

## PRAKTIKUM #12B

### Form Transaksi

Transaksi Penjualan Barang

No Jual

Kode Konsumen

Tgl. Jual

Nama Konsumen

Hapus Item

Kd Barang	Nama Barang	Harga Barang	Jumlah
-----------	-------------	--------------	--------

Total

Tambah Simpan Batal Cetak Keluar

Buat Tabel Jual dan Djual untuk penyimpanan data transaksi.

```
MySQL Command Line Client
mysql> desc jual;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| no_jual | varchar(10) | NO | PRI | NULL | |
| kd_kons | char(6) | YES | | NULL | |
| tgl_jual | date | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> desc djual;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| no_jual | varchar(10) | NO | PRI | NULL | |
| kd_brg | char(6) | NO | PRI | NULL | |
| harga_jual | float | YES | | NULL | |
| jml_jual | int(4) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)

mysql> _
```

### Alur Program :

1. frmTransaksi created
2. panggil konstruktor frmTransaksi sekaligus open database
3. isian jtextfiled off
4. button tambah, cetak, keluar on sedang simpan , batal dan delete item off.
5. pindahkan data kode barang dari table barang di Mysql ke dalam combo kode barang.
6. pindahkan data kode konsumen dari table konsumen di Mysql ke dalam combo kode konsumen.
7. Jika ditekan tombol tambah, maka actionPerformed pada button tambah dijalankan.
8. Saat dipilih kode konsumen pada combo maka nama konsumen keluar otomatis, juga pada combo barang maka nama barang, harga keluar otomatis.
9. Jumlah jual diinput user, jika di enter maka total jual akan otomatis terjumlahkan, dan data dipindah ke tabeljual.
10. Jika ditekan tombol simpan, maka actionPerformed pada button simpan dijalankan.
11. Jika ditekan tombol delete item, maka actionPerformed pada button delete item dijalankan.
12. Jika ditekan tombol keluar, maka actionPerformed pada button keluar dijalankan.

### Code dan Method-method yang dibuat

Tambahkan code di bawah

**public class frmTransaksi extends javax.swing.JFrame {**

```
Connection Con;
ResultSet RsBrg;
ResultSet RsKons;
Statement stm;
double total=0;
String tanggal;
Boolean edit=false;

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

**Tambahkan code di bawah konstruktor :**

```
public frmTransaksi() {
    initComponents();
    open_db();
    inisialisasi_tabel();
    aktif(false);
    setTombol(true);
    txtTgl.setEditor(new JSpinner.DateEditor(txtTgl,
"yyyy/MM/dd")); }
```

**Buat Method-method berikut :**

//method hitung penjualan

```
private void hitung_jual()
{
    double xtot,xhrg;
    int xjml;
    xhrg=Double.parseDouble(txtHarga.getText());
    xjml=Integer.parseInt(txtJml.getText());
    xtot=xhrg*xjml;
    String xtotal=Double.toString(xtot);
    txtTot.setText(xtotal);
    total=total+xtot;
    txtTotal.setText(Double.toString(total));
}
```

//method buka database

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

//method baca data konsumen

```
private void baca_konsumen()
{
    try{
        stm=Con.createStatement();
        ResultSet rs=stm.executeQuery("select kd_kons,nm_kons from
konsumen");

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

//method baca data barang

```
private void baca_barang()
{
    try{
        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);
    }
}
//method baca barang setelah combo barang di klik
private void detail_barang(String xkode)
{
    try{
        stm=Con.createStatement();
        ResultSet rs=stm.executeQuery("select * from barang where
kd_brg='"+xkode+"'");

        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);
    }
}
//method baca konsumen setelah combo konsumen di klik
private void detail_konsumen(String xkode)
{
    try{
        stm=Con.createStatement();
        ResultSet rs=stm.executeQuery("select * from konsumen where
kd_kons='"+xkode+"'");
        rs.beforeFirst();
        while(rs.next())
        {
            txtNama.setText(rs.getString(2).trim());
        }
        rs.close();
    }
    catch(SQLException e)
    {
        System.out.println("Error : "+e);
    }
}
//method set model tabel
public void inisialisasi_tabel()
{
    tblJual.setModel(tableModel);
}
```

```
//method pengkosongan isian
private void kosong()
{
    txtNoJual.setText("");
    txtNama.setText("");
    txtHarga.setText("");
    txtTotal.setText("");
}
//method kosongkan detail jual
private void kosong_detail()
{
    txtNm_Brg.setText("");
    txtHarga.setText("");
    txtJml.setText("");
    txtTot.setText("");
}
//method aktif on/off isian
private void aktif(boolean x)
{
    cmbKd_Kons.setEnabled(x);
    cmbKd_Brg.setEnabled(x);
    txtTgl.setEnabled(x);
    txtJml.setEditable(x);
}
//method set tombol on/off
private void setTombol(boolean t)
{
    cmdTambah.setEnabled(t);
    cmdSimpan.setEnabled(!t);
    cmdBatal.setEnabled(!t);
    cmdKeluar.setEnabled(t);
    cmdHapusItem.setEnabled(!t);
}
//method buat nomor jual otomatis
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);
    }
}
```

## Panduan Aplikasi Penjualan Java + Mysql + NetBean

```
//method simpan detail jual di tabel
private void simpan_ditabel()
{
    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_tabel();
    }
    catch(Exception e)
    {
        System.out.println("Error : "+e);
    }
}

//method simpan transaksi penjualan pada table di MySql
private void simpan_transaksi()
{
    try{
        String xnojual=txtNoJual.getText();
        format_tanggal();
        String xkode=cmbKd_Kons.getSelectedItem().toString();
        String msql="insert into jual
values('"+xnojual+"','"+xkode+"','"+tanggal+"')";
        stm.executeUpdate(msql);

        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 zsql="insert into djual
values('"+xnojual+"','"+xkd+"','"+xhr+"','"+xjml+"')";
            stm.executeUpdate(zsql);
        }
    }
    catch(SQLException e)
    {
        System.out.println("Error : "+e);
    }
}

//method membuat format tanggal sesuai dengan MySQL
private void format_tanggal()
{
    String DATE_FORMAT = "yyyy-MM-dd";
    java.text.SimpleDateFormat sdf = new
java.text.SimpleDateFormat(DATE_FORMAT);
    Calendar c1 = Calendar.getInstance();
    int year=c1.get(Calendar.YEAR);
    int month=c1.get(Calendar.MONTH)+1;
    int day=c1.get(Calendar.DAY_OF_MONTH);
    tanggal=Integer.toString(year)+"-"+Integer.toString(month)+"-
"+Integer.toString(day);
}
```

### Ubah Desain Transaksi menjadi seperti berikut :

The screenshot shows a Java Swing window titled "Transaksi Penjualan Barang". It features several input fields for transaction details: "No Jual" (31), "Tgl. Jual" (2018/05/28), "Kode Konsumen" (K01), and "Nama Konsumen" (Jon). Below these are fields for item selection: "B-0001", "buku tulis putih item", "1000.0", "2", and "2000.0". A "Pilih Barang" button is highlighted with a callout labeled "Tombol search brg". A table below shows the item list with columns: Kd Barang, Nama Barang, Harga Barang, Jumlah, and Total. The table contains one row: B-0001, buku tulis putih item, 1000.0, 2, 2000.0. A callout labeled "Text Area" points to the table. At the bottom, there are buttons: "Tambah", "Simpan", "Batal", "Cetak", and "Keluar".

### Tambahkan class berikut :

```
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;

    public PrintingTask(MessageFormat header, MessageFormat footer,
        boolean interactive) {
        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
    protected void done() {
        message(!complete, message);
    }
}
```

**Tambahkan code di method cetak berikut :**

```
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"+"-----";
    ctk=ctk+"\n"+"-----";

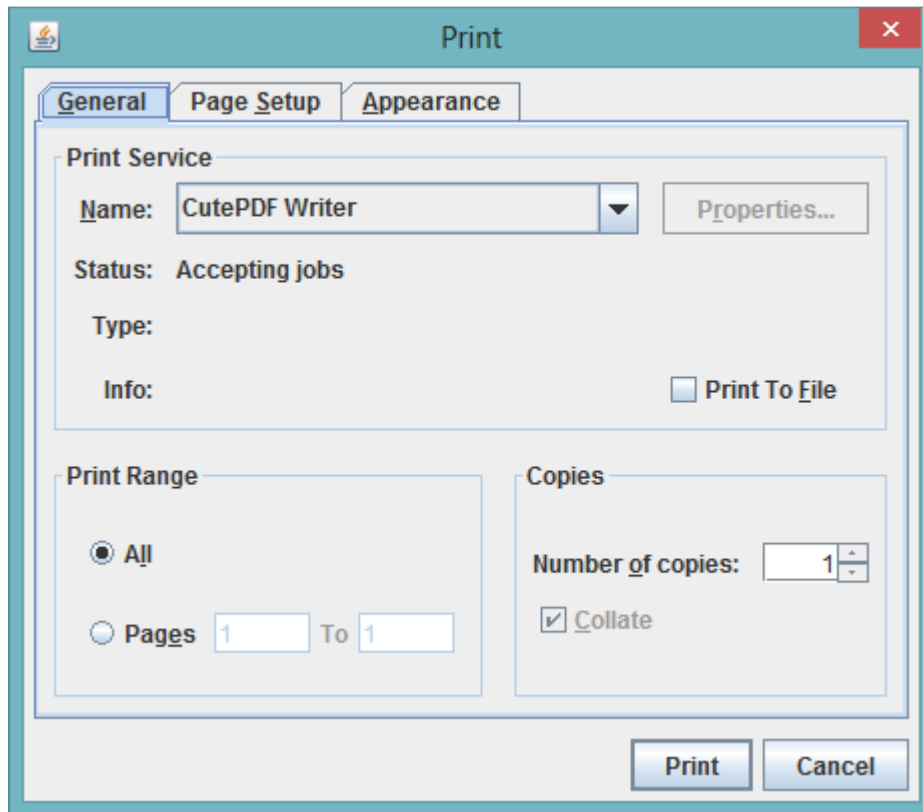
    for(int i=0;i<tblJual.getRowCount();i++)
    {
        String xkd=(String)tblJual.getValueAt(i,0);
        String xnama=(String)tblJual.getValueAt(i,1);
        double xhrg=(Double)tblJual.getValueAt(i,2);
        int xjml=(Integer)tblJual.getValueAt(i,3);
        double xtot=(Double)tblJual.getValueAt(i,4);
        ctk=ctk+"\n"+xkd+"\t"+xnama+"\t"+xhrg+"\t"+xjml+"\t"+xtot;
    }
    ctk=ctk+"\n"+"-----";
    ctk=ctk+"\n"+"-----";

    text.setText(ctk);
    String headerField="";
    String footerField="";
    MessageFormat header = createFormat(headerField);
    MessageFormat footer = createFormat(footerField);
    boolean interactive = true;//interactiveCheck.isSelected();
    boolean background = true;//backgroundCheck.isSelected();

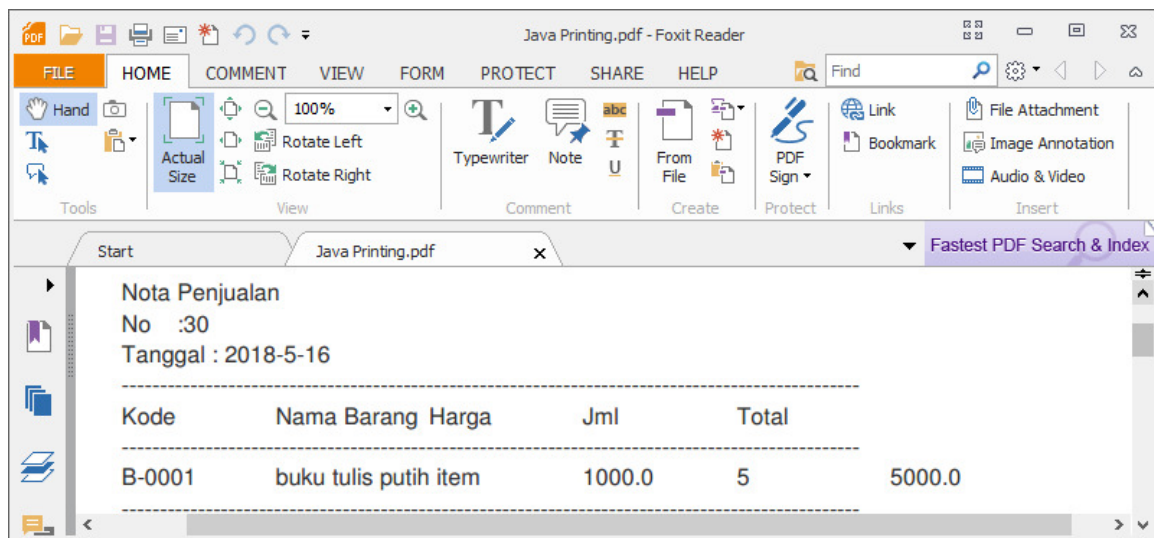
    PrintingTask task = new PrintingTask(header, footer,
interactive);
    if (background) {
        task.execute();
    } else {
        task.run();
    }
}
```



### Dialog Cetak :



### Hasil Cetak :



**Buat Form Cari Barang seperti berikut :**



The screenshot shows a Java Swing window with a title bar and standard OS controls. The window contains a table with the following data:

Kode	Nama Bar...	Satuan	Harga	Harga Beli	Stok	Stok Min
B-0001	buku tulis ...	Buah	1000.0	900.0	196	5
B-0002	Gethuk ma...	Box	1500.0	2000.0	214	10
B-0003	Pensil Lan...	Buah	3000.0	2500.0	98	5
B-0004	Gethuk Lin...	Buah	5000.0	4000.0	100	5
B-0005	Mangga	Buah	10000.0	9000.0	0	10
B-0006	Wingko	Buah	10000.0	9000.0	100	10
B-0007	Mangga	Buah	10000.0	9000.0	100	10
B-0008	HP	Buah	2000000.0	1800000.0	10	2
xxxx	xxxx 3	Buah	1000.0	900.0	5	1

Below the table, there is a text label: **\* Click pada baris untuk produk yang dipilih**

**Tambahkan code berikut di bawah Class :**

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

**Konstruktor :**

```
public frmSelectBarang() {  
    initComponents();  
    baca_data();  
}
```

**Tambahkan Method-method berikut :**

```
private void open_db()  
{  
    try{  
        KoneksiMysql kon = new KoneksiMysql  
            ("127.0.0.1", "root", "", "penjualan");  
        Con = kon.getConnection();  
    }  
    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_brg");

        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_brg");
            dataTable[x][1] = RsBrg.getString("nm_brg");
            dataTable[x][2] = RsBrg.getString("satuan");
            dataTable[x][3] = RsBrg.getDouble("harga");
            dataTable[x][4] = RsBrg.getDouble("harga_beli");
            dataTable[x][5] = RsBrg.getInt("stok");
            dataTable[x][6] = RsBrg.getInt("stok_min");
            x++;
        }
        tblBrg.setModel(new DefaultTableModel(dataTable,header));
    }
    catch(SQLException e)
    {
        JOptionPane.showMessageDialog(null, e);
    }
}

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

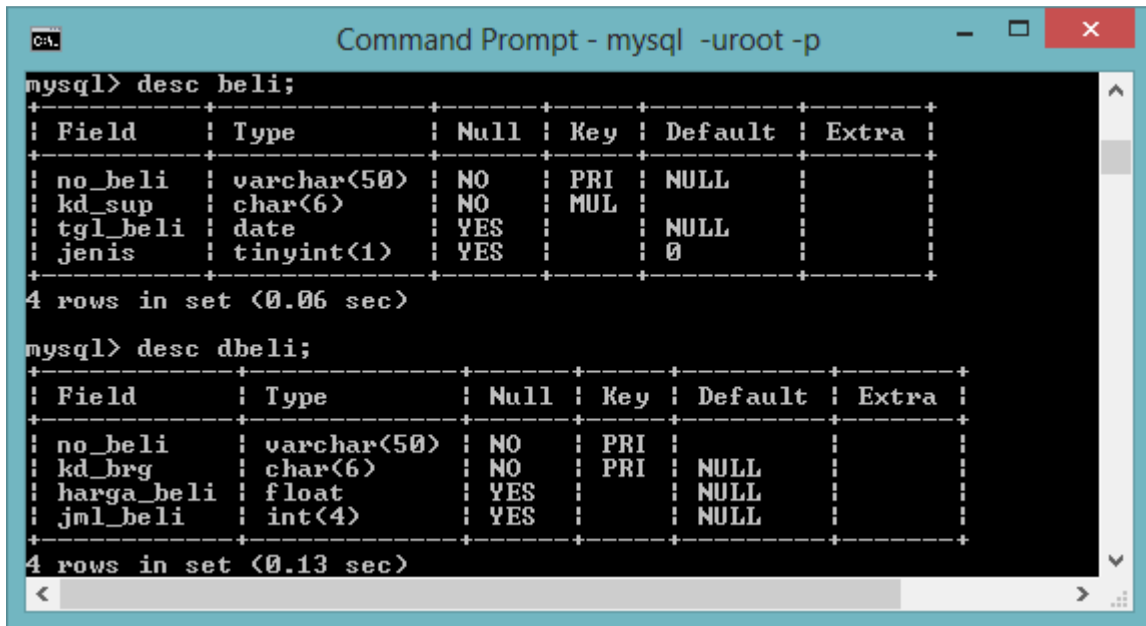
### Tambahkan Method di Form Transaksi :

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

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

### Latihan :

1. Buat form pembelian dengan table seperti berikut :



The screenshot shows a MySQL Command Prompt window with the following content:

```
mysql> desc beli;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| no_beli | varchar(50) | NO | PRI | NULL | |  
| kd_sup | char(6) | NO | MUL | | |  
| tgl_beli | date | YES | | NULL | |  
| jenis | tinyint(1) | YES | | 0 | |  
+-----+-----+-----+-----+-----+-----+  
4 rows in set (0.06 sec)  
  
mysql> desc dbeli;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| no_beli | varchar(50) | NO | PRI | NULL | |  
| kd_brg | char(6) | NO | PRI | NULL | |  
| harga_beli | float | YES | | NULL | |  
| jml_beli | int(4) | YES | | NULL | |  
+-----+-----+-----+-----+-----+-----+  
4 rows in set (0.13 sec)
```

2. Buat form Transaksi sesuai dengan tugas masing-masing.