolean edit=false;
    public TableCellRenderer kanan = new RataKanan();
    public TableCellRenderer tengah = new RataTengah();

    DefaultTableModel tableModel = new DefaultTableModel(
        new Object [][] {},
        new String [] {
            "Kode Barang", "Nama Barang", "Harga Barang", "Jumlah", "Total"
        }
    );

    public String idBrg, namaBrg, hargaBrg;

    public String getIdBrg() {
        return idBrg;
    }

    public String getnamaBrg() {
        return namaBrg;
    }

    public String gethargaBrg() {
        return hargaBrg;
    }
}

```

```

public frmTransaksi() {
    initComponents();
    open_db();
    inisialisasi_table();
    aktif(false);
    setTombol(true);
    txtTgl.setEditor(new JSpinner.DateEditor(txtTgl, "yyyy/MM/dd"));
    format_tanggal();
    java.text.SimpleDateFormat sdf = new java.text.SimpleDateFormat("yyyy-MM-dd");
    dtTgl.setDateFormat(sdf);
}

private void aturTabel()
{
    tblJual.getColumnModel().getColumn(2).setCellRenderer(kanan);
    tblJual.getColumnModel().getColumn(3).setCellRenderer(kanan);
    tblJual.getColumnModel().getColumn(4).setCellRenderer(kanan);
}

private void kosongTabel()
{
    try{
        DefaultTableModel dtm = (DefaultTableModel) tblJual.getModel();
        dtm.setNumRows(0);
    }catch(Exception e){
    }
}

private int getStok(String xkode)
{
    int jStok=0;

    try{
        stm=Con.createStatement();
        ResultSet rs=stm.executeQuery("select kd_barang,stok from barang where kd_barang='" + xkode + "' limit 1");
        rs.beforeFirst();

        while(rs.next())
        {
            jStok=rs.getInt("stok");
        }

        rs.close();

    }catch(SQLException e)
    {
        System.out.println("Error : "+e);
    }

    return jStok;
}

```

```

private boolean cekRowData(String cari)
{
    boolean cek=false;
    for(int i = 0; i < tblJual.getRowCount(); i++){//For each row
        for(int j = 0; j < tblJual.getColumnCount(); j++){//For each column in that row
            if(tblJual.getModel().getValueAt(i, j).equals(cari)){//Search the model
                System.out.println(tblJual.getModel().getValueAt(i, j));//Print if found string
                cek=true;
            }
        }
    }
    return cek;
}

private double getTotal()
{
    double xtotal=0;
    for(int i = 0; i < tblJual.getRowCount(); i++){//For each row
        {
            xtotal=xtotal+Double.parseDouble(tblJual.getModel().getValueAt(i, 4).toString());
        }
    }
    return xtotal;
}

private void hitung_jual() {
    double x_total, x_harga;
    int x_jumlah;
    x_harga = Double.parseDouble(txtHarga.getText());
    x_jumlah = Integer.parseInt(txtJml.getText());
    x_total = x_harga * x_jumlah;
    String x_t1 = Double.toString(x_total);
    txtTot.setText(x_t1);
    total = total + x_total;
    txtTotal.setText(Double.toString(total));
}

private void open_db() {
    try{
        KoneksiMysql kon = new KoneksiMysql("localhost","root","","penjualan");
        Con = kon.getConnection();
        //System.out.println("Berhasil ");
    }catch (Exception e) {
        System.out.println("Error : "+e);
    }
}

```



```

private void baca_konsumen() {
    try{
        // cmbKd_Kons.removeAllItems();
        stm=Con.createStatement();
        ResultSet rs=stm.executeQuery("select kd_konsumen,nm_konsumen from konsumen");

        rs.beforeFirst();
        while(rs.next())
        {
            //comboBox.addItem(new TypeEntry(rs.getInt(1),rs.getString(2).trim()));
            // cmbKd_Kons.addItem(new TypeEntry(RsKons.getString(1).trim(),RsKons.getStr
            cmbKd_Kons.addItem(rs.getString(1).trim());
        }
        rs.close();
    }
    catch(SQLException e)
    {
        System.out.println("Error : "+e);
    }
}

public void inisialisasi_table() {
    tblJual.setModel(tableModel);
}

```

```

private void baca_barang() {
    try{
        // cmbKd_Brg.removeAllItems();
        stm=Con.createStatement();
        ResultSet rs=stm.executeQuery("select * from barang");

        rs.beforeFirst();

        while(rs.next())
        {
            cmbKd_Brg.addItem(rs.getString(1).trim());
        }
        rs.close();
    }
    catch(SQLException e)
    {
        System.out.println("Error : "+e);
    }
}

private void detail_barang(String x_kode) {
    try{
        stm=Con.createStatement();
        ResultSet rs=stm.executeQuery("select * from barang where kd_barang='"+x_kode+"'");

        rs.beforeFirst();
    }
}

```

```

while(rs.next())
{
    txtNm_Brg.setText(rs.getString(2).trim());
    txtHarga.setText(Double.toString((Double)rs.getDouble(4)));
}
rs.close();
}
catch(SQLException e)
{
    System.out.println("Error : "+e);
}

private void detail_konsumen(String x_kode){
try{
    stm=Con.createStatement();
    ResultSet rs=stm.executeQuery("select * from konsumen where kd_konsumen='"+x_kode+"'");

    rs.beforeFirst();

    while(rs.next())
    {
        txtNama.setText(rs.getString(2).trim());
    }
    rs.close();
}
catch(SQLException e)
{
    ;
}
}

```

```

private void detail_konsumen(String x_kode){
try{
    stm=Con.createStatement();
    ResultSet rs=stm.executeQuery("select * from konsumen where kd_konsumen='"+x_kode+"'");

    rs.beforeFirst();

    while(rs.next())
    {
        txtNama.setText(rs.getString(2).trim());
    }
    rs.close();
}
catch(SQLException e)
{
    System.out.println("Error : "+e);
}

//pengkosongan isian
private void kosong(){
    txtNoJual.setText("");
    txtNama.setText("");
    txtHarga.setText("");
    txtTotal.setText("");
    txtBayar.setText("");
    txtKembali.setText("");

}

//kosongkan detail jual
private void kosong_detail(){
    txtNm_Brg.setText("");
    txtHarga.setText("");
    txtJml.setText("");
    txtTot.setText("");
    text.setText("");
}

//aktif on/off isian
private void aktif(boolean x){
    cmbKd_Kons.setEnabled(x);
    cmbKd_Brg.setEnabled(x);
    txtTgl.setEnabled(x);
    txtJml.setEditable(x);
    txtBayar.setEditable(x);
}

```

```

private void setTombol(boolean t){
//    btn_tambah.setEnabled(t);
//    btn_hpus_item.setEnabled(!t); //(t)
//    btn_simpan.setEnabled(!t);
//    btn_batal.setEnabled(!t);
//    btn_keluar.setEnabled(t);

    cmdTambah.setEnabled(t);
    cmdSimpan.setEnabled(!t);
    cmdBatal.setEnabled(!t);
    cmdKeluar.setEnabled(t);
    cmdHapusItem.setEnabled(!t);
}

private void nomor_jual(){
try{
    stm=Con.createStatement();
    ResultSet rs=stm.executeQuery("select no_jual from jual");
    int brs=0;

    while(rs.next())
    {
        brs=rs.getRow();
    }
    if(brs==0)
        txtNoJual.setText("1");
    else
    {
        int nom=brs+1;
        txtNoJual.setText(Integer.toString(nom));
    }
    rs.close();
}
catch(SQLException e)
{
    System.out.println("Error : "+e);
}
}

```

```

private void simpan_ditable() {
    try{
        String tKode=cmbKd_Brg.getSelectedItem().toString();
        String tNama=txtNm_Brg.getText();

        double hrg=Double.parseDouble(txtHarga.getText());
        int jml=Integer.parseInt(txtJml.getText());
        double tot=Double.parseDouble(txtTot.getText());
        tableModel.addRow(new Object[] {tKode,tNama,hrg,jml,tot});
        inisialisasi_table();

    }catch(Exception e)
    {
        System.out.println("Error : "+e);
    }

}

private void simpan_transaksi() {
    try{
        String x_nojual=txtNoJual.getText();
        format_tanggal();
        String x_kode=cmbKd_Kons.getSelectedItem().toString();
        String msq1="insert into jual values('"+x_nojual+"','"+x_kode+"','"+tanggal+"')";
        stm.executeUpdate(msq1);

        for(int i=0;i<tblJual.getRowCount();i++)
        {
            String xkd=(String)tblJual.getValueAt(i,0);
            double xhr=(Double)tblJual.getValueAt(i,2);
            int xjml=(Integer)tblJual.getValueAt(i,3);
            String zs1="insert into djual values('"+x_nojual+"','"+xkd+"','"+xhr+"','"+xjml+"')";
            stm.executeUpdate(zs1);

            //update stok
            String ys1="update barang set stok=stok-"xjml where kd_barang='"xkd+'";
            stm.executeUpdate(ys1);

        }
    }catch(SQLException e){
        System.out.println("Error : "+e);
    }

}

```

```

private void format_tanggal()
{
    String DATE_FORMAT = "yyyy-MM-dd";
    //String DATE_FORMAT = "dd-MM-yyyy"; //Refer Java DOCS for formats
    java.text.SimpleDateFormat sdf = new java.text.SimpleDateFormat(DATE_FORMAT);
    Calendar cl = Calendar.getInstance();
    int year=cl.get(Calendar.YEAR);
    int month=cl.get(Calendar.MONTH)+1;
    int day=cl.get(Calendar.DAY_OF_MONTH);
    //tgl1=sdf.format(txtTgl.getValue());
    tgl1=dtTgl.getText();
    tanggal=Integer.toString(year)+"-"+Integer.toString(month)+"-"+Integer.toString(day);
    System.out.println("Tanggal : " + tgl1);
    System.out.println("Tanggal2 : " + tanggal);
}

private class PrintingTask extends SwingWorker<Object, Object> {
    private final MessageFormat headerFormat;
    private final MessageFormat footerFormat;
    private final boolean interactive;
    private volatile boolean complete = false;
    private volatile String message;
    / private final TextArea teks;

    public PrintingTask( MessageFormat header, MessageFormat footer, boolean interactive) {
    / this.teks = teks;
        this.headerFormat = header;
        this.footerFormat = footer;
        this.interactive = interactive;
    }
    @Override
    protected Object doInBackground() {
        try {

            complete = text.print(headerFormat, footerFormat, true, null, null, interactive);
            message = "Printing " + (complete ? "complete" : "canceled");
        } catch (PrinterException ex) {
            message = "Sorry, a printer error occurred";
        } catch (SecurityException ex) {
            message = "Sorry, cannot access the printer due to security reasons";
        }
        return null;
    }

    @Override

```

```

@Override
protected void done() {
    JOptionPane.showMessageDialog(null, message);
    System.out.println(message);
    message(!complete, message);
}
}

private MessageFormat createFormat(String source) {
    String text = source;
    if (text != null && text.length() > 0) {
        try {
            return new MessageFormat(text);
        } catch (IllegalArgumentException e) {
            error("Sorry, this format is invalid.");
        }
    }
    return null;
}

private void message(boolean error, String msg) {
    int type = (error ? JOptionPane.ERROR_MESSAGE : JOptionPane.INFORMATION_MESSAGE);
    JOptionPane.showMessageDialog(this, msg, "Printing", type);
}

private void error(String msg) {
    message(true, msg);
}

public static void createAndShowGUI() {
    JFrame f = new frmTransaksi();
    f.setLocationRelativeTo(null);
    f.setVisible(true);
}

public void itemTerpilih(){
    frmCariBarang fDB = new frmCariBarang();
    fDB.fAB = this;
    txtId.setText(idBrg);
    txtNm_Brg.setText(namaBrg);
    txtHarga.setText(hargaBrg);
}

```

```
private void txtBayarFocusLost(java.awt.event.FocusEvent evt) {  
    // TODO add your handling code here:  
    double tot=Double.parseDouble(txtTotal.getText());  
    double byr=Double.parseDouble(txtBayar.getText());  
    double kembali=byr-tot;  
    txtKembali.setText(Double.toString(kembali));  
}
```

```
private void txtBayarActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    try{  
        if(Double.parseDouble(txtBayar.getText())< Double.parseDouble(txtTotal.getText()))  
        {  
            JOptionPane.showMessageDialog(null, "Jumlah Bayar lebih kecil (<) dari Total, Ulangi....!!!");  
            return;  
        }  
    }catch(NumberFormatException e){  
        JOptionPane.showMessageDialog(null, e);  
        return;  
    }  
    double tot=Double.parseDouble(txtTotal.getText());  
    double byr=Double.parseDouble(txtBayar.getText());  
    double kembali=byr-tot;  
    txtKembali.setText(Double.toString(kembali));  
}
```

```
private void txtBayarKeyPressed(java.awt.event.KeyEvent evt) {  
    // TODO add your handling code here:  
    if (evt.getKeyPressed()==13)  
    {  
        double tot=Double.parseDouble(txtTotal.getText());  
        double byr=Double.parseDouble(txtBayar.getText());  
        double kembali=byr-tot;  
        txtKembali.setText(Double.toString(kembali));  
    }  
}
```

```
private void cmbKd_BrgActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    detail_barang(cmbKd_Brg.getSelectedItem().toString());  
}
```



```

private void txtJmlActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try{
        if(cekRowData(cmbKd_Brg.getSelectedItem().toString())==true)
        {
            JOptionPane.showMessageDialog(null, "Item sudah ada di keranjang..... "+cmbKd_Brg.getSelectedItem().toString());
            return;
        }
        if(Integer.parseInt(txtJml.getText())==0)
        {
            JOptionPane.showMessageDialog(null, "Jumlah Jual masih kosong /0 ");
        }
        else{
            if(getStok(cmbKd_Brg.getSelectedItem().toString())<Integer.parseInt(txtJml.getText()))
            {
                JOptionPane.showMessageDialog(null, "Stok tidak mencukupi...!! "+getStok(cmbKd_Brg.getSelectedItem().toString()));
            }
            else {
                hitung_jual();
                simpan_ditable();
            }
        }
    }
    catch(NumberFormatException e)
    {
        JOptionPane.showMessageDialog(null, e);
    }
}

```

```

private void btnPilihActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    frmCariBarang fDB = new frmCariBarang();
    fDB.fAB = this;
    fDB.setVisible(true);
    fDB.setResizable(false);
}

```

```

private void cmdHapusItemActionPerformed(java.awt.event.ActionEvent evt)
{
    // TODO add your handling code here:
    int row=tblJual.getSelectedRow();
    ((DefaultTableModel) tblJual.getModel()).removeRow(row);
    txtTotal.setText(Double.toString(getTotal()));
    //  inisialisasi_tabel();
}

```

```

private void cmdTambahActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    kosong();
    total=0;
    kosong_detail();
    ComboBoxModel c1=new DefaultComboBoxModel();
    ComboBoxModel c2=new DefaultComboBoxModel();
    cmbKd_Kons.setModel(c1);
    cmbKd_Brg.setModel(c2);
    baca_konsumen();
    baca_barang();
    aktif(true);
    setTombol(false);
    nomor_jual();
    //        kosongTabel();
    aturTabel();
}

```

```

private void cmdSimpanActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try{
        if(Double.parseDouble(txtBayar.getText())< Double.parseDouble(txtTotal.getText()))
        {
            JOptionPane.showMessageDialog(null, "Jumlah Bayar lebih kecil (<) dari Total, Ulangi...!!!!");
            return;
        }

        simpan_transaksi();
    }catch (NumberFormatException e){
        JOptionPane.showMessageDialog(null, e);
        return;
    }
    JOptionPane.showMessageDialog(null, "Data Berhasil di simpan");
    aktif(false);
    setTombol(true);
}

```

```

private void cmdBatalActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    aktif(false);
    setTombol(true);
}

```

```

private void cmbKd_KonsActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    detail_konsumen(cmbKd_Kons.getSelectedItem().toString());
}

```

```

private void cmdCetakActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String ctk="Nota Penjualan\nNo      :"+txtNoJual.getText()+"\nTanggal : "+tanggal;
    ctk=ctk+"\n"+-----";
    ctk=ctk+"\n"+"Kode\tNama Barang\tHarga\tJml\tTotal";
    ctk=ctk+"\n"+-----";

    for (int i=0;i<tblJual.getRowCount();i++)
    {
        String xkd=(String)tblJual.getValueAt(i,0);
        String xnama=(String)tblJual.getValueAt(i,1);
        double xhrg=(Double)tblJual.getValueAt(i,2);
        int xjml=(Integer)tblJual.getValueAt(i,3);
        double xtota=(Double)tblJual.getValueAt(i,4);
        ctk=ctk+"\n"+xkd+"\t"+xnama+"\t"+xhrg+"\t"+xjml+"\t"+xtota;
    }
    ctk=ctk+"\n"+-----";
    txt.setText(ctk);
    String headerField="";
    String footerField="";
    MessageFormat header = createFormat(headerField);
    MessageFormat footer = createFormat(footerField);
    boolean interactive = true;//interactiveCheck.isSelected();
    boolean background = true;//backgroundCheck.isSelected();

    PrintingTask task = new PrintingTask(header, footer, interactive);
    if (background) {
        task.execute();
    } else {
        task.run();
    }

    task.run();
}

}

private void cmdKeluarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    //System.exit(0);
    dispose();
}

}

public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    Look and feel setting code (optional)

    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new frmTransaksi().setVisible(true);
        }
    });
}
}

```

```

private javax.swing.JButton btnPilih;
private javax.swing.JComboBox cmbKd_Brg;
private javax.swing.JComboBox cmbKd_Kons;
private javax.swing.JButton cmdBatal;
private javax.swing.JButton cmdCetak;
private javax.swing.JButton cmdHapusItem;
private javax.swing.JButton cmdKeluar;
private javax.swing.JButton cmdSimpan;
private javax.swing.JButton cmdTambah;
private datechooser.beans.DateChooserDialog dateChooserDialog1;
private datechooser.beans.DateChooserDialog dateChooserDialog2;
private datechooser.beans.DateChooserCombo dtTgl;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel10;
private javax.swing.JLabel jLabel11;
private javax.swing.JLabel jLabel12;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
    private javax.swing.JLabel jLabel9;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JScrollPane jScrollPane2;
    private javax.swing.JTable tblJual;
    private javax.swing.JTextArea text;
    private javax.swing.JTextField txtBayar;
    private javax.swing.JTextField txtHarga;
    private javax.swing.JTextField txtId;
    private javax.swing.JTextField txtJml;
    private javax.swing.JTextField txtKembali;
    private javax.swing.JTextField txtNama;
    private javax.swing.JTextField txtNm_Brg;
    private javax.swing.JTextField txtNoJual;
    private javax.swing.JSpinner txtTgl;
    private javax.swing.JTextField txtTot;
    private javax.swing.JTextField txtTotal;
    // End of variables declaration
}

```

Nota Penjualan

No :7

Tanggal : 2024-7-10

Kode	Nama Barang	Harga	Jml	Total
K07	Wortel	7000.0	10	70000.0

3. Tugas 3 : membuat form cari barang

Hasil program :

DAFTAR BARANG						
Kode	Nama Barang	Satuan	Harga Jual	Harga Beli	Stok	Stok Min
K01	Apel	Buah	10000.0	7000.0	47	10
K02	Jeruk	Buah	8000.0	5000.0	40	8
K03	Pisang	Buah	6000.0	4000.0	60	12
K04	Mangga	Buah	12000.0	8000.0	30	6
K05	Semangka	Buah	15000.0	10000.0	0	4
K06	Bayam	Sayur	5000.0	3000.0	65	14
K07	Wortel	Sayur	7000.0	4500.0	40	10
K08	Kubis	Sayur	8000.0	6000.0	40	8
K09	Buncis	Sayur	6500.0	4000.0	60	12
K10	Tomat	Sayur	9000.0	6500.0	35	7

*Klik pada baris untuk produk yang dipilih

Code program :

```

package pjppenjualan;

import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.ResultSetMetaData;
import java.sql.SQLException;
import java.sql.Statement;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;

public class frmCariBarang extends javax.swing.JFrame {

    Connection Con;
    ResultSet RsBrg;
    Statement stm;
    public frmTransaksi fAB = null;
    private Object[][] dataTable = null;
    private String[] header = {"Kode", "Nama Barang", "Satuan", "Harga Jual", "Harga Beli", "Stok", "Stok Min"};

    /**
     * Creates new form frmCariBarang
     */
    public frmCariBarang() {
        initComponents();
        baca_data();
        daftar_table.addMouseListener(new MouseAdapter() {
            public void mouseClicked(MouseEvent evt) {
                daftar_tableMouseClicked(evt);
            }
        });
    }

    private void open_db()
    { try{
        KoneksiMysql kon = new KoneksiMysql("localhost","root","","penjualan");
        Con = kon.getConnection();
        //System.out.println("Berhasil ");
    }catch (Exception e) {
        System.out.println("Error : "+e);
    }
    }
}

```

```

private void baca_data()
{
    open_db();
    try{
        stm = Con.createStatement();
        RsBrg = stm.executeQuery("select * from barang order by kd_barang");

        ResultSetMetaData meta = RsBrg.getMetaData();
        int col = meta.getColumnCount();
        int baris = 0;
        while(RsBrg.next()) {
            baris = RsBrg.getRow();
        }

        dataTable = new Object[baris][col];
        int x = 0;
        RsBrg.beforeFirst();
        while(RsBrg.next()) {
            dataTable[x][0] = RsBrg.getString("kd_barang");
            dataTable[x][1] = RsBrg.getString("nm_barang");
            dataTable[x][2] = RsBrg.getString("satuan");
            dataTable[x][3] = RsBrg.getDouble("harga_jual");
            dataTable[x][4] = RsBrg.getDouble("harga_beli");
            dataTable[x][5] = RsBrg.getInt("stok");
            dataTable[x][6] = RsBrg.getInt("stok_min");
            x++;
        }
        daftar_table.setModel(new DefaultTableModel(dataTable, header));
    }
    catch(SQLException e)
    {
        JOptionPane.showMessageDialog(null, e);
    }
}

private void daftar_tableMouseClicked(java.awt.event.MouseEvent evt) {
    int row=daftar_table.getSelectedRow();
    int tabelBarang = daftar_table.getSelectedRow();
    fAB.idBrg = daftar_table.getValueAt(tabelBarang, 0).toString();
    fAB.namaBrg = daftar_table.getValueAt(tabelBarang, 1).toString();
    fAB.hargaBrg = daftar_table.getValueAt(tabelBarang, 3).toString();
    fAB.itemTerpilih();
    this.dispose();
}

```

```

    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        Look and feel setting code (optional)

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new frmCariBarang().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify
    private javax.swing.JTable daftar_table;
    private javax.swing.JLabel dftr_brg;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JScrollPane jScrollPane1;
    // End of variables declaration
}

```

Database :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	no_jual	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/> 2	kd_brg	char(6)	utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/> 3	harga_jual	double			No	None			Change Drop More
<input type="checkbox"/> 4	jml_jual	int(4)			No	None			Change Drop More

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	no_jual	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/> 2	kd_kons	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/> 3	tgl_jual	date			No	None			Change Drop More