

Modul Praktikum Pemrograman Visual II (Java II)

Bina Sarana Informatika



Disusun Oleh:
TIM PENYUSUN

KATA PENGANTAR

Segala puji syukur kepada Tuhan Yang Maha Esa karena berkat rahmatNya penulisan modul Mata Kuliah Pemrograman Visual II dapat terselesaikan dengan baik. Modul ini disusun untuk memenuhi kebutuhan mahasiswa dalam mata kuliah Pemrograman Visual II yang disajikan dalam bentuk praktikum dan diharapkan dapat membekali mahasiswa dalam memahami bahasa pemrograman Java lanjutan.

Dengan konsep database dan networking, diharapkan mahasiswa dapat mengimplementasikannya dalam bentuk final project sebagai prototype hasil karya dan syarat kelulusan mata kuliah Pemrograman Visual II. Teknik penyajiannya dilakukan secara terpadu dan sistematis.

Seperti layaknya sebuah modul, maka pembahasan dimulai dengan menjelaskan target pembelajaran yang hendak dicapai. Dengan demikian pengguna modul ini secara mandiri dapat mengukur tingkat ketuntasan yang dicapainya.

Penulis menyadari sepenuhnya bahwa modul ini tentu memiliki banyak kekurangan. Untuk itu penulis dengan lapang dada menerima masukan dan kritik yang konstruktif dari berbagai pihak demi kesempurnaannya di masa yang akan datang. Semoga modul ini dapat bermanfaat bagi para penggunanya.

Jakarta, Februari 2018

Tim Penulis

DAFTAR ISI

	Hal
KATA PENGANTAR.....	2
DAFTAR ISI.....	3
PERTEMUAN 1.....	4
PERTEMUAN 2.....	8
PERTEMUAN 3.....	21
PERTEMUAN 4.....	22
PERTEMUAN 5.....	37
PERTEMUAN 6.....	43
PERTEMUAN 7.....	48
PERTEMUAN 8.....	52
PERTEMUAN 9.....	53
PERTEMUAN 10.....	58
PERTEMUAN 11.....	74
PERTEMUAN 12.....	91
PERTEMUAN 13-16.....	92

PERTEMUAN 1

Penjelasan Kontrak Kuliah dan TUGAS 1 (Pembuatan Database)

1. Penjelasan Kontrak Kuliah

menjelaskan ke siswa untuk mengerjakan sebuah projek program, secara berkelompok (msl. 5 orang/kelompok) atau disesuaikan dengan jumlah mahasiswa per kelasnya.

Ketentuan nilai matakuliah Pemograman Visual II :

20% dari nilai Absen + 25% dari nilai Tugas + 50% dari nilai Project (Quis Project+Final Project)

Ketentuan nilai Tugas :

70% dari tugas yang diberikan oleh dosen + 30% dari nilai kehadiran mahasiswa di kelas

Contoh :

Tugas I : 80, Tugas II : 80, Jumlah pertemuan : 14, jumlah hadir : 10, maka perhitungan nilai tugas adalah :

Jumlah pertemuan : 14

Tugas I	Tugas II	Rata-rata	NT Dosen	Jumlah Hadir	N.Kehadiran	Total Nilai Tugas
80	80	80	56	10	21	77

Presentasi QUIZ Project :

Lingkup presentasi Quis Project yaitu:

1. Pembuatan database yang benar
2. Pembuatan form Login, menu utama, dan Master yang sudah dapat dijalankan
3. Pembuatan desain form transaksi

Presentasi Final Project :

Lingkup presentasi Final Project yaitu:

1. Melanjutkan hasil dari Quis Project yang telah dilakukan
2. Pembuatan form transaksi
3. Pembuatan laporan
4. Pengumpulan Paper Project

Kriteria penilaian projek program**Penilaian secara kelompok QUIZ Project:**

Kriteria Penilaian	Bobot
- Pembuatan Database - Pembuatan Tabel - Kesesuaian tabel dengan kebutuhan	30%
- Pembuatan Form, coding, dan tampilan (pembuatan form, penggunaan objek dan properties, koneksi database, sintaks program dan prosedur, logika rancangan program)	60%
- Cara Presentasi	10%

Penilaian secara kelompok Final PROJECT:

Kriteria Penilaian	Bobot
- Penyajian dan kelengkapan projek secara keseluruhan	30%
- Pembuatan Form, coding, dan report (Pembuatan form, Penggunaan Objek dan Properties, Koneksi Database, Sintaks Program Dan Prosedur, Logika program, dan tampilan report)	60%
- Cara Presentasi	10%

Penilaian Secara Individu (Bobot Nilai : 0-100)

Nilai individu didapat dengan memberikan soal studi kasus kepada mahasiswa untuk mengetahui sejauh mana mahasiswa mampu dan menguasai materi Pemrograman Visual II. Penilaian individu diambil sebelum melakukan presentasi Final Project (Pertemuan 12)

Nilai Rata-rata per individu

Nilai Rata-rata per individu adalah nilai akhir yang didapat dari mahasiswa untuk nilai project.

Nilai Rata-rata per individu = (Total nilai project kelompok + nilai individu)/2

Nilai Quis Project	Nilai Final Project	Nilai Project Kelompok	Nilai Individu	Nilai Akhir Project
80	80	80	50	65

Contoh Project program yang dapat digunakan:

- Pembayaran
- Peminjaman (rental, perpustakaan)
- Penggajian
- Penerimaan (karyawan, siswa, dll)
- Inventori
- Absensi
- Rekam medis
- Dll (jangan gunakan tema penjualan karena sudah digunakan dalam studi kasus)

Ketentuan Penulisan Buku Pemrograman Visual II (Java II)

Final project dijadikan buku. Satu buku terdiri dari 3 kelompok dengan tema yang berbeda (**Nama Dosen Pengajar Nama-nama Mahasiswa Anggota Kelompok, dan Kelas diketik di halaman dalam buku (contoh di halaman pendahuluan), bukan di halaman sampul/cover**). Halaman sampul hanya dituliskan "Pemrograman Visual II dengan Java Net Beans – Tema Projectt"

Contoh: Pemrograman Visual II dengan Java Net Beans – Aplikasi Penjualan, Aplikasi Penggajian, Aplikasi Penyewaan Mobil

- **Catatan: Buku dicetak di percetakan dengan soft cover dan kertas HVS ukuran A5. Untuk teknik penulisan bebas (seperti layaknya buku tutorial; lihat modul untuk gaya penulisan buku tutorial)**

Ketentuan Program Final Project

1. Setiap Kelompok **WAJIB** mengirimkan isi Buku dan file program final project serta Presentasi ke email dosen masing-masing.
2. Dosen wajib mengirimkan 1 project terbaik dari setiap kelas (isi buku dan file program) ke email: **pv2prodimi.bsi@gmail.com**
3. Desain Cover buku berwarna dan bebas

*NB : Menggunakan pakaian putih hitam pada saat presentasi

2. TUGAS 1 (Pembuatan Database)

Dosen memberikan soal yang terdapat pada modul/slide dosen

PERTEMUAN 2

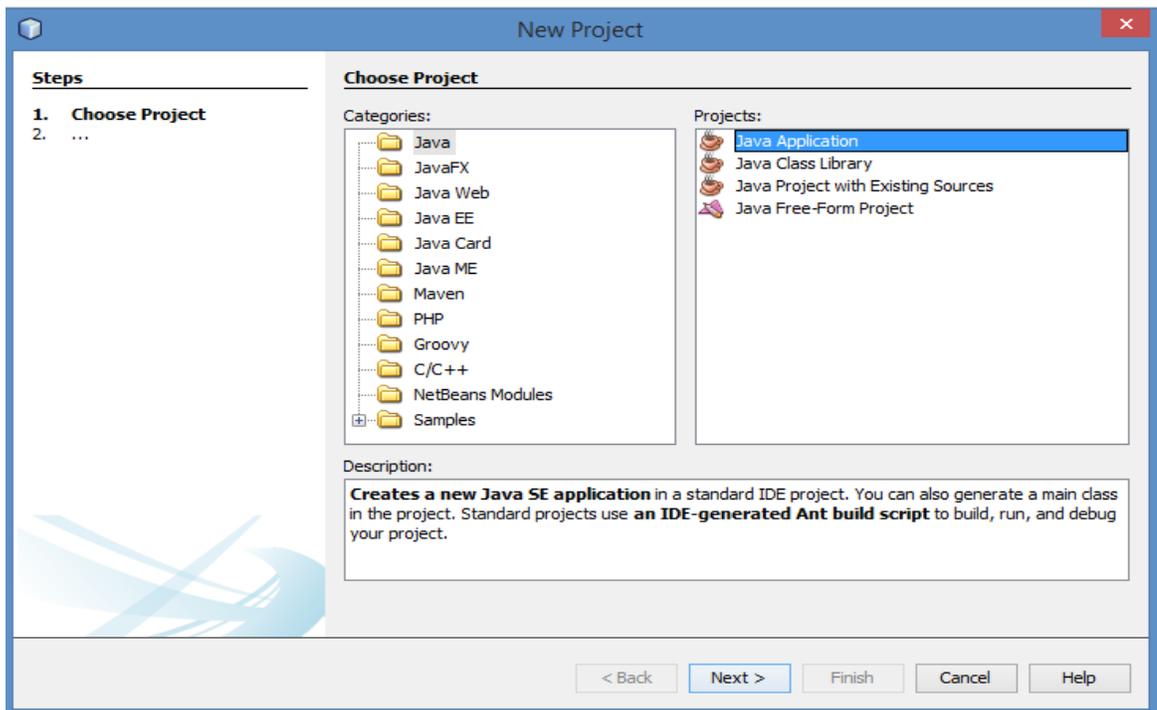
MEMBUAT PROJECT APLIKASI TOKO, MERANCANG CLASS KONEKSI DAN FORM MASTER BARANG

Aplikasi yang dibutuhkan dalam pembelajaran matakuliah Pemrograman Visual II ini adalah:

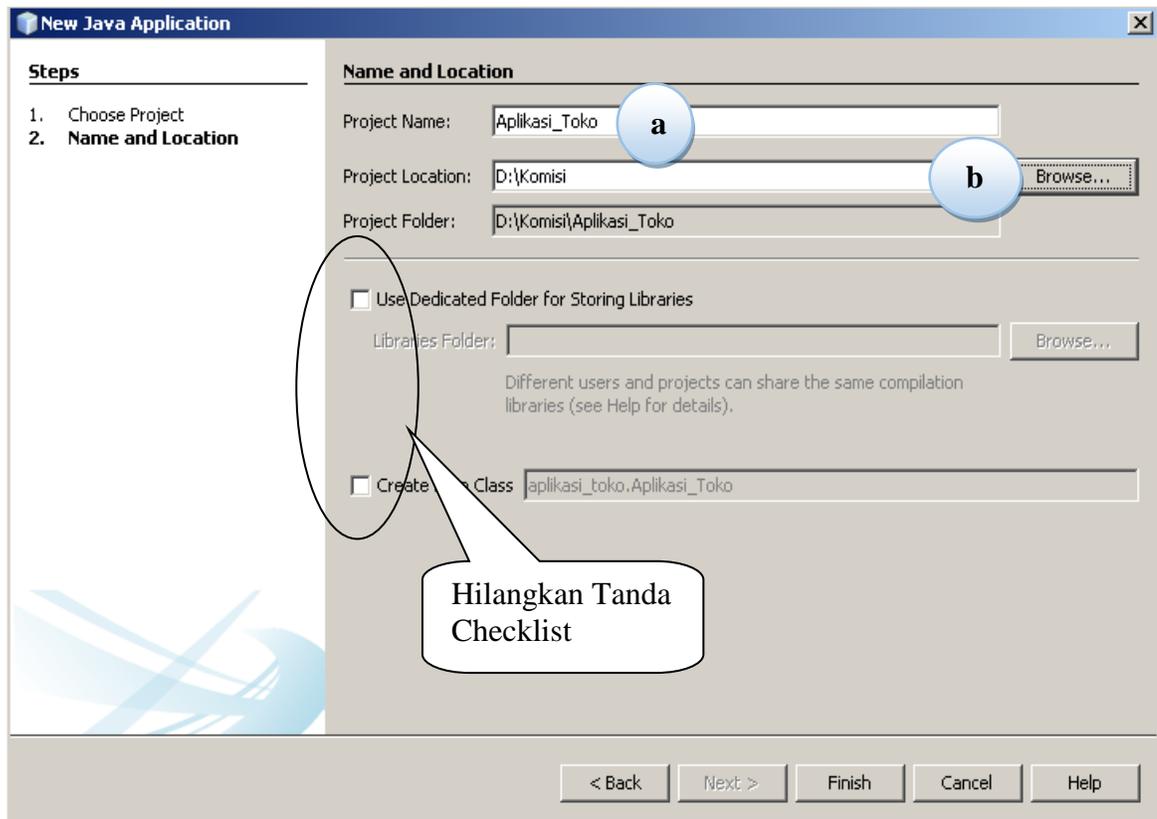
- Netbeans IDE 8.2
- JDK 1.8.0
- Ireport 4.5.0 plugin
- Ireport 4.5.0 jar

A. Membuat Project Aplikasi Toko

1. Buka Program Aplikasi Netbeans, siapkan folder untuk menyimpan project
File → New Project

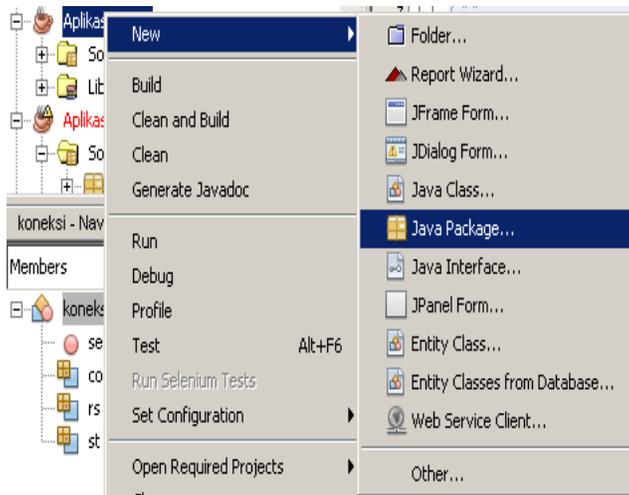


2. Next

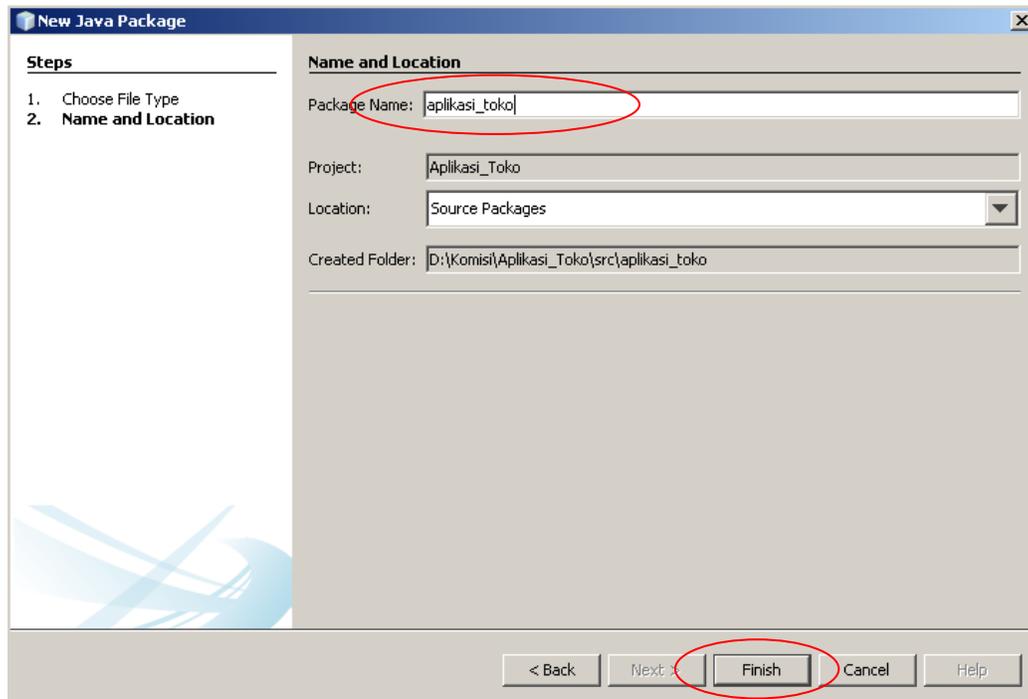


- a. Isi Project Name : **Aplikasi_Toko**
- b. Pilih Tempat Penyimpanan <Browse...>
- c. Finish

3. Membuat Package



- ✓ Klik kanan pada Project – New – JavaPackage
- ✓ Beri Package dengan nama: **aplikasi_toko**
- ✓ Finish



B. Merancang Class Koneksi

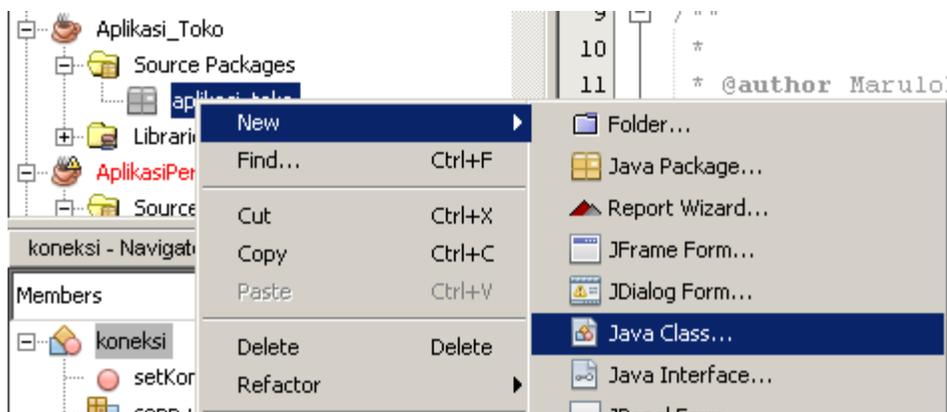
Class koneksi digunakan sebagai penghubung antara database dengan form aplikasi yang kita buat pada netbeans

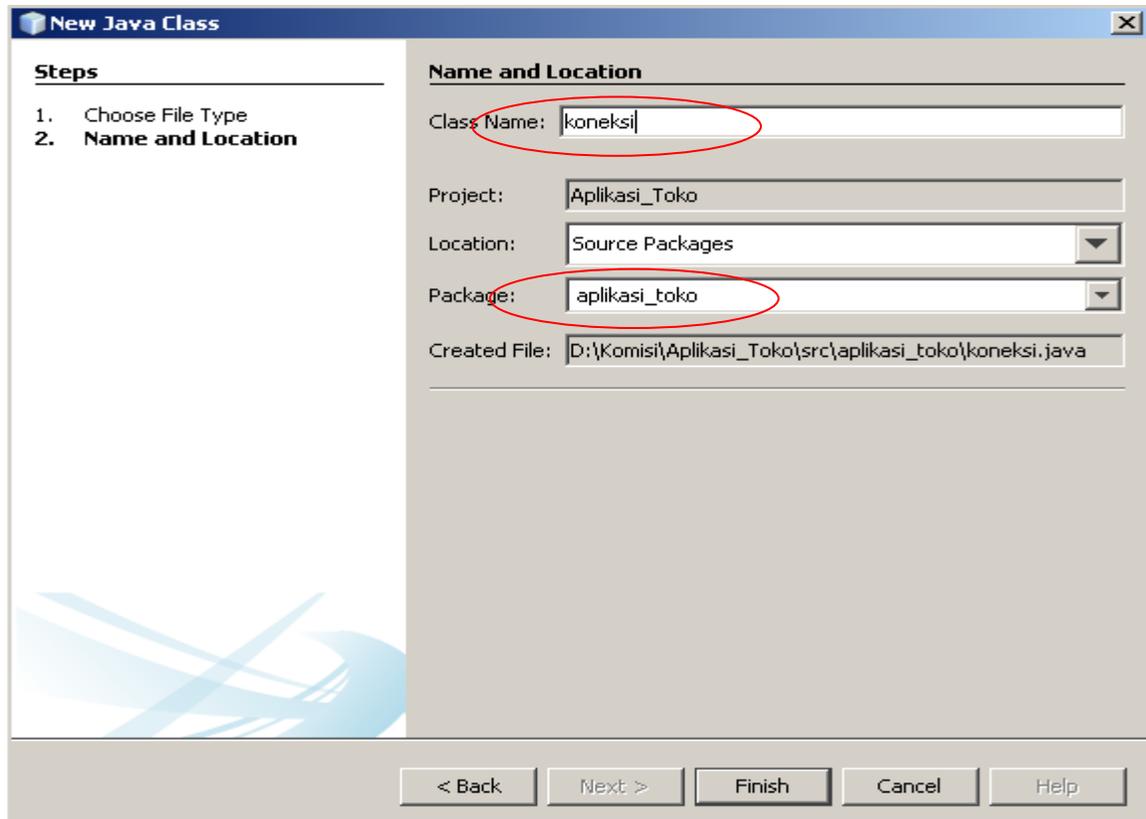
Langkah-langkahnya adalah sebagai berikut:

1. Klik kanan pada package aplikasi_toko – New – Java Class
2. Beri nama class dengan nama : **koneksi**

Package : aplikasi_toko

Finish





3. Ketikkan codingan berikut:

```

package aplikasi_toko;
import java.sql.*;
import javax.swing.JOptionPane;

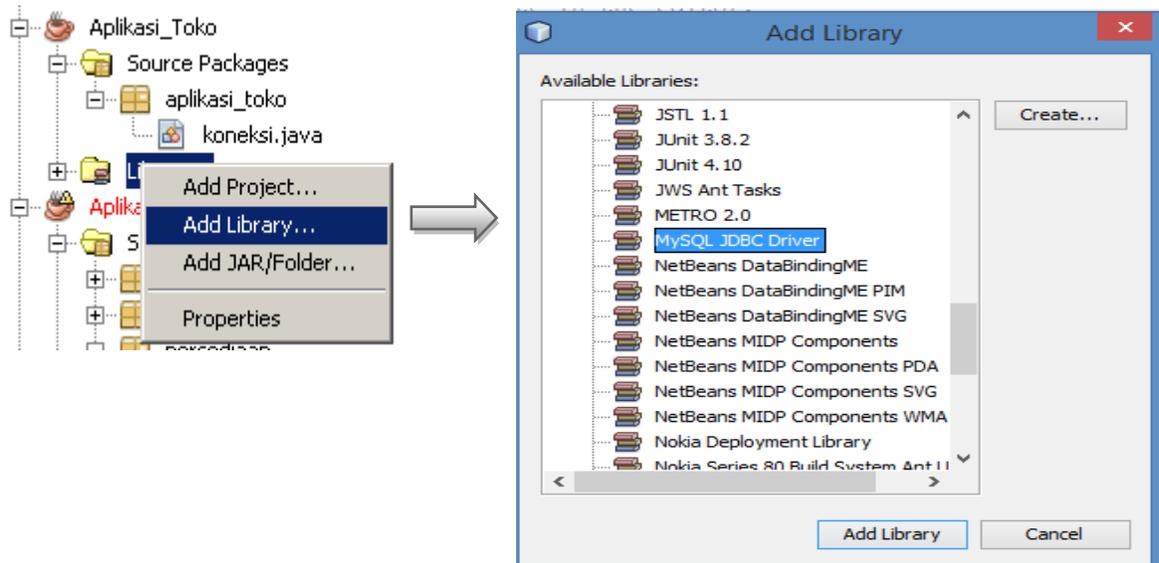
/**
 *
 * @author Maruloh
 */
public class koneksi {
    Connection conn;
    Statement st;
    ResultSet rs;

    public Connection setKoneksi ()
    {
        try{
            Class.forName("com.mysql.jdbc.Driver");
            conn=DriverManager.getConnection("jdbc:mysql://localhost/toko_db","root","");
            st=conn.createStatement();
        }
        catch(Exception e){
            JOptionPane.showMessageDialog(null,"Koneksi Gagal : " +e);
        }
        return conn;
    }
}

```

4. Tambahkan Libraries untuk bisa terhubung ke database

Klik kanan pada Libraries – Add Libraries – Pilih MySQL JDBC Driver – Add Library



C. Membuat Form Master Barang

Desain Form Master Barang

TABEL DATA BARANG

Cari Nama Barang

Title 1	Title 2	Title 3	Title 4

INPUT DATA BARANG

DATA BARANG

Kode Barang

Nama Barang

Harga Jual

Stok

TAMBAH KELUAR

Running Form Master Barang

TABEL DATA BARANG

Cari Nama Barang

Kode Barang	Nama Barang	Harga Jual	Stok
LP15AS	Laptop Asus ...	5000000	10
NT10AS	Netbook Asu...	2750000	3
TV21LG	TV LG 21 inc	3500000	10
TV29SA	TV Samsung ...	4500000	10

INPUT DATA BARANG

DATA BARANG

Kode Barang

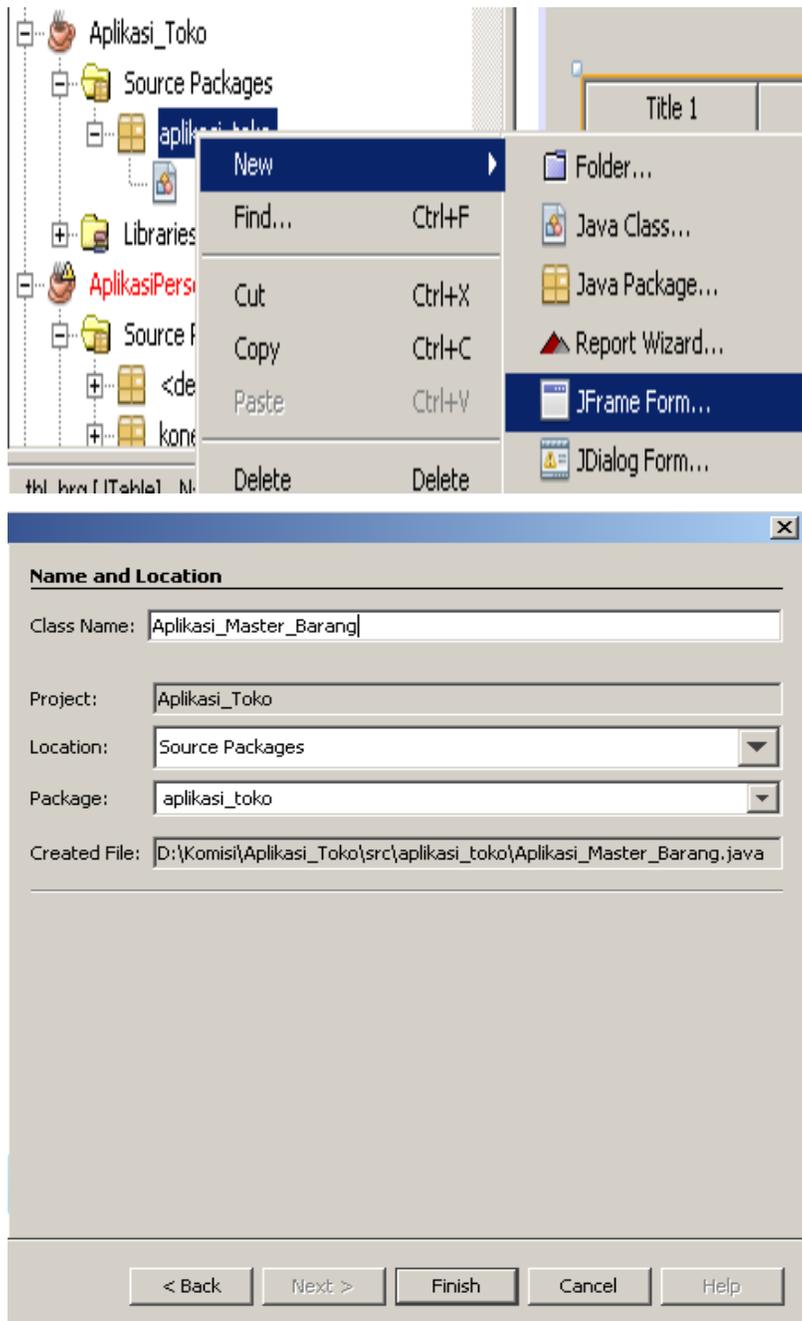
Nama Barang

Harga Jual

Stok

TAMBAH KELUAR

1. Klik kanan pada package aplikasi_toko – New – JFrame Form
2. Isi Class Name : Aplikasi_Master_Barang – Finish



Form Aplikasi_Master_Barang

Spesifikasi properties komponen form Aplikasi_Master_Barang

No	Komponen	Properties
1	JTextField1	Nama Variabel : tkd_brg Text : (kosongkan)
2	JTextField2	Nama Variabel : tnm_brg Text : (kosongkan)
3	JTextField3	Nama Variabel : thrg_jual Text : (kosongkan)
4	JTextField4	Nama Variabel : tstok Text : (kosongkan)
5	JTextField5	Nama Variabel : tcari Text : (kosongkan)
6	JButton1	Nama Variabel : bt_tambah Text : TAMBAH
7	JButton2	Nama Variabel : bt_keluar Text : KELUAR
8	JTable1	Nama Variabel : tbl_brg

Logika Jalannya Program:

1. Pertama run semua textfield nonaktif, data barang tampil di grid. INPUT DAN TUTUP aktif
2. Tekan INPUT semua textfield Aktif. Tombol INPUT berubah menjadi SIMPAN, tombol TUTUP menjadi BATAL
3. Ketikan kode barang. Jika sudah ada tampil nama ,stok,harga. Kd barang nonaktif. SIMPAN berubah jadi UPDATE. Lakukan perubahan data barang untuk menyimpan tekan UPDATE, tekan BATAL untuk membatalkan perubahan data
4. Ketikan kode barang. Jika tidak ada, Cursor pindah ke nama barang. Isikan data barang baru, tekan SIMPAN untuk menyimpan
5. Untuk menghapus data barang, silakan pilih pada tabel data mana yang akan dihapus, setelah itu tekan tombol Backspace untuk menghapusnya
6. Tekan TUTUP untuk keluar dari program

1. Ketik Script method-method dibawah ini pada tampilan listing:

```

package aplikasi_toko;
import java.awt.event.KeyEvent;
import java.sql.*;
import javax.swing.*;
import javax.swing.table.DefaultTableModel;

/**
 *
 * @author Maruloh
 */
public class Aplikasi_Master_Barang extends javax.swing.JFrame {
koneksi kon = new koneksi();
private Object [][] databarang=null;
private String[] label={"Kode Barang","Nama Barang","Harga Jual","Stok"};

/**
 * Creates new form Aplikasi_Master_Barang
 */
public Aplikasi_Master_Barang() {
    initComponents();
    kon.setKoneksi();
    BacaTabelBarang();
}

private void BacaTabelBarang()
{
    try{
        String sql="Select *From barang order by kd_brg";
        kon.rs=kon.st.executeQuery(sql);
        ResultSetMetaData m=kon.rs.getMetaData();
        int kolom=m.getColumnCount();
        int baris=0;
        while(kon.rs.next()){
            baris=kon.rs.getRow();
        }
        databarang=new Object[baris][kolom];
        int x=0;
        kon.rs.beforeFirst();
        while(kon.rs.next()){
            databarang[x][0]=kon.rs.getString("kd_brg");
            databarang[x][1]=kon.rs.getString("nm_brg");
            databarang[x][2]=kon.rs.getString("hrg_jual");
            databarang[x][3]=kon.rs.getString("stok");
            x++;
        }
        tbl_brg.setModel(new DefaultTableModel(databarang, label));
    }

    catch(SQLException e) {
        JOptionPane.showMessageDialog(null, e);
    }
}
}

```

```

private void BacaTabelProdukCari ()
{
    try{
        String sql="Select *From barang where nm_brg like '%" +tcari.getText()+ "%' ";
        kon.rs=kon.st.executeQuery(sql);
        ResultSetMetaData m=kon.rs.getMetaData();
        int kolom=m.getColumnCount();
        int baris=0;
        while(kon.rs.next()){
            baris=kon.rs.getRow();
        }
        databarang=new Object[baris][kolom];
        int x=0;
        kon.rs.beforeFirst();
        while(kon.rs.next()){
            databarang[x][0]=kon.rs.getString("kd_brg");
            databarang[x][1]=kon.rs.getString("nm_brg");
            databarang[x][2]=kon.rs.getString("hrg_jual");
            databarang[x][3]=kon.rs.getString("stok");
            x++;
        }
        tbl_brg.setModel(new DefaultTableModel(databarang,label));
    }

    catch(SQLException e) {
        JOptionPane.showMessageDialog(null, e);
    }
}

```

```

private void SetTabel ()
{
    int row=tbl_brg.getSelectedRow();
    tkd_brg.setText ((String) tbl_brg.getValueAt(row, 0));
    tnm_brg.setText ((String) tbl_brg.getValueAt(row, 1));
    thrg_jual.setText ((String) tbl_brg.getValueAt(row, 2));
    tstok.setText ((String) tbl_brg.getValueAt(row, 3));
}

private void Bersih ()
{
    tkd_brg.setText ("");
    tnm_brg.setText ("");
    thrg_jual.setText ("");
    tstok.setText ("");
}

private void aktif ()
{
    tkd_brg.setEnabled(true);
    tnm_brg.setEnabled(true);
    thrg_jual.setEnabled(true);
    tstok.setEnabled(true);
}

```

```
private void nonaktif ()
{
    tkd_brg.setEnabled(false);
    tnm_brg.setEnabled(false);
    thrg_jual.setEnabled(false);
    tstok.setEnabled(false);
}

private void SimpanData ()
{
    try{
        String sql="insert into barang values ('"+tkd_brg.getText()+"', "
            + "'"+tnm_brg.getText()+"', '"+thrg_jual.getText()+"', "
            + "'"+tstok.getText()+"')";
        kon.st.executeUpdate(sql);
        JOptionPane.showMessageDialog(null, "Data berhasil disimpan");
        Bersih();
        BacaTabelBarang();
    }
    catch(SQLException e)
    {
        JOptionPane.showMessageDialog(null, e);
    }
}
```

```
private void UpdatetData ()
{
    try{
        String sql="update barang set kd_brg='"+tkd_brg.getText()+"', "
            + "nm_brg='"+tnm_brg.getText()+"', "
            + "hrg_jual='"+thrg_jual.getText()+"', "
            + "stok='"+tstok.getText()+"' where kd_brg='"+tkd_brg.getText()+"'";
        kon.st.executeUpdate(sql);
        JOptionPane.showMessageDialog(null, "Data berhasil diedit");
        Bersih();
        BacaTabelBarang();
    }
    catch(SQLException e)
    {
        JOptionPane.showMessageDialog(null, e);
    }
}
```

```

private void HapusData()
{
    try{
        String sql="delete from barang where kd_brg='"+tkd_brg.getText()+"'";
        kon.st.executeUpdate(sql);
        JOptionPane.showMessageDialog(null,"Data berhasil dihapus");
        Bersih();
        BacaTabelBarang();
    }
    catch(SQLException e)
    {
        JOptionPane.showMessageDialog(null, e);
    }
}

```

2. Klik kanan pada JFrame, pilih Event – Window – Window Activated, ketikkan Script Berikut: (kondisi saat form dijalankan)

```

private void formWindowActivated(java.awt.event.WindowEvent evt) {
    // TODO add your handling code here:
    nonaktif();
    BacaTabelBarang();
}

```

3. Klik kanan pada bt_keluar, pilih Event – Action – Action Performed, ketikkan Script Berikut:

```

private void bt_keluarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    if(bt_keluar.getText().equals("KELUAR"))
    {
        dispose();
        //Form_menu_Utama tampilin
    }
    else if(bt_keluar.getText().equals("BATAL"))
    {
        Bersih();
        nonaktif();
        bt_keluar.setText("KELUAR");
        bt_tambah.setText("TAMBAH");
    }
}

```

4. Klik kanan pada `bt_tambah`, pilih Event – Action – Action Performed, ketikkan Script Berikut:

```
private void bt_tambahActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    if(bt_tambah.getText().equals("TAMBAH"))
    {
        bt_tambah.setText("SIMPAN");
        bt_keluar.setText("BATAL");
        Bersih();
        aktif();
        tkd_brg.requestFocus();
    }
    else if(bt_tambah.getText().equals("SIMPAN"))
    {
        SimpanData();
        BacaTabelBarang();
        bt_tambah.setText("TAMBAH");
        bt_keluar.setText("KELUAR");
        Bersih();
        nonaktif();
    }
    else if(bt_tambah.getText().equals("UPDATE"))
    {
        UpdatetData();
        BacaTabelBarang();
        bt_tambah.setText("TAMBAH");
        bt_keluar.setText("KELUAR");
        Bersih();
        nonaktif();
    }
}
}
```

5. Klik kanan pada `tkd_brg`, pilih Event – key – KeyPressed, ketikkan Script Berikut:

```
private void tkd_brgKeyPressed(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if(evt.getKeyCode()==KeyEvent.VK_ENTER) {
        try {
            String sql = "Select *from barang where kd_brg='" + tkd_brg.getText() + "'";
            kon.rs = kon.st.executeQuery(sql);
            if (kon.rs.next()) {
                tkd_brg.setEnabled(false);
                tkd_brg.setText(kon.rs.getString("kd_brg"));
                tnm_brg.setText(kon.rs.getString("nm_brg"));
                thrj_jual.setText(kon.rs.getString("hrj_jual"));
                tstok.setText(kon.rs.getString("stok"));
                bt_tambah.setText("UPDATE");
            } else {
                tnm_brg.requestFocus();
            }
        } catch (SQLException e) {
            JOptionPane.showMessageDialog(null, e);
        }
    }
}
}
```

6. Klik kanan pada `tnm_brg`, dan `thrg_jual`, pilih Event – key – KeyPressed, ketikan Script Berikut: (memindahkan posisi kursor ke object tertentu saat dienter)

```

private void tnm_brgKeyPressed(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if (evt.getKeyCode() == KeyEvent.VK_ENTER) {
        thrg_jual.requestFocus();
    }
}

private void thrg_jualKeyPressed(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if (evt.getKeyCode() == KeyEvent.VK_ENTER) {
        tstok.requestFocus();
    }
}

```

7. Klik kanan pada `tbl_brg`, pilih Event – Mouse – MouseClicked, ketikan Script Berikut: (memanggil method `SetTabel` yang berfungsi untuk menampilkan nilai-nilai pada tabel ke dalam form)

```

private void tbl_brgMouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    SetTabel();
}

```

8. Klik kanan pada `tbl_brg`, pilih Event – Key – KeyPressed, ketikan Script Berikut: (memanggil method `HapusData` untuk menghapus data pada tabel terpilih pada saat ditekan tombol Backspace)

```

private void tbl_brgKeyPressed(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if (evt.getKeyCode() == KeyEvent.VK_BACK_SPACE)
    {
        HapusData();
    }
}

```

9. Klik kanan pada `tcari`, pilih Event – Key – KeyTyped, ketikan Script Berikut:

```

private void tcariKeyTyped(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    kon.setKoneksi();
    BacaTabelProdukCari();
}

```

PERTEMUAN 3
TUGAS 2 MEMBUAT FORM USER

Mahasiswa mengerjakan Form Master User, Dosen menilai sebagai Nilai Tugas 2.

Beri nama form tersebut dengan nama : **Aplikasi_Master_User**

Dosen memberikan tampilan form yang ada pada modul/slide dosen

PERTEMUAN 4

MEMBUAT FORM MENU UTAMA DAN FORM LOGIN

A. Membuat Form Menu Utama

Dalam studi kasus penjualan tunai ini, penulis menggunakan 2 akses level, yaitu level sebagai admin, dan akses sebagai kasir. Adapun ruang lingkup akses level adalah sebagai berikut:

Akses Level	Ruang Lingkup
Admin	Master (Barang dan User) Transaksi Laporan (Barang, User, Transaksi) Utility (Ganti Password)
Kasir	Transaksi Laporan (Barang, Transaksi) Utility (Ganti Password)

Untuk membedakan akses level tersebut, salah satu cara yang bisa dilakukan adalah dengan membuat 2 form menu utama. Menu pertama untuk Admin, menu lainnya untuk level kasir.

Akses Level Admin

Akses Level Kasir

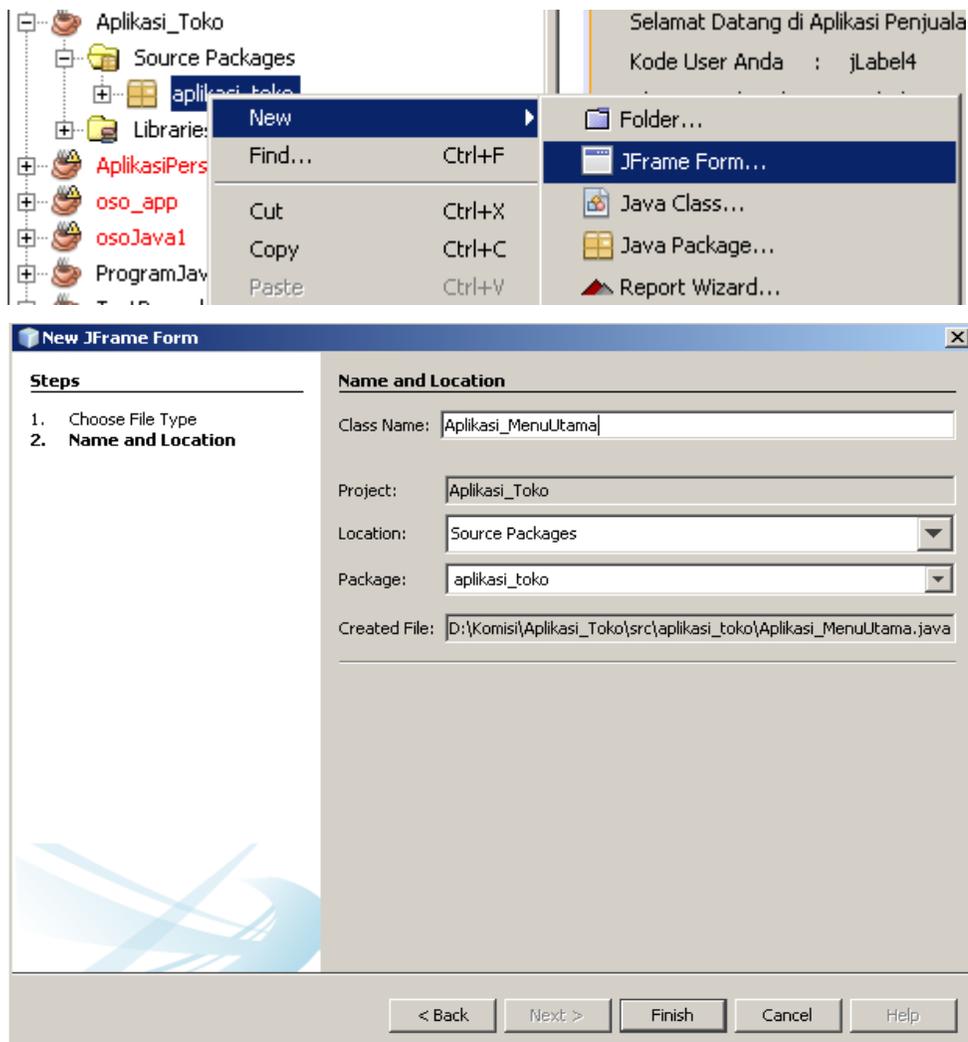
Membuat Menu Utama Level Admin

Desain Tampilan Level Admin:

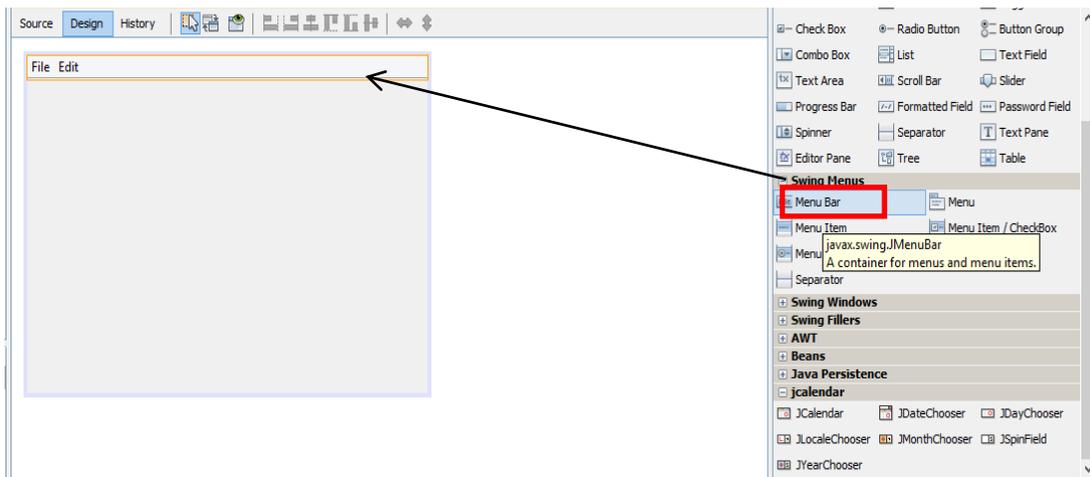
File	Master	Transaksi	Laporan	Utility
Logout	Barang		Data Barang	Ganti Password
Keluar	User		Data Pengguna	
			Data Transaksi	

Langkah-langkah dalam pembuatan Menu utama untuk level Admin adalah sebagai berikut:

1. Klik kanan pada package aplikasi_toko – New – JFrame Form
2. Isi Class Name : Aplikasi_MenuUtama – Finish

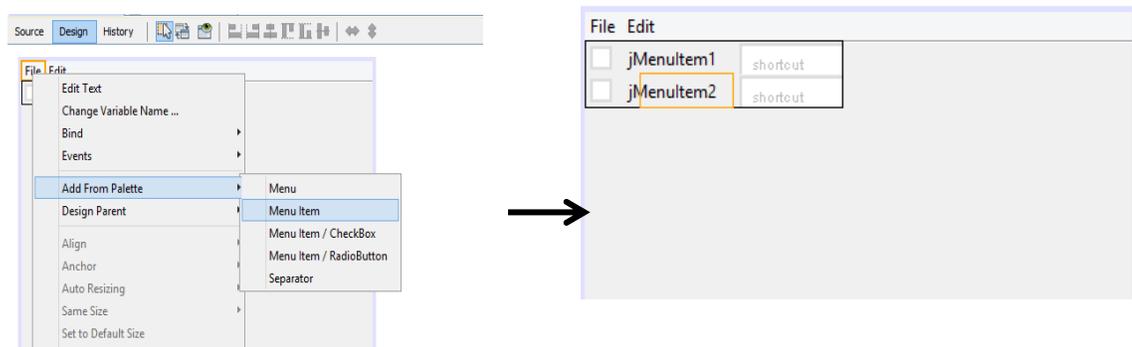


3. Tambahkan Komponen JMenu Bar pada Form



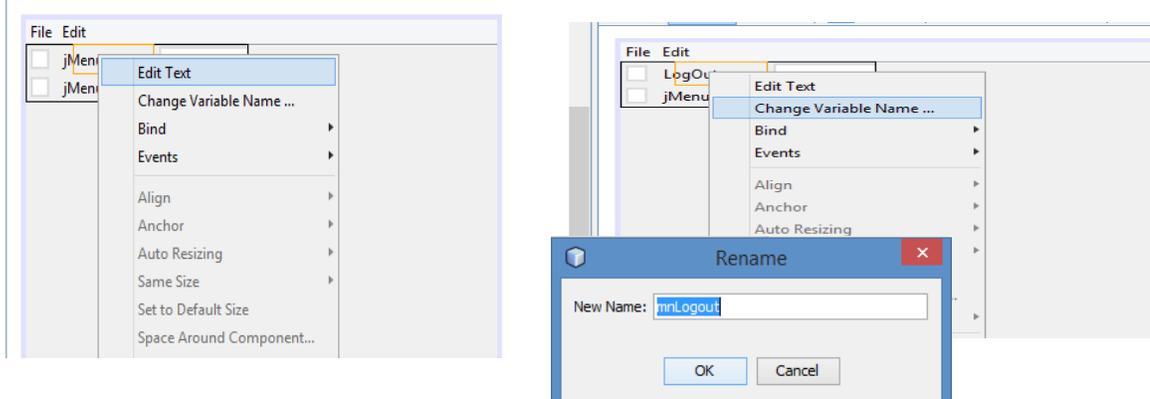
4. Tambahkan 2 Menu Item sebagai sub menu, dengan cara:

Klik kanan pada menu File – Add From Palette – Menu Item



5. Klik Kanan pada JMenuItem1 – Edit Text – Ketik **LogOut**

6. Klik kanan pada JMenuItem1 – Change Variable Name – rename menjadi **mnLogout**

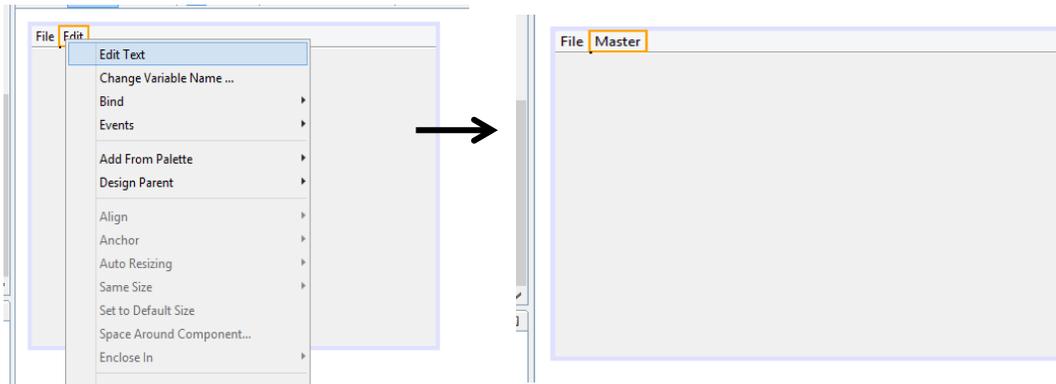


7. Lakukan langkah 7 dan 8 dengan mengatur JMenuItem2 menjadi :

Edit Text = **Keluar**

Change Variable Name = **mnKeluar**

8. Klik kanan pada menu Edit – Edit Text – beri nilai **Master**



9. Tambahkan 2 Menu Item dengan cara yang sama seperti langkah no.4, dengan pengaturan sebagai berikut:

JMenuItem1, Edit Text = **Barang**, Change Variable Name = **mnBarang**

JMenuItem2, Edit Text = **User**, Change Variable Name = **mnUser**



10. Untuk membuat menubar baru (Transaksi) silakan klik kanan pada **JMenuBar1** –

Add Menu



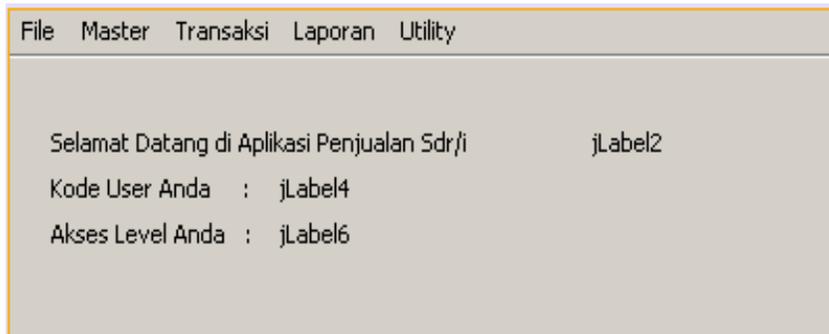
Silakan atur jMenuItem3 sebagai berikut:

Edit Text = **Transaksi**, Change Variable Name = **mnTransaksi**

11. Lakukan hal yang sama seperti langkah di atas untuk membuat menu Laporan dan Utility, dengan ketentuan berikut:

Komponen		Properties
jMenu3		Edit Text : Laporan
	jMenuItem1	Edit Text : Data Barang Change Variable Name : mnLapBarang
	jMenuItem2	Edit Text : Data Pelanggan Change Variable Name : mnlapbarang
	jMenuItem2	Edit Text : Data Barang Change Variable Name : mnlapuser
jMenu4		Edit Text : Utility
	jMenuItem1	Edit Text : Ganti Passord Change Variable Name : mnGantiPassword

12. Tambahkan komponen JLabel ke dalam Form Menu Utama Admin seperti di bawah ini, untuk jLabel2, jLabel4, dan jLabel6 biarkan untuk Text dan sedangkan untuk nama variabelnya beri nama masing-masing **lnama**, **lkode**, **llevel**.



13. Hasil Akhir dari Menu Utama Level Admin



14. Ketikkan script koneksi dan pembuatan Variabel seperti dibawah ini:

```

package aplikasi_toko;
import java.util.HashMap;
// Paket JasperReports
import javax.swing.JOptionPane;
import net.sf.jasperreports.engine.JasperFillManager;
import net.sf.jasperreports.engine.JasperPrint;
import net.sf.jasperreports.view.JasperViewer;

/**
 *
 * @author Maruloh
 */
public class Aplikasi_MenuUtama extends javax.swing.JFrame {
    koneksi kon = new koneksi();

    public Aplikasi_Ganti_Password gantipass=null;

    /**
     * Creates new form Aplikasi_MenuUtama
     */
    public Aplikasi_MenuUtama() {
        initComponents();
        kon.setKoneksi();
    }

```

Digunakan untuk memanggil laporan, ditulis setelah melakukan **import library jasperreport** (pert.10), jika belum, script akan error

```

public String KodeUser;
public String Level;
public String NamaUser;
public String getKodeUser()
{
    return KodeUser;
}
public String getNamaUser()
{
    return NamaUser;
}
public String getLevel()
{
    return Level ;
}

```

Digunakan untuk memanggil nilai yang diinput pada form login

15. Klik kanan pada jForm, pilih Event – Window – windowActivated, ketik script

```

private void formWindowActivated(java.awt.event.WindowEvent evt) {
    // TODO add your handling code here:
    lnama.setText(NamaUser);
    lkode.setText(KodeUser);
    llevel.setText(Level);
}

```

Script diatas berfungsi untuk memberikan nilai lnama, lkode, llevel dari variabel NamaUser, KodeUser, dan Level yang diambil dari Form Login.

16. Klik kanan pada **mnLogout** , pilih **Event – Action – ActionPerformed**, ketik script

```

private void mnLogoutActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    this.dispose();
    Aplikasi_Login login = new Aplikasi_Login();
    login.setLocationRelativeTo(null);
    login.setVisible(true);
}

```

Script diatas berfungsi untuk memanggil Form Aplikasi_Login sekaligus menutup form Aplikasi_MenuUtama.

17. Klik Kanan pada **mnKeluar** , pilih **Event – Action – ActionPerformed**, ketik Script

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

18. Klik Kanan pada **mnBarang** , pilih **Event – Action – ActionPerformed**, ketik Script

```
private void mnBarangActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    Aplikasi_Master_Barang barang = new Aplikasi_Master_Barang();
    barang.setLocationRelativeTo(null);
    barang.setVisible(true);
}
```

Script diatas berfungsi untuk memanggil form Aplikasi_Master_Barang

19. Klik Kanan pada **mnUser** , pilih **Event – Action – ActionPerformed**, ketik script

```
private void mnUserActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    Aplikasi_Master_User user = new Aplikasi_Master_User();
    user.setLocationRelativeTo(null);
    user.setVisible(true);
}
```

Script diatas berfungsi untuk memanggil form Aplikasi_Master_User

20. Klik Kanan pada **mntransaksi** , pilih **Event – Mouse – MouseClicked**, ketik script

```
private void mnTransaksiMouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    Aplikasi_Transaksi transaksi = new Aplikasi_Transaksi();
    transaksi.setLocationRelativeTo(null);
    transaksi.setVisible(true);
    transaksi.KodeUser=lkode.getText();
}
```

Script diatas berfungsi untuk memanggil form Aplikasi_Transaksi, dan memberikan nilai pada variabel KodeUser yang diambil dari komponen lkode untuk ditampilkan di Form Aplikasi_Transaksi.

21. Klik Kanan pada **mnLapBarang** , pilih **Event – Action – ActionPerformed**, ketik script

```
private void mnLapbarangActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    try{  
        String path="src/aplikasi_toko/Laporan_Barang.jasper";  
        HashMap parameter = new HashMap();  
        JasperPrint print = JasperFillManager.fillReport(path,parameter, kon.setKoneksi());  
        JasperViewer.viewReport(print, false);  
    }  
    catch(Exception e) {  
        JOptionPane.showMessageDialog(null,e);  
    }  
}
```

Script diatas digunakan untuk memanggil laporan barang. Ketikkan script diatas jika anda **telah membuat file Laporan_Barang.jasper** dan **mengimport file jasperreport.jar**

22. Klik Kanan pada **mnLapUser** , pilih **Event – Action – ActionPerformed**, ketik script

```
private void mnLapUserActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    try{  
        String path="src/aplikasi_toko/Laporan_User.jasper";  
        HashMap parameter = new HashMap();  
        JasperPrint print = JasperFillManager.fillReport(path,parameter, kon.setKoneksi());  
        JasperViewer.viewReport(print, false);  
    }  
    catch(Exception e) {  
        JOptionPane.showMessageDialog(null,e);  
    }  
}
```

Script diatas digunakan untuk memanggil laporan user. Ketikkan script diatas jika anda **telah membuat file Laporan_User.jasper** dan **mengimport file jasperreport.jar**

23. Klik Kanan pada **mnLapTransaksi** , pilih **Event – Action – ActionPerformed**, ketik script

```
private void mnLapTransaksiActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    Aplikasi_Laporan_Penjualan laporan_Jual = new Aplikasi_Laporan_Penjualan();
    laporan_Jual.setLocationRelativeTo(null);
    laporan_Jual.setVisible(true);
}
```

Script diatas berfungsi untuk memanggil form **Aplikasi_Laporan_Penjualan**. Ketikkan Script diatas jika anda telah membuat form **Aplikasi_Laporan_Penjualan**.

24. Klik Kanan pada **mnGantiPassword** , pilih **Event – Action – ActionPerformed**, ketik script

```
private void mnGantiPasswordActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    Aplikasi_Ganti_Password gantipass = new Aplikasi_Ganti_Password();
    gantipass.setLocationRelativeTo(null);
    gantipass.setVisible(true);
    gantipass.Kode_User=lkode.getText();
}
```

Script diatas berfungsi untuk memanggil form **Aplikasi_Ganti_Password**. Ketikkan Script diatas jika anda telah membuat form **Aplikasi_Ganti_Password**

Membuat Menu Utama Level Kasir

Untuk membuat Menu Utama Level Kasir silakan anda ikuti langkah-langkah seperti membuat menu utama level admin. Rancangan menu untuk level kasir adalah sebagai berikut:

File	Transaksi	Laporan	Utility
Logout		Data Barang	Ganti Password
Keluar		Data Transaksi	

Beri nama untuk form tersebut : **Aplikasi_MenuUtama_Kasir**

B. Membuat Form Login

Desain Form Login



The design of the login form is a rectangular window with a light gray background. At the top center, the word "LOGIN" is written in a large, bold, black font. Below the title, there are two input fields: the first is labeled "Kode User" and the second is labeled "Password". At the bottom of the form, there are two buttons: "LOGIN" on the left and "KELUAR" on the right.

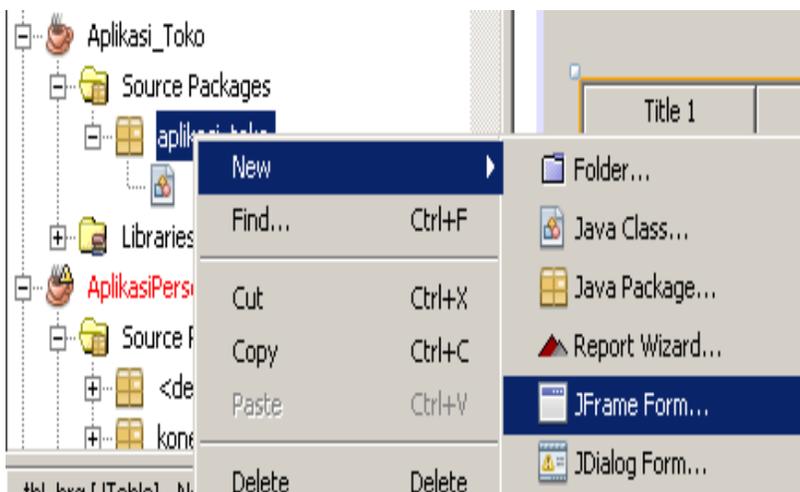
Running Form Login



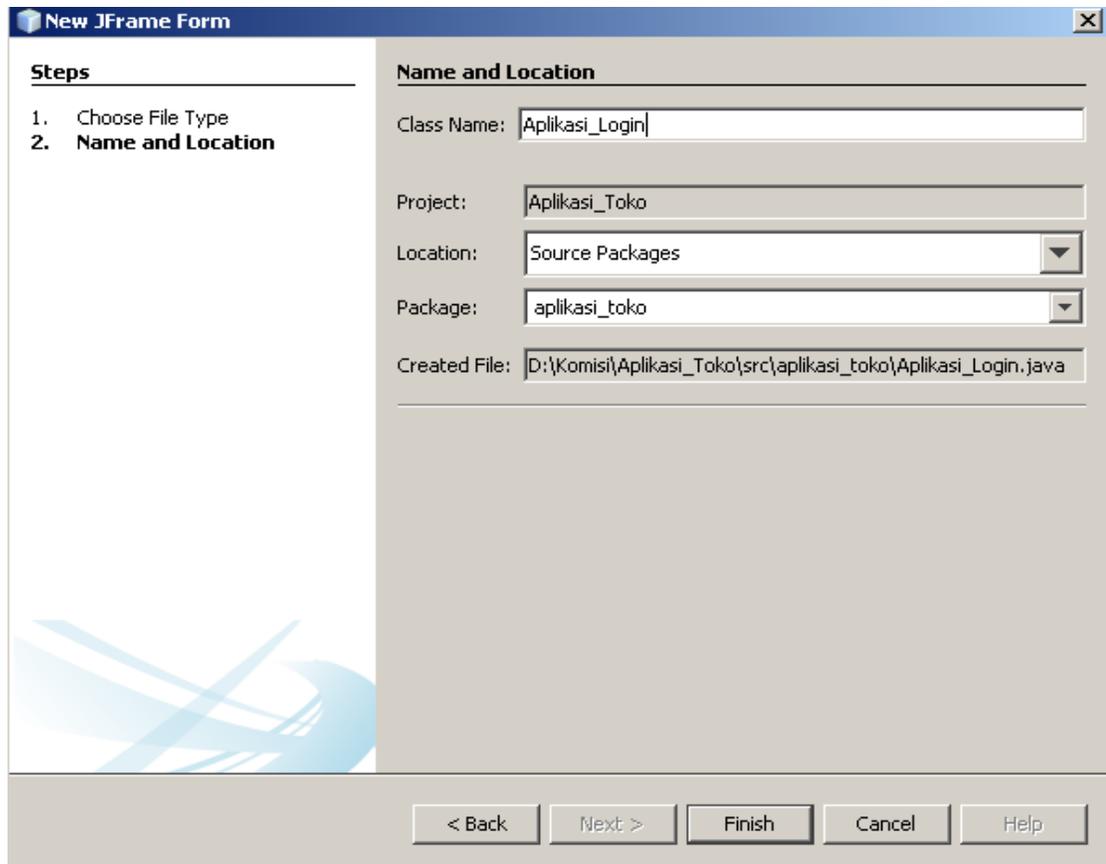
The running login form is shown as a window with a standard Windows-style title bar (minimize, maximize, close buttons). The content is identical to the design shown in the previous image, with the "LOGIN" title, "Kode User" and "Password" input fields, and "LOGIN" and "KELUAR" buttons.

Langkah-langkah untuk membuat form login adalah sebagai berikut:

1. Klik kanan pada package aplikasi_toko – New – JFrame Form



2. Isi Class Name : Aplikasi_Login– Finish



Mendesain Form Login

Spesifikasi properties komponen form Aplikasi_Login

No	Komponen	Properties
1	JTextField1	Nama Variabel : tkodeuser Text : (kosongkan)
2	JTextField2	Nama Variabel : tpassword Text : (kosongkan)
3	JTextField3	Nama Variabel : tlevel Text : (kosongkan)
4	JButton1	Nama Variabel : bt_tambah Text : TAMBAH
5	JButton2	Nama Variabel : bt_keluar Text : KELUAR

Logika jalannya Program:

1. Saat form dijalankan textField yang aktif hanya kode user.
2. Kode user diinput, jika kode user ditemukan, maka textField akan aktif dan password diisi. Jika kode user salah, maka akan muncul pesan bahwa kode user salah
3. Tombol login untuk memeriksa username dan password, jika benar maka akan masuk ke form menu utama. Ada 2 kondisi saat masuk ke form menu utama. Jika level 1, maka akan masuk ke menu utama admin, jika level 2 maka akan masuk ke menu utama kasir. jika Username atau password salah akan muncul pesan bahwa kode user atau password salah.

Script program untuk form Aplikasi_Login:

1. Tambahkan script koneksi pada jendela coding, dan method-method di bawah ini:

```

package aplikasi_toko;
import java.sql.*;
import javax.swing.JOptionPane;

/**
 *
 * @author Maruloh
 */
public class Aplikasi_Login extends javax.swing.JFrame {
    koneksi kon=new koneksi();
    public Aplikasi_MenuUtama menu=null;

    /**
     * Creates new form Aplikasi_Login
     */
    public Aplikasi_Login() {
        initComponents();
        kon.setKoneksi();
        tlevel.setVisible(false);
    }

    void bersih(){
        tkodeuser.setText("");
        tpassword.setText("");
        tlevel.setText("");
    }
}

```

```

private void prosesLogin(){
    String kodeuser="";
    String password="";
    kodeuser=tkodeuser.getText();
    try(String sql="Select kd_user,nama_user,password, level"
        + " from user where kd_user='"+kodeuser+"' AND password='"+tpassword.getText()+"'");
    kon.rs=kon.st.executeQuery(sql);
    if(kon.rs.next()){
        if(tlevel.getText().equals("1")) {
            Aplikasi_MenuUtama menu = new Aplikasi_MenuUtama();
            Aplikasi_Ganti_Password gantipass = new Aplikasi_Ganti_Password();
            menu.setLocationRelativeTo(null);
            menu.setVisible(true);
            menu>NamaUser=kon.rs.getString("nama_user");
            menu.KodeUser=kon.rs.getString("kd_user");
            menu.Level=kon.rs.getString("level");
            gantipass.Kode_User=kon.rs.getString("kd_user");
            this.dispose();
        }
        else if(tlevel.getText().equals("2")) {
            Aplikasi_MenuUtama_Kasir menu = new Aplikasi_MenuUtama_Kasir();
            Aplikasi_Ganti_Password gantipass = new Aplikasi_Ganti_Password();
            menu.setLocationRelativeTo(null);
            menu.setVisible(true);
            menu>NamaUser=kon.rs.getString("nama_user");
            menu.KodeUser=kon.rs.getString("kd_user");
            menu.Level=kon.rs.getString("level");
            gantipass.Kode_User=kon.rs.getString("kd_user");
            this.dispose();
        }
    }
}

else{
    JOptionPane.showMessageDialog(null, "KODE user atau PASSWORD SALAH");
    bersih();
    tkodeuser.setEnabled(true);
    tkodeuser.requestFocus();
}

}

catch(SQLException e){
    JOptionPane.showMessageDialog(null, e);
}

}
}

```

2. Klik kanan pada JFrame, pilih **Event – Window – windowActivated**, ketik script:

```

private void formWindowActivated(java.awt.event.WindowEvent evt) {
    // TODO add your handling code here:
    tpassword.setEnabled(false);
    btlogin.setEnabled(false);
    tkodeuser.requestFocus();
}

```

3. Klik kanan pada tkodeuser, pilih **Event – Action – actionPerformed**, ketik script:

```
private void tkodeuserActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String kodeuser = "";
    kodeuser = tkodeuser.getText();
    try {
        String sql = "Select *from user where kd_user='" + kodeuser + "'";
        kon.rs = kon.st.executeQuery(sql);
        if (kon.rs.next()) {
            tkodeuser.setEnabled(false);
            tlevel.setText(kon.rs.getString("level"));
            tpassword.setEnabled(true);
            tpassword.requestFocus();
            btlogin.setEnabled(true);
            tkodeuser.setEnabled(false);
        } else {
            JOptionPane.showMessageDialog(null, "kode user salah");
            bersih();
            tkodeuser.setEnabled(true);
            tkodeuser.requestFocus();
        }
    } catch (SQLException e) {
        JOptionPane.showMessageDialog(null, e);
    }
}
```

4. Klik kanan pada btkeluar, dan btlogin pilih **Event – Action – actionPerformed**, ketik script:

```
private void btkeluarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    dispose();
}

private void btloginActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    prosesLogin();
}
```

PERTEMUAN 5

FORM APLIKASI DATA BARANG DAN FORM TRANSAKSI

A. Membuat Form Aplikasi Data Barang

Form aplikasi data barang digunakan untuk menampilkan data barang yang ada di tabel barang, dan digunakan untuk proses transaksi. Langkah-langkah dalam membuat form aplikasi Data barang adalah sebagai berikut:

1. Masih dalam package aplikasi_toko, buat form baru dengan jDialog Form. Beri nama form dengan nama : **Aplikasi_Data_Barang**
2. Desain tampilan form seperti dibawah ini:

Spesifikasi properties komponen form Aplikasi_Data_Barang

No	Komponen	Properties
1	JTextField1	Nama Variabel : tcari Text : (kosongkan)
2	JTable	Nama Variabel : tbl_barang

3. Ketikkan script koneksi dan metode-metode yang diperlukan

```

package aplikasi_toko;
import java.sql.*;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;

/**
 *
 * @author Maruloh
 */
public class Aplikasi_Data_Barang extends javax.swing.JDialog {
    koneksi kon=new koneksi();
    public Aplikasi_Transaksi transaksi = null;
    private Object [][] databarang=null;
    private String[] label={"Kode Barang","Nama Barang","Harga Barang","Stok"};

    /**
     * Creates new form Aplikasi_Data_Barang
     */
    public Aplikasi_Data_Barang(java.awt.Frame parent, boolean modal) {
        super(parent, modal);
        initComponents();
        kon.setKoneksi();
        BacaTabelBarang();
    }

    private void BacaTabelBarang() {
        try{
            String sql="Select *From barang order by kd_brg";
            kon.rs=kon.st.executeQuery(sql);
            ResultSetMetaData m=kon.rs.getMetaData();
            int kolom=m.getColumnCount();
            int baris=0;
            while(kon.rs.next()){
                baris=kon.rs.getRow();
            }
            databarang=new Object[baris][kolom];
            int x=0;
            kon.rs.beforeFirst();
            while(kon.rs.next()){
                databarang[x][0]=kon.rs.getString("kd_brg");
                databarang[x][1]=kon.rs.getString("nm_brg");
                databarang[x][2]=kon.rs.getString("hrg_jual");
                databarang[x][3]=kon.rs.getString("stok");

                x++;
            }
            tbl_barang.setModel(new DefaultTableModel(databarang, label));
        }
        catch(SQLException e){
            JOptionPane.showMessageDialog(null, e);
        }
    }
}

```

Digunakan untuk menampilkan data barang keseluruhan

```

private void BacaTabelBarang2() {
    try{
        String sql="select *from barang where nm_brg like '%" +tcari.getText()+ "%' ";
        kon.rs=kon.st.executeQuery(sql);
        ResultSetMetaData m=kon.rs.getMetaData();
        int kolom=m.getColumnCount();
        int baris=0;
        while(kon.rs.next()){
            baris=kon.rs.getRow();
        }
        databarang=new Object[baris][kolom];
        int x=0;
        kon.rs.beforeFirst();
        while(kon.rs.next()){
            databarang[x][0]=kon.rs.getString("kd_brg");
            databarang[x][1]=kon.rs.getString("nm_brg");
            databarang[x][2]=kon.rs.getString("hrg_jual");
            databarang[x][3]=kon.rs.getString("stok");
            x++;
        }
        tbl_barang.setModel(new DefaultTableModel(databarang,label));
    }
    catch(SQLException e){
        JOptionPane.showMessageDialog(null, e);
    }
}

```

Digunakan untuk menampilkan data barang berdasarkan pencarian melalui tcari

4. Klik kanan pada tbl_barang, pilih **Event – Mouse – MouseClicked**, ketikkan script berikut:

```

private void tbl_barangMouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    int tabelBarang = tbl_barang.getSelectedRow();
    transaksi.KodeBarang = tbl_barang.getValueAt(tabelBarang, 0).toString();
    transaksi>NamaBarang=tbl_barang.getValueAt(tabelBarang, 1).toString();
    transaksi.HargaBarang=tbl_barang.getValueAt(tabelBarang, 2).toString();
    this.dispose();
}

```

5. Klik kanan pada tcari, pilih **Event – Key – KeyTyped**, ketikkan script berikut:

```

private void tcariKeyTyped(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    BacaTabelBarang2();
}

```

B. Membuat Form Transaksi

Desain program form transaksi

The design shows a transaction form with the following components:

- Header fields: No. Transaksi, Tanggal, and Kode User.
- Item selection row: Kode Barang (with a dropdown menu), Nama Barang, Harga Barang, Jumlah Beli, and Sub Total.
- Table with 4 columns: Title 1, Title 2, Title 3, and Title 4.
- Action buttons: TAMBAH, SIMPAN, BATAL, and KELUAR.
- Summary section: Rp **total** and input fields for Uang Bayar and Uang Kembali.

Implementasi program form transaksi

The screenshot shows the implemented form with the following data:

- Header fields: No. Transaksi: FK-1802515, Tanggal: 2018-02-12, Kode User: MRU.
- Item selection row: Kode Barang: LP15AS, Nama Barang: Laptop Asus 15, Harga Barang: 5000000, Jumlah Beli: 2, Sub Total: 10000000.
- Table:

Kode Barang	Nama Barang	Harga	Jumlah Beli	Subtotal
LP15AS	Laptop Asus 15	5000000	2	10000000
- Action buttons: TAMBAH, SIMPAN, BATAL, and KELUAR.
- Summary section: Rp **10000000**, Uang Bayar: 10000000, Uang Kembali: 0.

Langkah-langkah dalam membuat form Transaksi adalah sebagai berikut:

1. Masih dalam package aplikasi_toko, buat form baru dengan jDialog Form. Beri nama form dengan nama : **Aplikasi_Data_Barang**
2. Desain tampilan form seperti dibawah ini:

Spesifikasi properties komponen form
Aplikasi_Transaksi

No	Komponen	Properties
1	JTextField1	Nama Variabel : tnotrans Text : (kosongkan)
2	JTextField2	Nama Variabel : ttanggal Text : (kosongkan)
3	JTextField3	Nama Variabel : tkodeuser Text : (kosongkan)
4	JTextField4	Nama Variabel : tkodebarang Text : (kosongkan)
5	JTextField5	Nama Variabel : tnamabarang Text : (kosongkan)
6	JTextField6	Nama Variabel : tharga Text : (kosongkan)
7	JTextField7	Nama Variabel : tjumbel Text : (kosongkan)
8	JTextField8	Nama Variabel : tsubtotal Text : (kosongkan)
9	JTextField9	Nama Variabel : tbayar Text : (kosongkan)

Spesifikasi properties komponen form Aplikasi_
Transaksi

No	Komponen	Properties
10	JTextField10	Nama Variabel : tkembali Text : (kosongkan)
11	jButton1	Nama Variabel : btbrowse Text : ...
12	jButton2	Nama Variabel : bttambah Text : TAMBAH
13	jButton3	Nama Variabel : btsimpan Text : SIMPAN
14	jButton4	Nama Variabel : btbatal Text : TAMBAH
15	jButton5	Nama Variabel : bt_keluar Text : BATAL
16	jLabel	Text : total Nama Variabel : ltotal
17	jTable1	Nama Variabel : tbtransaksi

Logika Jalannya Program:

1. Saat form dijalankan, tanggal dan kode user muncul otomatis
2. Tekan tombol TAMBAH, no.transaksi muncul otomatis
3. Kode barang bisa langsung diinput, lalu tekan enter. Jika ditemukan maka nama barang, dan harga tampil.
4. Atau bisa dengan klik tombol browse (...), pilih kode barang yang akan dibeli, maka nama barang dan harga tampil
5. Jumlah beli diinput, lalu tekan enter, maka akan muncul pesan:”Mau Tambah Barang?”. Jika pilih Yes, ulangi perintah no.3 atau 4. Jika pilih No, total bayar akan muncul.
6. Input uang bayar, untuk melakukan pembayaran. Uang kembali akan muncul setelah dienter.
7. Tombol simpan untuk melakukan penyimpanan transaksi, dan struk akan muncul (pembuatan struk akan dibahas dipertemuan berikutnya saat pembuatan laporan)

PERTEMUAN 6

FORM TRANSAKSI (LANJUTAN)

C. Membuat Script Program Form Transaksi

Berikut Langkah-langkah dalam membuat script program form transaksi:

1. Ketikkan script koneksi dan metode-metode yang diperlukan

```

package aplikasi_toko;
import java.sql.*;
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.awt.event.KeyEvent;
import java.util.HashMap;
import net.sf.jasperreports.view.JasperViewer;
import net.sf.jasperreports.engine.JasperFillManager;
import net.sf.jasperreports.engine.JasperPrint;

/**
 *
 * @author Maruloh
 */
public class Aplikasi_Transaksi extends javax.swing.JFrame {
    koneksi kon = new koneksi();
    private Object [][] datasementara=null;
    private String[] labelsementara={"Kode Barang", "Nama Barang", "Harga", "Jumlah Beli", "Subtotal"};
    * Creates new form Aplikasi_Transaksi
    */
    public Aplikasi_Transaksi() {
        initComponents();
        kon.setKoneksi();
        setTanggal();
        awal();
    }

    public Date date=new Date();
    public SimpleDateFormat noformat=new SimpleDateFormat("yyMM");
    public String KodeBarang;
    public String NamaBarang;
    public String HargaBarang;
    public String KodeUser;

    public String getKodeBarang() {
        return KodeBarang;
    }

    public String getNamaBarang() {
        return NamaBarang;
    }

    public String getHargaBarang() {
        return HargaBarang;
    }

    public String getKodeUser() {
        return KodeUser;
    }

```

```
private void bersih()
{
    tnotrans.setText("");
    tkodebarang.setText("");
    tnamabarang.setText("");
    tharga.setText("");
    tjumbel.setText("");
    tsubtotal.setText("");
    tbayar.setText("");
    tkembali.setText("");
    ltotal.setText("0");
}

private void nonaktif()
{
    tnotrans.setEditable(false);
    ttanggal.setEditable(false);
    tkodebarang.setEditable(false);
    tkodeuser.setEditable(false);
    tnamabarang.setEditable(false);
    tharga.setEditable(false);
    tjumbel.setEditable(false);
    tsubtotal.setEditable(false);
    tbayar.setEditable(false);
    tkembali.setEditable(false);
}

private void aktif()
{
    tkodebarang.setEditable(true);
    tjumbel.setEditable(true);
    tbayar.setEditable(true);
}

private void awal()
{
    nonaktif();
}

void setTanggal(){
    java.util.Date skrg = new java.util.Date();
    java.text.SimpleDateFormat kal = new java.text.SimpleDateFormat("yyyy-MM-dd");
    ttanggal.setText(kal.format(skrg));
}
```

```

public String nomor()
{
    String urutan=null;
    try{
        kon.rs=kon.st.executeQuery("select right(notransaksi,3)+1 "
            + "from transaksi as Nomor order by notransaksi desc");
        if(kon.rs.next())
        {
            urutan=kon.rs.getString(1);
            while(urutan.length()<3)
                urutan="0"+urutan;
            urutan="FK-"+noformat.format(date)+urutan;
        }else
        {
            urutan="FK-"+noformat.format(date)+"001";
        }
    }
    catch (Exception e){
        JOptionPane.showMessageDialog(null, e);
    }
    return urutan;
}

```

```

private void TampilTabelSementara(){
    try{
        String sql="Select *From sementara order by kd_brg";
        kon.rs=kon.st.executeQuery(sql);
        ResultSetMetaData m=kon.rs.getMetaData();
        int kolom=m.getColumnCount();
        int baris=0;
        while(kon.rs.next()){
            baris=kon.rs.getRow();
        }
        datasementara=new Object[baris][kolom];
        int x=0;
        kon.rs.beforeFirst();
        while(kon.rs.next()){
            datasementara[x][0]=kon.rs.getString("kd_brg");
            datasementara[x][1]=kon.rs.getString("nm_brg");
            datasementara[x][2]=kon.rs.getString("hrg_jual");
            datasementara[x][3]=kon.rs.getString("jumlahbeli");
            datasementara[x][4]=kon.rs.getString("subtotal");
            x++;
        }
        tbtransaksi.setModel(new DefaultTableModel(datasementara,labelsementara));
    }
    catch(SQLException e){
        JOptionPane.showMessageDialog(null, e);
    }
}

```

```
private void tampildatabarang()
{
    try{
        String sql="select * from barang where kd_brg='" +tkodebarang.getText()+"'";
        kon.rs=kon.st.executeQuery(sql);
        if(kon.rs.next())
        {
            tnamabarang.setText(kon.rs.getString("nm_brg"));
            tharga.setText(kon.rs.getString("hrg_jual"));
            tjumbel.requestFocus();
        }
        else
        {
            JOptionPane.showMessageDialog(null,"Kode Barang "+tkodebarang.getText()+"tidak ditemukan");
        }
    }
    catch(SQLException e)
    {
        JOptionPane.showMessageDialog(null, e);
    }
}

private void SimpanSementara(){
    try{
        String sql="insert into sementara values('" +tkodebarang.getText()+"', "
            + "'" +tnamabarang.getText()+"', '" +tharga.getText()+"', "
            + "'" +tjumbel.getText()+"', '" +tsubtotal.getText()+"' )";
        kon.st.executeUpdate(sql);
        TampilTabelSementara();
    }
    catch(SQLException e){
        JOptionPane.showMessageDialog(null,e);
    }
}

private void HapusIsiSementara(){
    int row=tbtransaksi.getSelectedRow();
    int x;
    int total=Integer.parseInt(ltotal.getText());
    x=Integer.parseInt((String)tbtransaksi.getValueAt(row, 4));
    total=total-x;
    ltotal.setText(Integer.toString(total));
    try{
        String sql="Delete from sementara where kd_brg='"+(String)tbtransaksi.getValueAt(row,0)+"'";
        kon.st.executeUpdate(sql);
        TampilTabelSementara();
    }
    catch(SQLException e){
        JOptionPane.showMessageDialog(null, e);
    }
}
```

```
private void SimpanTransaksi() {
    try{
        String sql="insert into transaksi values('"+tnotrans.getText()+"', "
            + "'"+ttanggal.getText()+"', '"+ltotal.getText()+"', "
            + "'"+tkodeuser.getText()+"' )";
        kon.st.executeUpdate(sql);
    }
    catch(SQLException e){
        System.out.println("koneksi gagal"+ e.toString());
    }
}

private void simpanDetailTransaksi()
{
    try{
        String detail = "insert detailtransaksi select ' " +tnotrans.getText() + "', "
            + "jumlahbeli,subtotal,kd_brg from sementara";
        kon.st.executeUpdate(detail);
    }
    catch(SQLException e) {
        System.out.println("koneksi gagal"+ e.toString());
    }
}

private void HapusTabelSementara(){
    try{
        String sql="Delete from sementara ";
        kon.st.executeUpdate(sql);
        TampilTabelSementara();
    }
    catch(SQLException e){
        JOptionPane.showMessageDialog(null, e);
    }
}
```

PERTEMUAN 7

FORM TRANSAKSI (LANJUTAN)

2. Klik kanan pada **tkodebarang**, pilih **Event – Action – actionPerformed**, ketik script:

```
private void tkodebarangActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    tampilkandatabarang();  
}
```

Memanggil method **tampilkandatabarang()**, dengan melakukan pencarian terhadap kode barang yang diinput pada textfield tkodebarang.

3. Klik kanan pada **btbrowse**, pilih **Event – Action – actionPerformed**, ketik script:

```
private void btbrowseActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    boolean closable = true;  
    Aplikasi_Data_Barang dataBarang=new Aplikasi_Data_Barang(null, closable);  
    dataBarang.transaksi = this;  
    dataBarang.setVisible(true);  
    dataBarang.setResizable(true);  
    tkodebarang.setText(KodeBarang);  
    tnamabarang.setText>NamaBarang);  
    tharga.setText(HargaBarang);  
}
```

Memanggil form Aplikasi_Data_Barang, dan memanggil nilai yg ada pada variabel KodeBarang,>NamaBarang, HargaBarang kedalam textfield tkodebarang, tnamabarang, dan tharga.

4. Klik kanan pada **tkodebarang**, pilih **Event – Component – ComponentAdded**, ketik script:

```
private void tkodebarangComponentAdded(java.awt.event.ContainerEvent evt) {  
    // TODO add your handling code here:  
    tjumbel.requestFocus();  
}
```

Script untuk memindahkan_kursor ke textfield tjumbel.

- Klik kanan pada **btkeluar**, pilih **Event – Action – actionPerformed**, ketik script:

```
private void btkeluarActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    HapusTabelSementara();  
    dispose();  
}
```

- Klik kanan pada **btambah**, pilih **Event – Action – actionPerformed**, ketik script:

```
private void btambahActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    aktif();  
    tnotrans.setText(nomor());  
}
```

Memberikan nilai kepada textfield tnotrans, yang diambil dari method **nomor()**

- Klik kanan pada **jForm**, pilih **Event – Window – windowActivated**, ketik script:

```
private void formWindowActivated(java.awt.event.WindowEvent evt) {  
    // TODO add your handling code here:  
    tkodeuser.setText(KodeUser);  
    TampilTabelSementara();  
}
```

Memberi nilai kepada tkodeuser diambil dari variabel KodeUser, dan menampilkan tabel sementara dari method **TampilTabelSementara()**

- Klik kanan pada **btbatal**, pilih **Event – Action – actionPerformed**, ketik script:

```
private void btbatalActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    awal();  
    HapusTabelSementara();  
    tnotrans.setText("");  
}
```

Saat diklik batal, maka isi tabel sementara akan dihapus melalui method **HapusTabelSementara**, dan nomor transaksi menjadi kosong kembali.

9. Klik kanan pada **tjumbel**, pilih **Event – Action – actionPerformed**, ketik script:

```
private void tjumbelActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    int harga, jumbel, total;
    harga=Integer.parseInt(tharga.getText());
    jumbel=Integer.parseInt(tjumbel.getText());
    total=harga*jumbel;
    tsubtotal.setText(Integer.toString(total));
    SimpanSementara();
    TampilTabelSementara();
    int ttl=0;
    for(int a=0;a<tbtransaksi.getRowCount();a++)
    {
        int sub=Integer.parseInt((String)tbtransaksi.getValueAt(a, 4));
        ttl+=sub;
    }
    ltotal.setText(Integer.toString(ttl));
    if (JOptionPane.showConfirmDialog(this, "Mau Tambah Barang?",
        "konfirmasi", JOptionPane.YES_NO_OPTION) == JOptionPane.YES_OPTION) {
        aktif();
        tkodebarang.requestFocus();
        tkodebarang.setText("");
        tnamabarang.setText("");
        tharga.setText("");
        tjumbel.setText("");
        tsubtotal.setText("");
    } else {
        tbayar.requestFocus();
    }
}
}
```

Script untuk menghitung subtotal dari tiap penjualan, untuk disimpan kedalam tabel sementara, dan memberikan total keseluruhan dari semua subtotal jika transaksi lebih dari satu.

10. Klik kanan pada **tbtransaksi**, pilih **Event – Key – KeyPressed**, ketik script:

```
private void tbtransaksiKeyPressed(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if(evt.getKeyCode()==KeyEvent.VK_BACK_SPACE)
    {
        HapusIsiSementara();
    }
}
}
```

Script untuk menghapus isi tabel sementara yang telah tersimpan.

11. Klik kanan pada **btsimpan**, pilih **Event – Action – actionPerformed**, ketik script:

```
private void btsimpanActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    SimpanTransaksi();  
    simpanDetailTransaksi();  
    JOptionPane.showMessageDialog(this, "Berhasil disimpan",  
        "Informasi", JOptionPane.INFORMATION_MESSAGE);  
    TampilTabelSementara();  
    HapusTabelSementara();  
    awal();  
    bersih();  
    tnotrans.setText("");  
    TampilTabelSementara();  
}
```

Script untuk melakukan proses penyimpanan ke tabel transaksi dan detailtransaksi melalui pemanggilan method **SimpanTransaksi()** dan **simpanDetailTransaksi()**.

12. Klik kanan pada **tbayar**, pilih **Event – Action – actionPerformed**, ketik script:

```
private void tbayarActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    int kembali, total, bayar;  
    total=Integer.parseInt(ltotal.getText());  
    bayar=Integer.parseInt(tbayar.getText());  
    kembali=bayar-total;  
    tkembali.setText(Integer.toString(kembali));  
}
```

Script untuk menghitung uang kembali setelah penginputan uang bayar.

PERTEMUAN 8

Quis Project

Mempresentasikan perkembangan project program kelompok kepada dosen (\pm 10-20 menit). Lingkup presentasi Quis Project yaitu:

1. Pembuatan database yang benar
2. Pembuatan form Login, menu utama, dan Master yang sudah dapat dijalankan
3. Pembuatan desain form transaksi

PERTEMUAN 9

Membuat Form Ganti Password User

Form aplikasi Ganti Password digunakan untuk merubah password user yang sedang login. Langkah-langkah dalam membuat form aplikasi Ganti Password adalah sebagai berikut:

1. Masih dalam package aplikasi_toko, buat form baru dengan jDialog Form. Beri nama form dengan nama : **Aplikasi_Ganti_Password**
2. Desain tampilan form seperti dibawah ini:

The image shows a dialog box titled "GANTI PASSWORD" with a light gray background. It contains four text input fields and one button. The fields are labeled "Kode User", "Password Lama", "Password Baru", and "Konfirmasi Password". The button is labeled "KELUAR". Each field and the button are circled with a number from 1 to 5, corresponding to the table below.

No	Komponen	Properties
1	JTextField1	Nama Variabel : tKode_user Text : (kosongkan)
2	JTextField2	Nama Variabel : tpassLama Text : (kosongkan)
3	JTextField3	Nama Variabel : tpassBaru Text : (kosongkan)
4	JTextField4	Nama Variabel : tconfirmPass Text : (kosongkan)
5	JButton1	Nama Variabel : jbutton1

Logika jalannya program:

1. Pertama run yang aktif hanya password lama. Kode user akan otomatis terisi sesuai dengan login.
2. Kursor muncul di password lama. Ketikkan password lama lalu enter jika tidak ada, muncul pesan (password tidak ada). Jika ada, kursor pindah ke password baru.
3. Ketikkan password baru, enter. Ketikkan konfirmasi password baru lalu enter. Jika isi password baru dan konfirmasi password tidak sama, muncul pesan: (konfirmasi dan password baru berbeda). Jika sama, muncul pesan:(password telah diupdate).
4. Tekan TUTUP untuk keluar kembali ke menu utama. Untuk membatalkan tekan TUTUP aja deh.

Listing Program Form Ganti Password:

```
package aplikasi_toko;
import java.io.File;
import javax.swing.JOptionPane;
import java.sql.*;
import javax.swing.*;
import java.awt.event.KeyEvent;

/**
 *
 * @author Maruloh
 */
public class Aplikasi_Ganti_Password extends javax.swing.JFrame {
    koneksi kon = new koneksi();

    /**
     * Creates new form Aplikasi_Ganti_Password
     */
    public Aplikasi_Ganti_Password() {
        initComponents();
    }
}
```

```
        kon.setKoneksi();
    }

    public String Kode_User;
    public String getKodeUser()
    {
        return Kode_User;
    }

    void nonaktif()
    {
        tKode_user.setEnabled(false);
        tpassLama.setEnabled(false);
        tpassBaru.setEnabled(false);
        tkonfirmPass.setEnabled(false);
    }

    void bersih()
    {
        tpassLama.setText("");
        tpassBaru.setText("");
        tkonfirmPass.setText("");
    }

    private void passlama()
    {
        try {
            String sql = "Select *from user where password='" + tpassLama.getText()
            kon.rs = kon.st.executeQuery(sql);
            if (kon.rs.next()) {
                // JOptionPane.showMessageDialog(null,"Kode Barang ditemukan, silakan
                tpassBaru.setEnabled(true);
                tpassBaru.requestFocus();
            } else {
                JOptionPane.showMessageDialog(null, "Kode Barang tidak ada, Silakan"
                + " Isi Data");
                tpassLama.requestFocus();
            }
        }
    }
}
```

```
        } catch (SQLException e) {
            JOptionPane.showMessageDialog(null, e);
        }
    }

    private void konfirmasiPassword()
    {
        try{
            if(tkonfirmPass.getText().equals(tpassBaru.getText()))
            {
                String sql = "update user set password='" + tkonfirmPass.getText()
                    + "' where kd_user='" + tKode_user.getText() + "'";
                kon.st.executeUpdate(sql);
                JOptionPane.showMessageDialog(null, "Password telah terupdate");
                nonaktif();
                bersih();
            }

            else {
                JOptionPane.showMessageDialog(null, "Password baru dan konfirmasi "
                    + "password harus sama");
                nonaktif();
                bersih();
            }
        } catch (SQLException e) {
            JOptionPane.showMessageDialog(null, e);
        }
    }

    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    Generated Code
```

```
private void initComponents() {...} // </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    dispose();
}

private void formWindowActivated(java.awt.event.WindowEvent evt) {
    // TODO add your handling code here:
    tKode_user.setText(Kode_User);
    nonaktif();
    tpassLama.setEnabled(true);
}

private void tpassLamaActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    passlama();
}

private void tpassBaruActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    tkonfirmPass.setEnabled(true);
    tkonfirmPass.requestFocus();
}

private void tkonfirmPassActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}

private void tkonfirmPassKeyPressed(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if(evt.getKeyCode()==KeyEvent.VK_ENTER)
    {
        konfirmpassword();
    }
}
```

Tampilan running form Aplikasi_Ganti_Password



The screenshot shows a Java Swing window titled "GANTI PASSWORD". The window contains the following elements:

- Title Bar:** Standard Windows window controls (minimize, maximize, close).
- Header:** The text "GANTI PASSWORD" is centered at the top.
- Form Fields:**
 - "Kode User": A text field containing the value "IAI".
 - "Password Lama": An empty text field.
 - "Password Baru": An empty text field.
 - "Konfirmasi Password": An empty text field.
- Button:** A button labeled "KELUAR" is positioned at the bottom center of the window.

PERTEMUAN 10

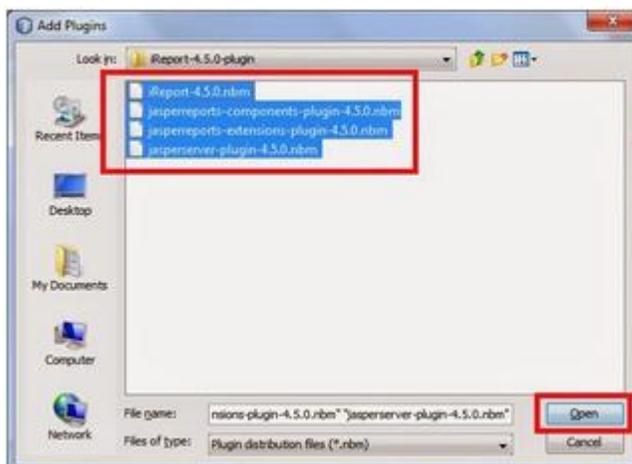
Pembuatan Laporan Data Barang, User, Dan Struk Penjualan

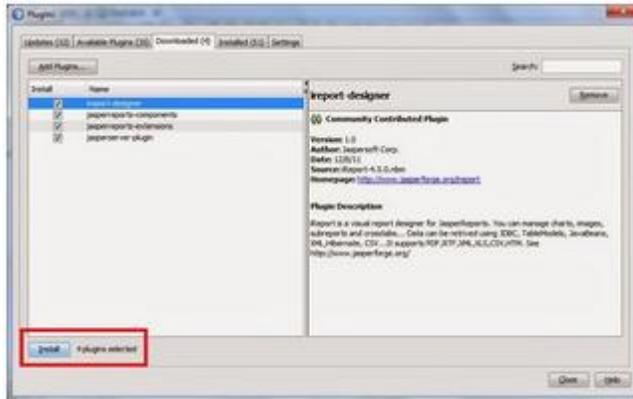
A. Persiapan pembuatan Laporan

Dalam bahasa pemrograman Java kita dapat membuat laporan dengan menggunakan Plugin tambahan, yaitu iReport. Ireport atau yang juga dikenal dengan nama Jasper Report adalah plugin milik IDE Netbeans, yang dapat digunakan secara bebas karena bersifat Open Source. Sebelum kita mulai pembuatan laporan, silakan download plugin iReport melalui link [disini](#). Selain plugin iReport, kita juga membutuhkan Library Jar untuk memanggil dan menampilkan iReport silakan download melalui link [disini](#).

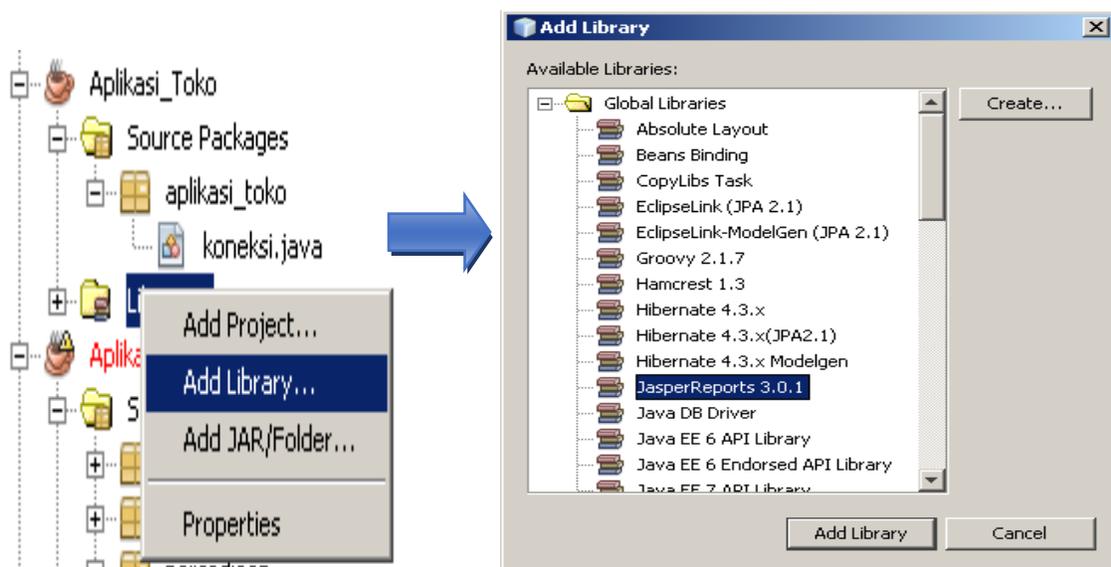
Langkah-langkah dalam Instalasi Plugin iReport di Netbeans adalah sebagai berikut:

1. Jalankan Netbeans, pilih menu Tools – Plugin
2. Pilih tab Downloaded – Add Plugins
3. Cari hasil download Plugin iReport nya, pastikan sudah mengekstrak Plugin iReport
4. Pilih semua file *.nbm yang ada didalam folder plugin iReport





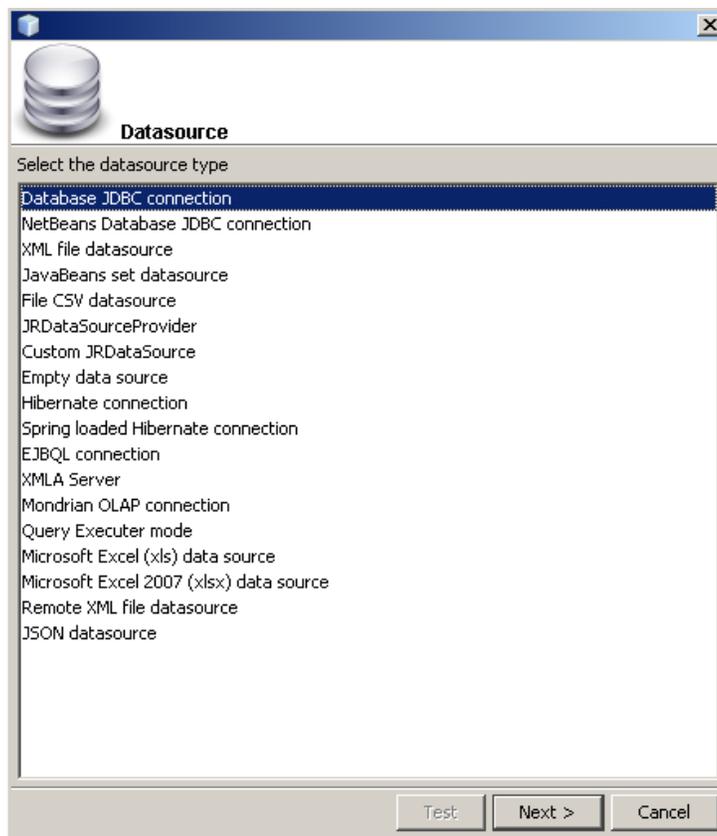
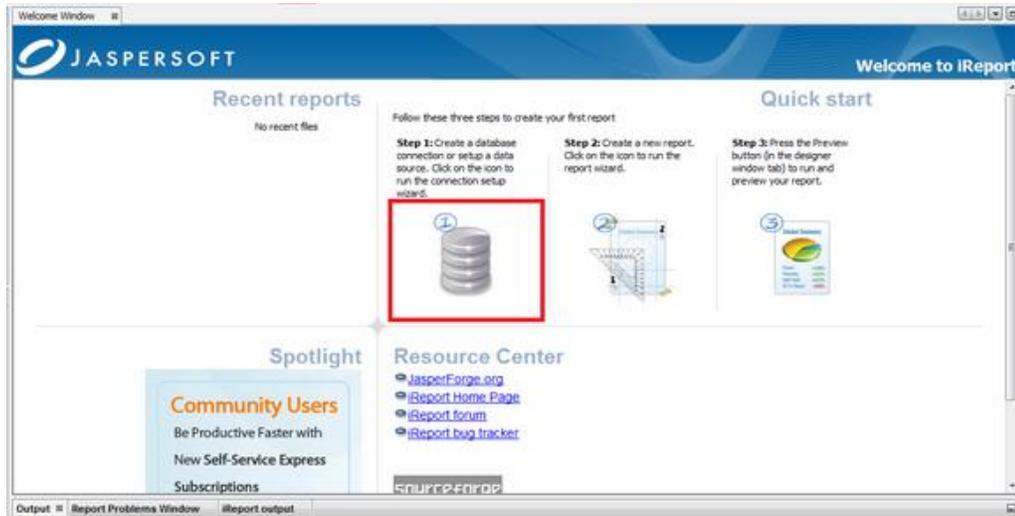
5. Ikuti langkah-langkah seperti saat anda menginstall program aplikasi. Jika proses sudah selesai, biarkan netbeans untuk restart.
6. Tambahkan Libraries untuk bisa terhubung ke database
Klik kanan pada Libraries – Add Libraries – Pilih JasperReports 3.0.1 – Add Library



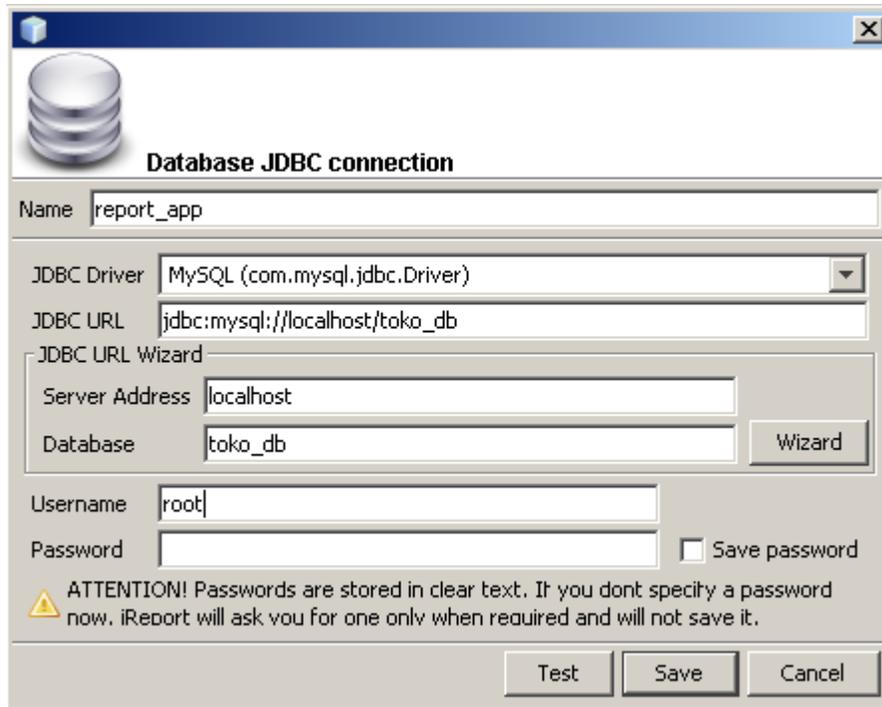
B. Pembuatan Laporan Data Barang

Langkah-langkah untuk membuat Laporan Data barang adalah sebagai berikut:

1. Pada jendela Welcome Window klik gambar drum (Datasource) untuk membuat Datasource baru yang digunakan untuk koneksi laporan ke Database.
2. Pilih Database JDBC Connection - Next



3. Silakan isi data berikut, klik test sebelum melakukan penyimpanan (Save)



Database JDBC connection

Name: report_app

JDBC Driver: MySQL (com.mysql.jdbc.Driver)

JDBC URL: jdbc:mysql://localhost/toko_db

JDBC URL Wizard

Server Address: localhost

Database: toko_db

Username: root

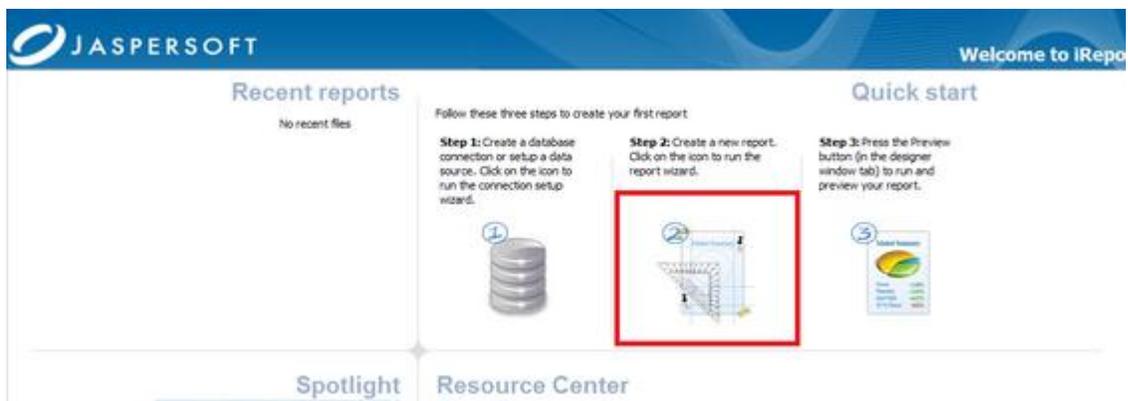
Password:

Save password

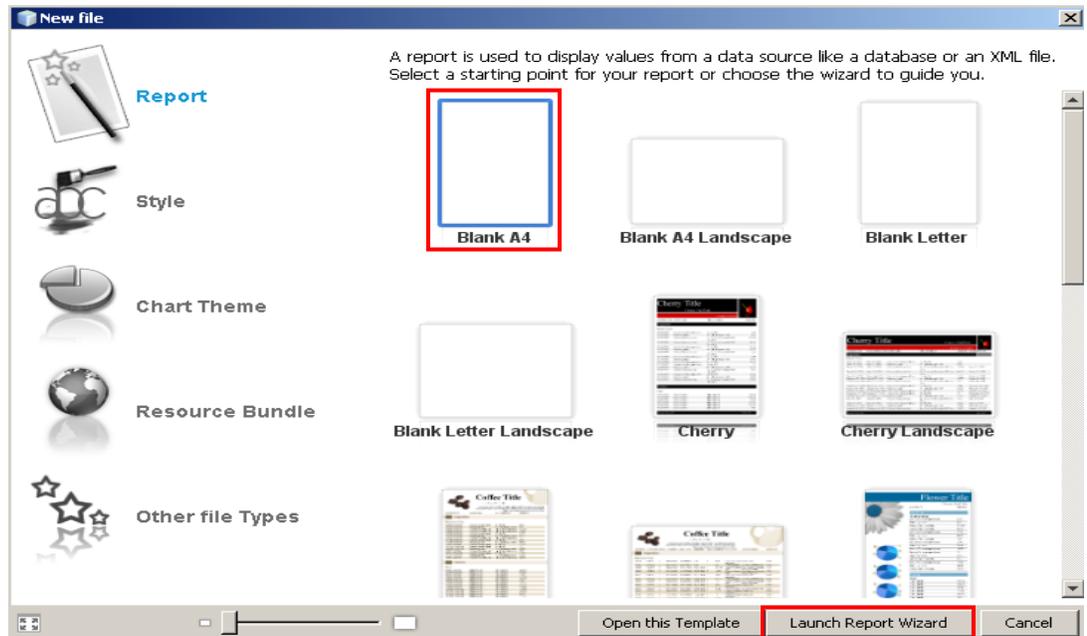
ATTENTION! Passwords are stored in clear text. If you dont specify a password now, iReport will ask you for one only when required and will not save it.

Test Save Cancel

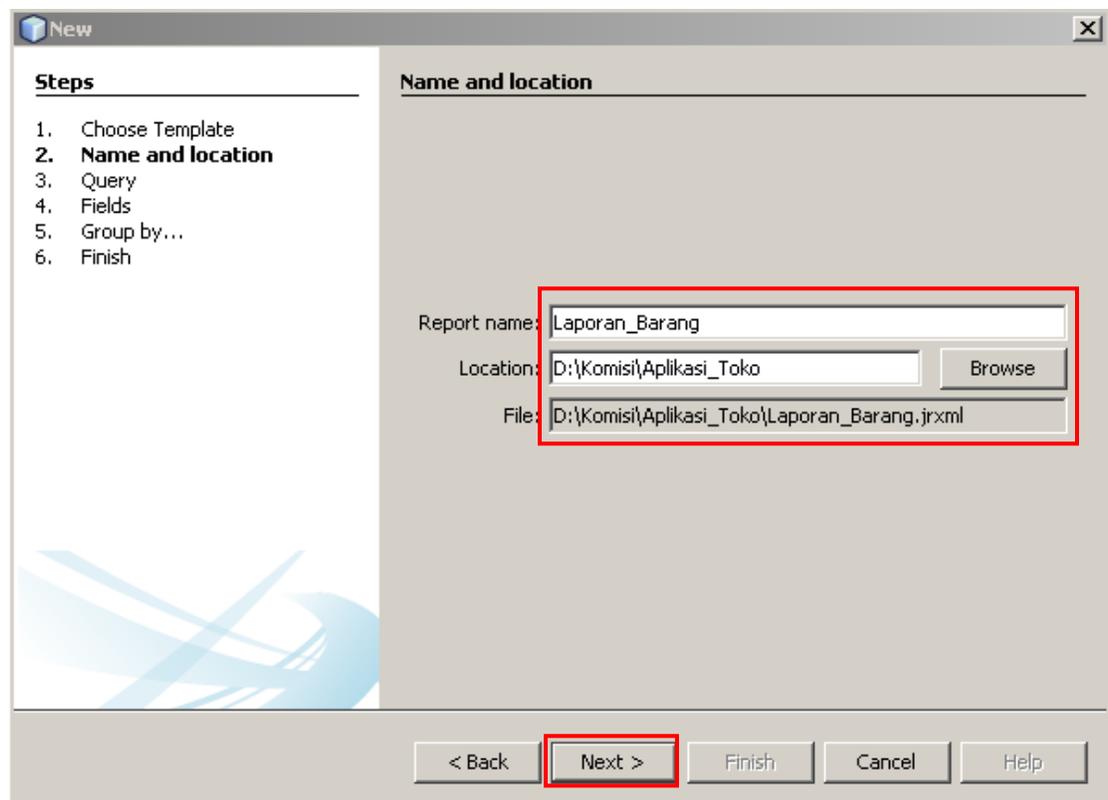
4. Setelah kita membuat Datasource untuk koneksi laporan ke Database, langkah berikutnya adalah membuat laporan dengan iReport. Masih pada jendela Welcome Window, klik gambar laporan



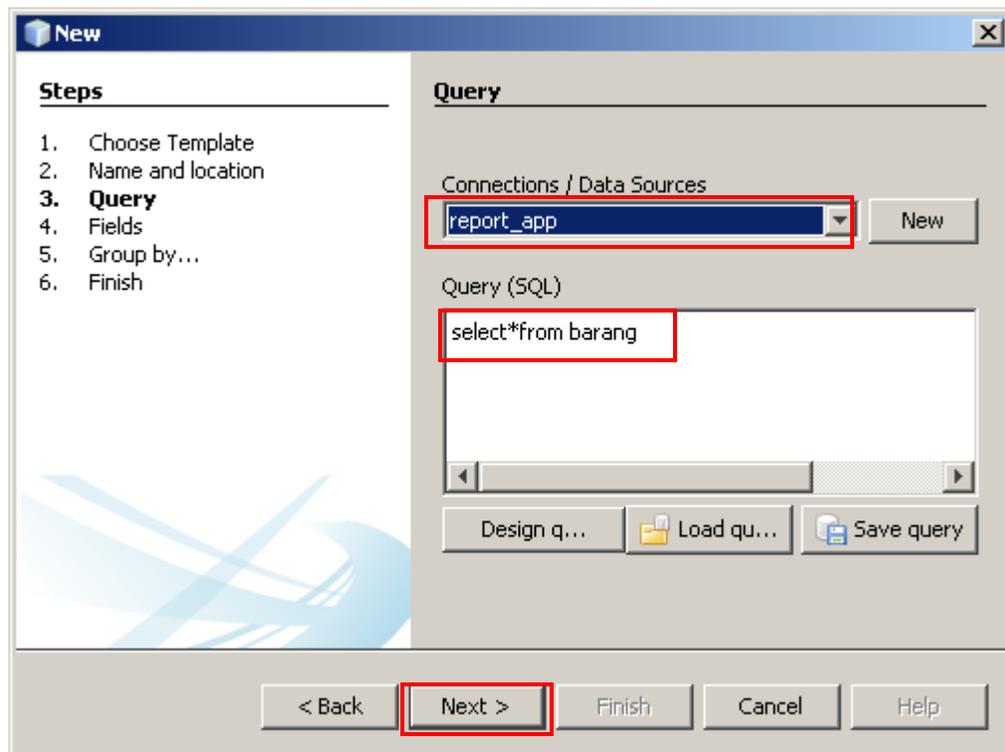
5. Pilih template (**pilih Blank A4, kita desain sendiri**) – pilih **Launch Report Wizard**



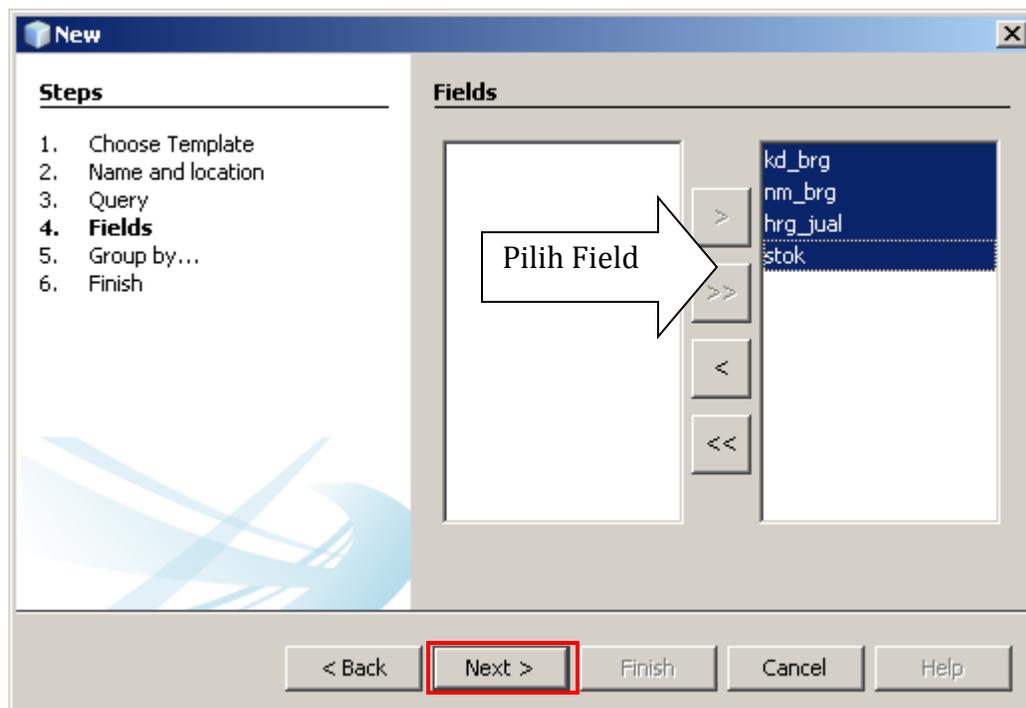
6. Beri nama report dan atur tempat penyimpanannya. Masukkan dalam package aplikasi_toko. Nama report : **Laporan_Barang**. Klik **Next >**



7. Tuliskan Query yang diperlukan. Karena kita akan menampilkan laporan data Barang, maka kita tuliskan query dengan **“Select * from barang”** || **Next >**



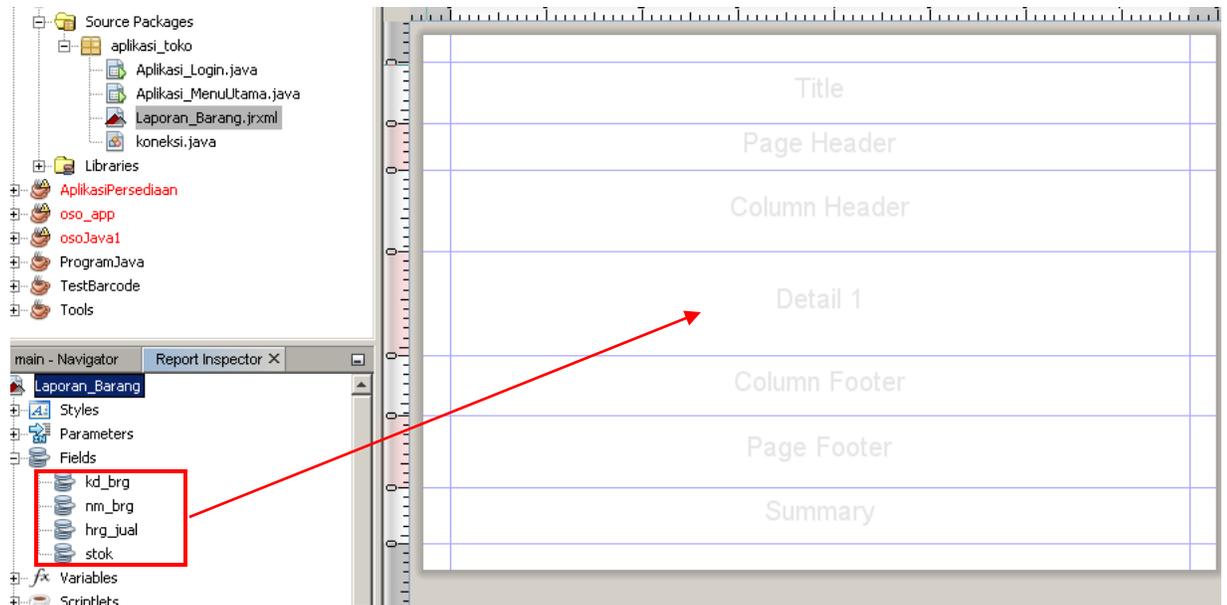
8. Pilih field yang akan ditampilkan pada laporan || **Next >**



9. Pada jendela Group By klik Next saja. Setelah itu **Finish**

10. Masukkan field yang ingin ditampilkan ke laporan (drag ke dalam desain report

Detail 1)



11. Hasil setelah dimasukkan kedalam desain report

Title			
Page Header			
kd_brg	nm_brg	hrng_jual	stok
Column Header			
\${kd_brg}	\${nm_brg}	\${hrng_jual}	\${stok}
Detail 1			
Column Footer			
Page Footer			
Summary			

12. Silakan desain report seperti berikut:

LAPORAN DATA BARANG PT SUKSES SELALU			
Page Header			
Kode Barang	Nama Barang	Harga	Stok
\${f{kd_brg}}	\${f{nm_brg}}	\${f{hrg_jual}}	\${f{stok}}
Detail 1			
Column Footer			
Page Footer			
Summary			

Gunakan Static Text pada Palette

13. Klik Preview Jika ingin melihat hasil reportnya.

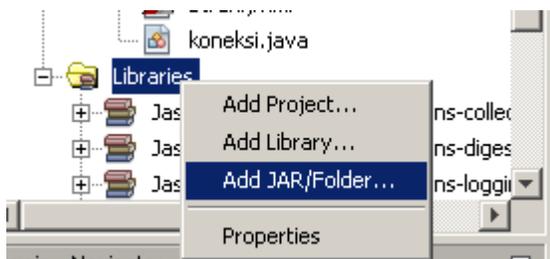
LAPORAN DATA BARANG PT SUKSES SELALU

Kode Barang	Nama Barang	Harga	Stok
LP15AS	Laptop Asus 15	5000000.0	10.0
NT10AS	Netbook Asus 10 Inc	2750000.0	3.0
TV21LG	TV LG 21 inc	3500000.0	10.0
TV29SA	TV Samsung 29 inc	4500000.0	10.0

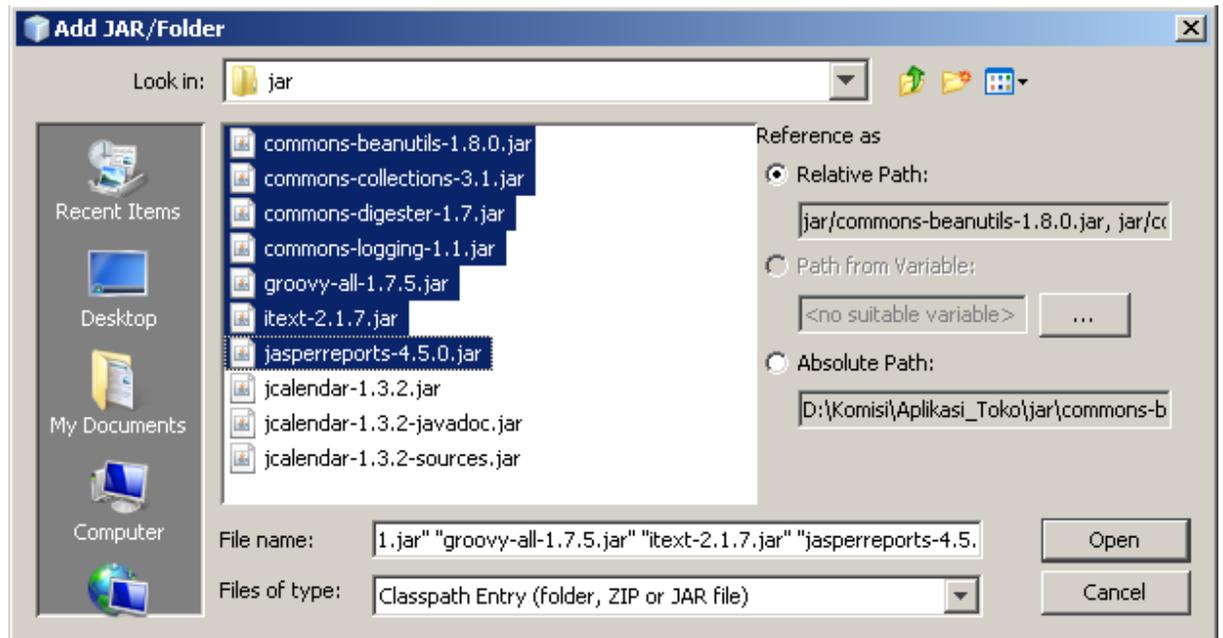
Menampilkan Laporan iReport

Seperti dijelaskan diawal, Untuk bisa menampilkan laporan barang, kita perlu Library tambahan iReport jar. Langkah-langkahnya adalah sebagai berikut:

1. pastikan anda telah mendownload iReport jar. Simpan file-file jar tersebut ke dalam folder jar (silakan buat folder jar didalam project Aplikasi _Toko)
2. klik kanan pada Libraries – Add JAR/Folder



3. pilih File JAR yang dibutuhkan, lihat gambar, || **Open**



4. untuk script coding pemanggilan laporan barang ada di menu Utama (Pert. 4, langkah 21)

C. Pembuatan Laporan Data User

Untuk membuat laporan data User silakan anda ikuti langkah-langkah seperti membuat laporan data barang. Dimulai dari langkah ke 4 sampai dengan selesai (langkah ke 1-3 adalah membuat datasource, cukup sekali untuk membuatnya). Beri nama laporan user dengan **Laporan_User**.

D. Pembuatan Struk Penjualan

Langkah-langkah dalam membuat struk penjualan adalah sebagai berikut:

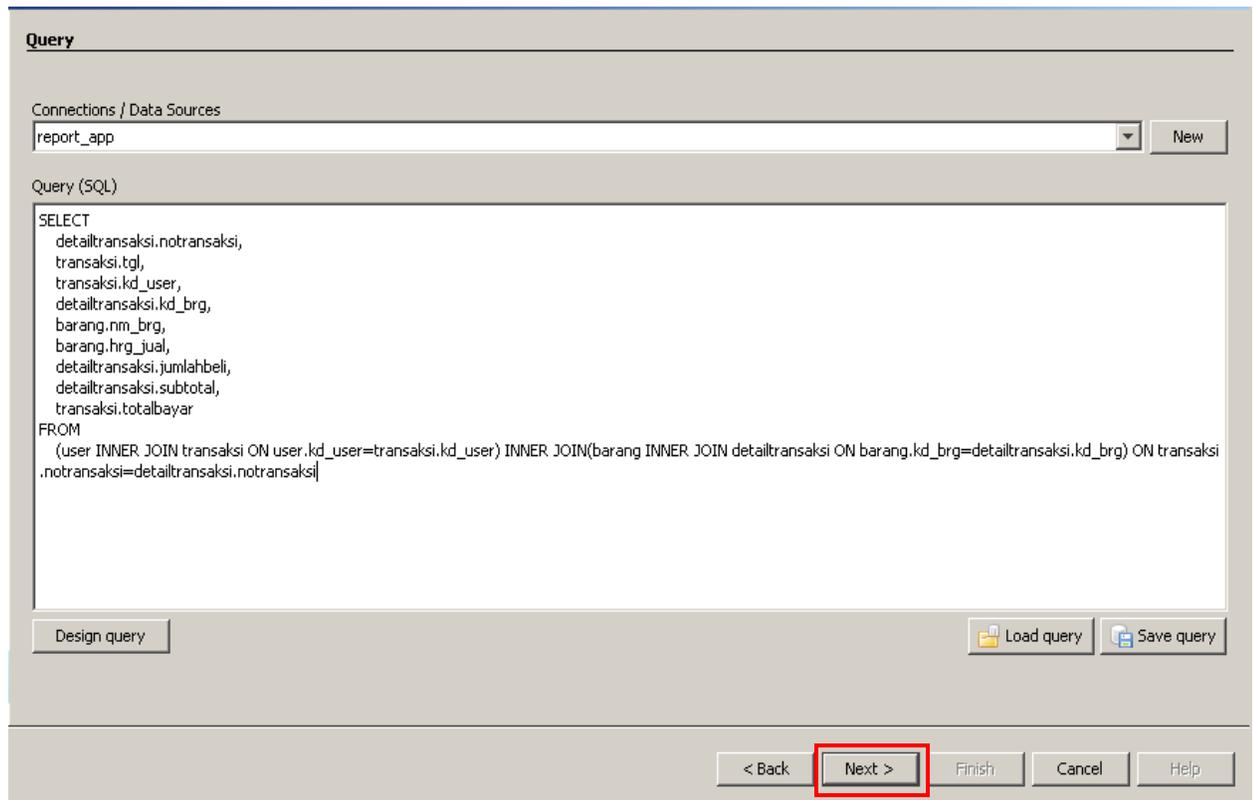
1. Ikuti langkah-langkah ke 4 sampai dengan langkah ke 6 dalam pembuatan laporan data barang.

Template : **Blank A4 Landscape**

Nama report : **Struk**

Tempat penyimpanan disamakan dengan penyimpanan laporan barang

2. Tuliskan query berikut, || Next >



The screenshot shows a "Query" window with the following content:

Connections / Data Sources
report_app

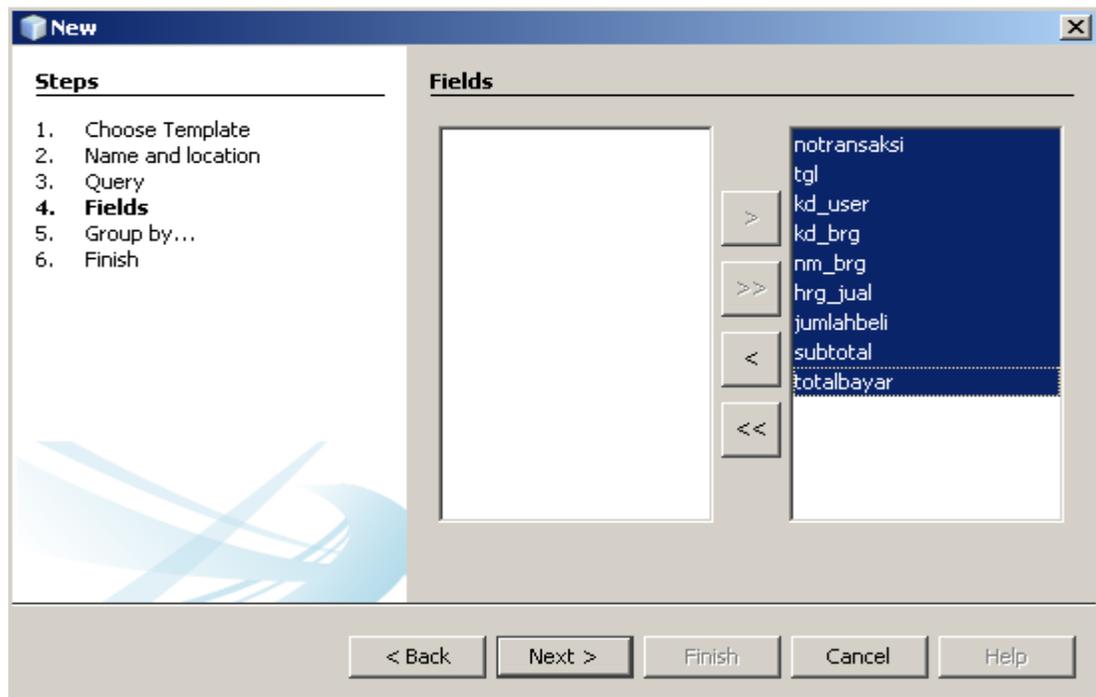
Query (SQL)

```
SELECT
detailtransaksi.notransaksi,
transaksi.tgl,
transaksi.kd_user,
detailtransaksi.kd_brg,
barang.nm_brg,
barang.hrg_jual,
detailtransaksi.jumlahbeli,
detailtransaksi.subtotal,
transaksi.totalbayar
FROM
(user INNER JOIN transaksi ON user.kd_user=transaksi.kd_user) INNER JOIN(barang INNER JOIN detailtransaksi ON barang.kd_brg=detailtransaksi.kd_brg) ON transaksi
.notransaksi=detailtransaksi.notransaksi
```

Buttons: Design query, Load query, Save query

Navigation buttons: < Back, **Next >**, Finish, Cancel, Help

3. Pilih field yang akan ditampilkan di struk



The screenshot shows a "New" dialog box with the following content:

Steps

1. Choose Template
2. Name and location
3. Query
- 4. Fields**
5. Group by...
6. Finish

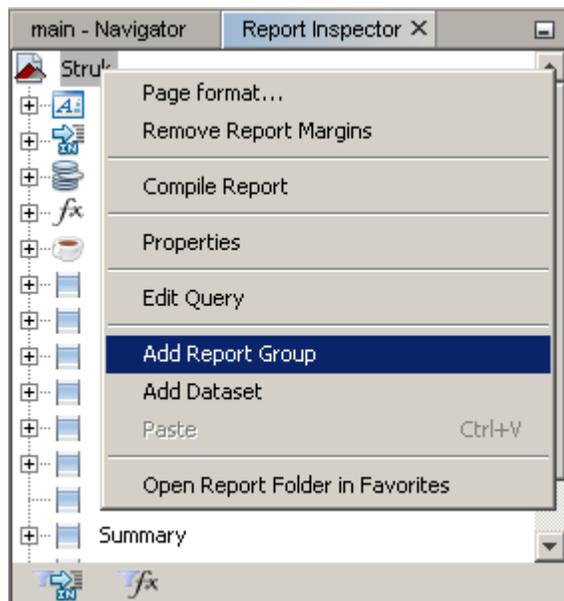
Fields

notransaksi
tgl
kd_user
kd_brg
nm_brg
hrj_jual
jumlahbeli
subtotal
totalbayar

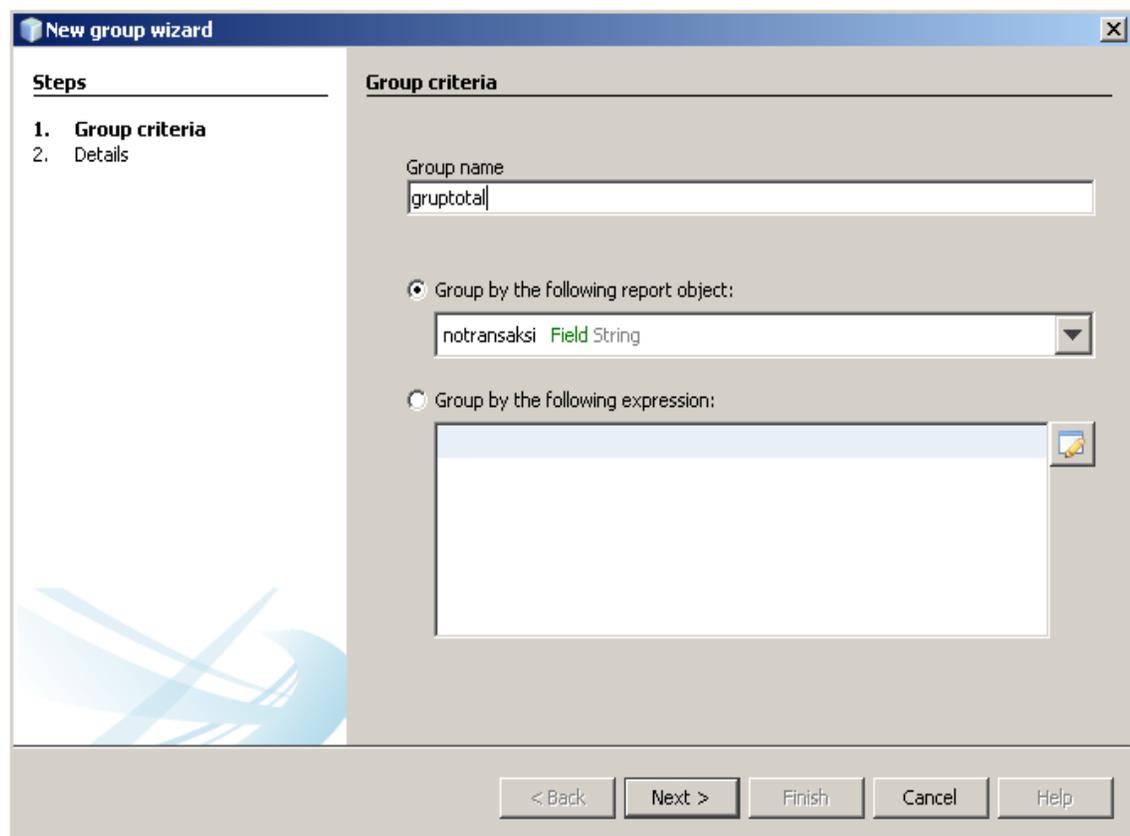
Navigation buttons: < Back, Next >, Finish, Cancel, Help

4. Pada jendela Group By klik Next saja. Setelah itu **Finish**
5. Membuat report Group di Footer untuk menampilkan field Total. Klik Pada

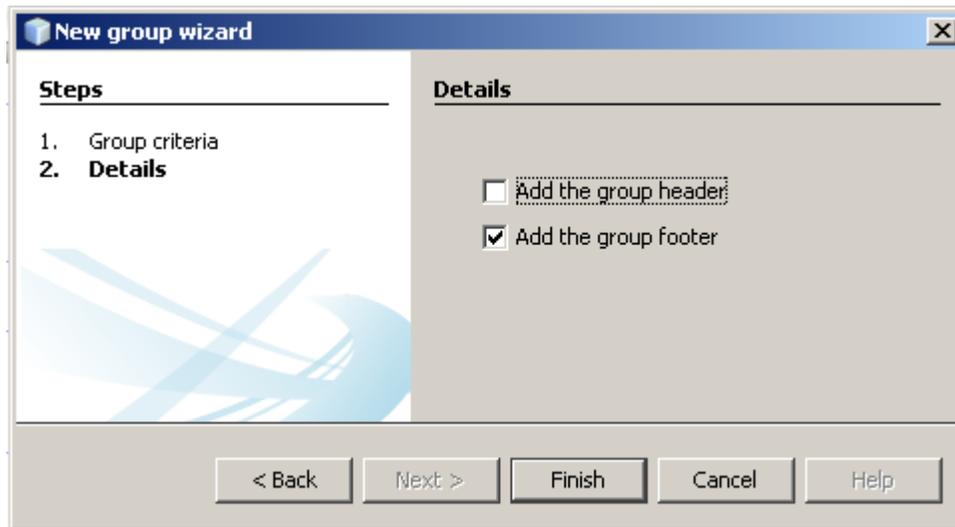
Report Inspector – Add Report Group



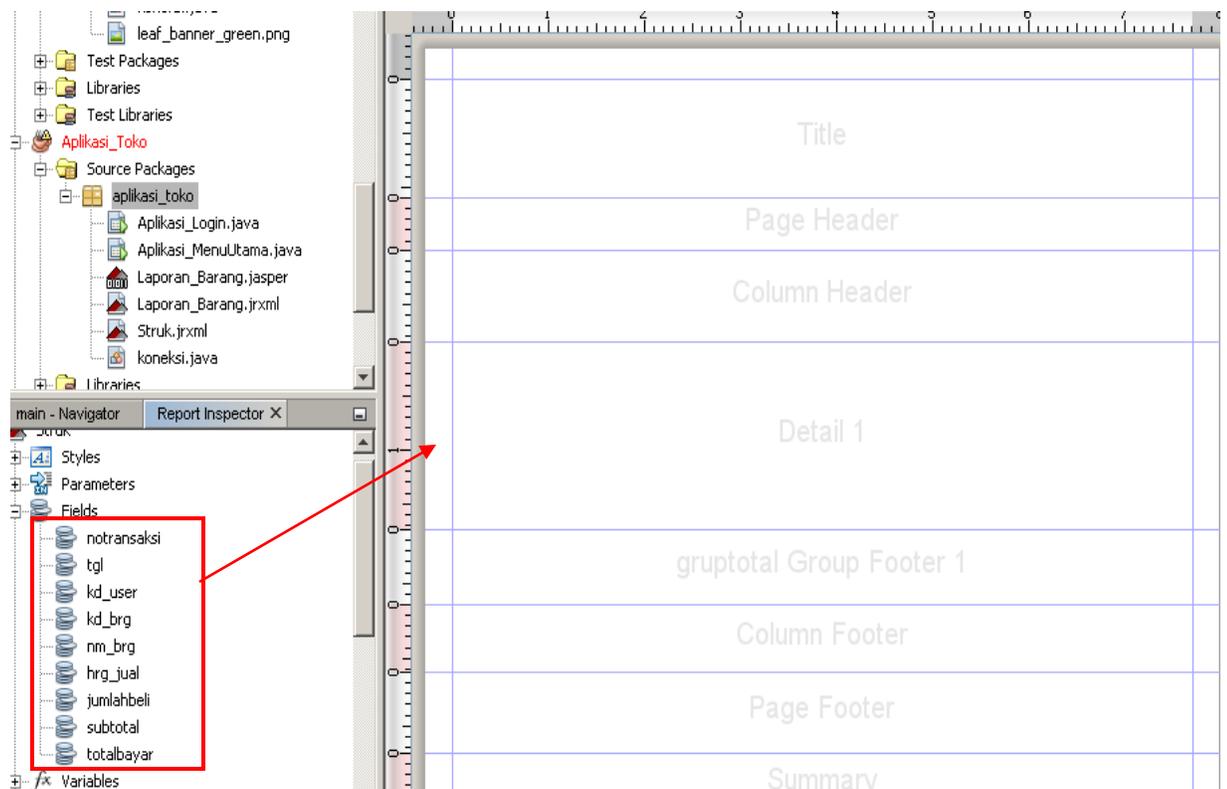
6. Beri nama grup seperti berikut



7. Untuk Details, ceklist **Add the Group Footer** || **Finish**



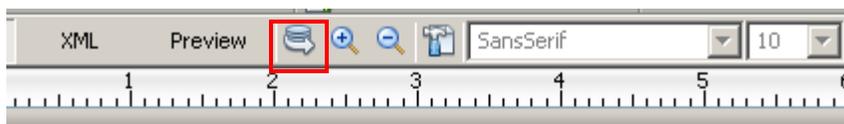
8. Masukkan field yang ingin ditampilkan kedalam Struk



9. Desain struk seperti di bawah ini:

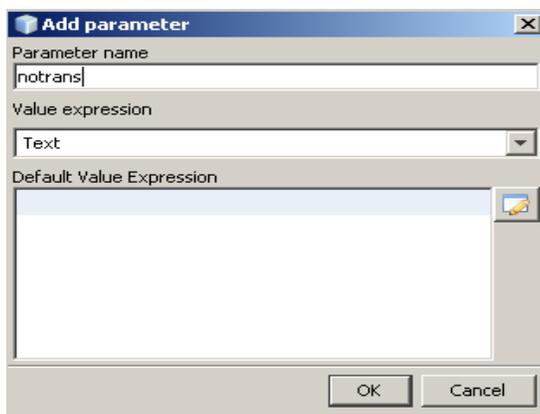
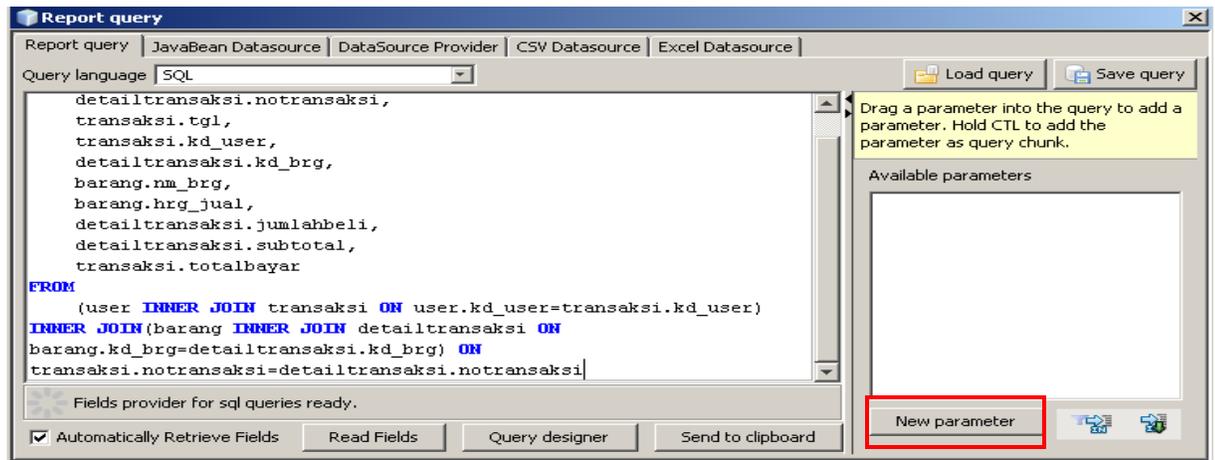
Title				
Struk Pembelian				
No. Transaksi	#{notransaksi}	Kode User		#{kd_user}
Tanggal	#{tgl}			
Kode Barang	Nama Barang	Harga	Qty	subtotal
#{kd_brg}	#{nm_brg}	#{hrg_jual}	1	#{subtotal}
				#{totalbayar}
<ul style="list-style-type: none"> - Barang yang sudah dibeli tidak dapat ditukar kembali - Terima kasih telah melakukan pembelian ditempat kami, semoga anda Puas dengan Pelayanan Kami 				
Summary				

10. Langkah berikutnya adalah kita membuat parameter nomor transaksi, karena kita akan mencetak struk berdasarkan nomor transaksi. Pilih **report query** yang berada di sebelah Preview

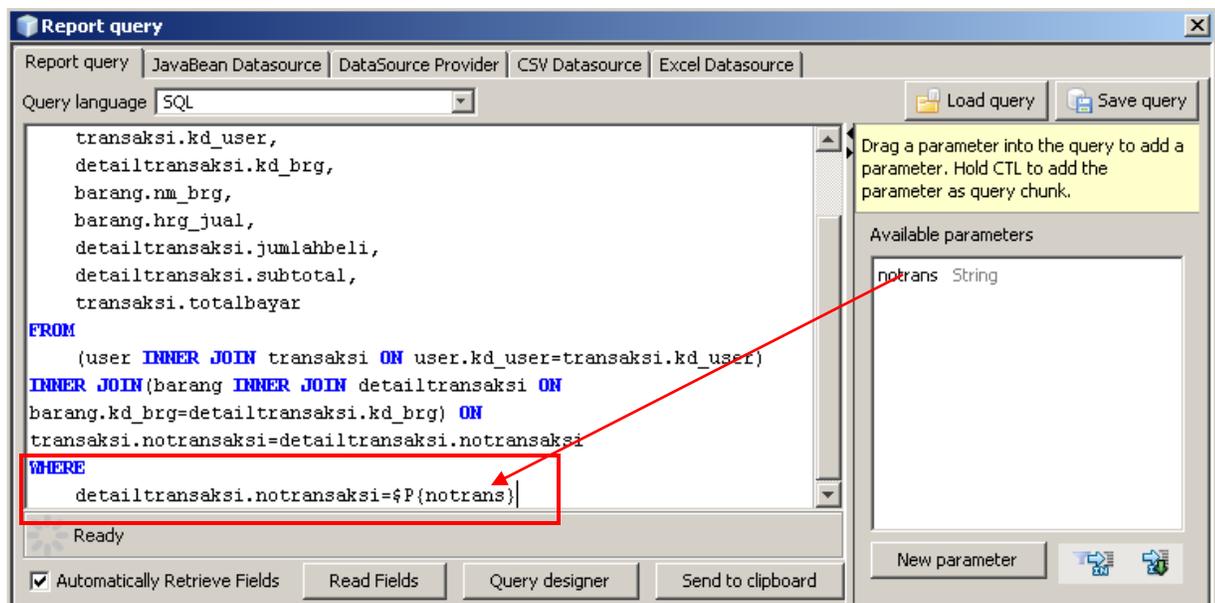


Title				
Struk Pembelian				

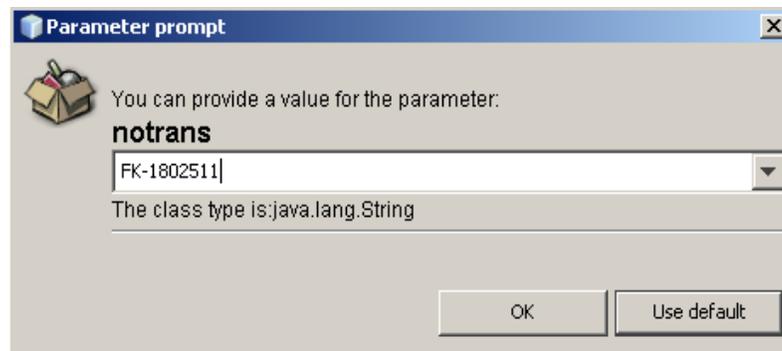
11. Klik New Parameter, beri nama **notrans**, value Expression : **Text** || **OK**



12. Edit query yang ada pada report query seperti dibawah ini, setelah itu **OK**



13. Klik Preview, maka akan muncul jendela **Parameter Prompt**, masukan salah satu notarnsaksi yang ada di tabel. Contoh saya masukkan no.transaksi **FK-1802511** || **OK**



14. Hasilnya adalah sebagai berikut:

Struk Pembelian

No.Transaksi	FK-1802511			
Tanggal	2/5/18 12:00 AM	Kode User	MRU	
Kode Barang	Nama Barang	Harga	Qty	subtotal
LP15AS	Laptop Asus 15	5000000.0	1	5000000.0
TV21LG	TV LG 21 inc	3500000.0	1	3500000.0
				8500000.0

- Barang yang sudah dibeli tidak dapat ditukar kembali
- Terima kasih telah melakukan pembelian ditempat kami, semoga anda Puas dengan Pelayanan Kami

15. Untuk script pemanggilan Struk, ada di Form Transaksi penjualan. Silakan buka kembali **Aplikasi_Transaksi**. Tambahkan method, beri nama **cetakstruk**.

```

private void cetakstruk(){
    try{
        String file = "src/aplikasi_toko/struk.jasper";
        HashMap param = new HashMap();
        param.put("notrans", tnotrans.getText());
        JasperPrint print = JasperFillManager.fillReport(file, param, kon.setKoneksi());
        JasperViewer.viewReport(print, false);
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(null, e.getMessage());
    }
}

```

16. Pada **btsimpan**, tambahkan script untuk memanggil method **cetakstruk**.

```
private void btsimpanActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    SimpanTransaksi();  
    simpanDetailTransaksi();  
    JOptionPane.showMessageDialog(this, "Berhasil disimpan",  
        "Informasi", JOptionPane.INFORMATION_MESSAGE);  
    TampilTabelSementara();  
    if(JOptionPane.showConfirmDialog(this, "Mau Cetak Struk?",  
        "konfirmasi", JOptionPane.YES_NO_OPTION) == JOptionPane.YES_OPTION) {  
        cetakstruk();  
    }  
    HapusTabelSementara();  
    awal();  
    bersih();  
    tnotrans.setText("");  
    TampilTabelSementara();  
    }  
    else  
    {  
        HapusTabelSementara();  
        awal();  
        bersih();  
        tnotrans.setText("");  
    }  
}
```

PERTEMUAN 11

Pembuatan Laporan Dan Form Penjualan Per Periode Dan Per Bulan Dan Setting Ip Untuk Pemakaian Database Bersama

A. Membuat Laporan Penjualan Per Priode

Langkah-langkah untuk membuat laporan penjualan per priode adalah sebagai berikut:

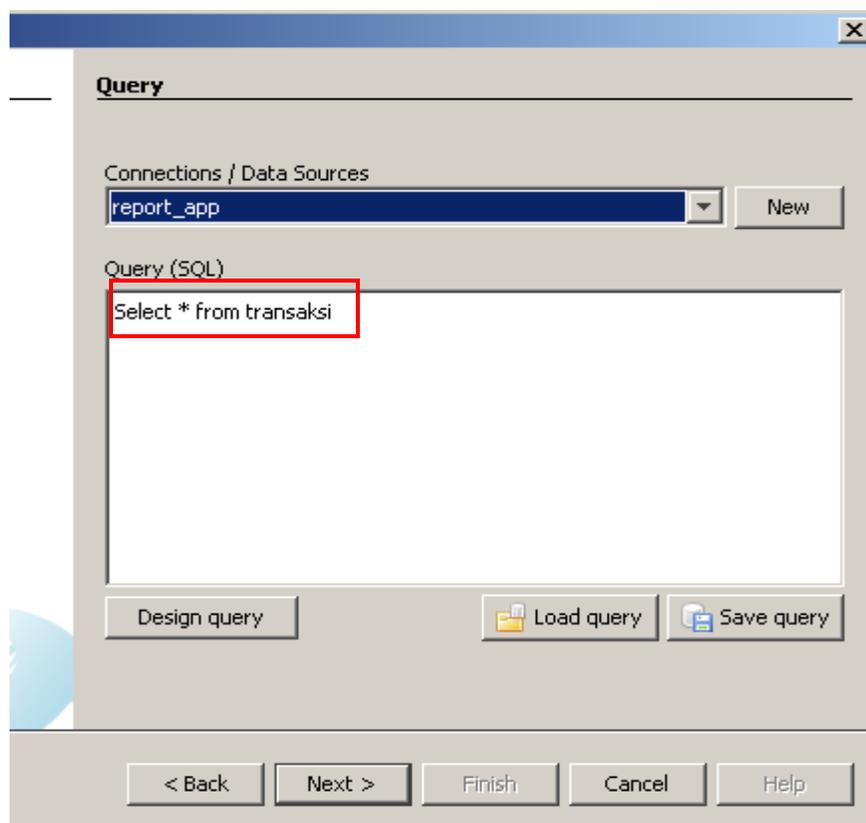
1. Ikuti langkah-langkah ke 4 sampai dengan langkah ke 6 dalam pembuatan laporan data barang pada pembahasan sebelumnya.

Template : **Blank A4**

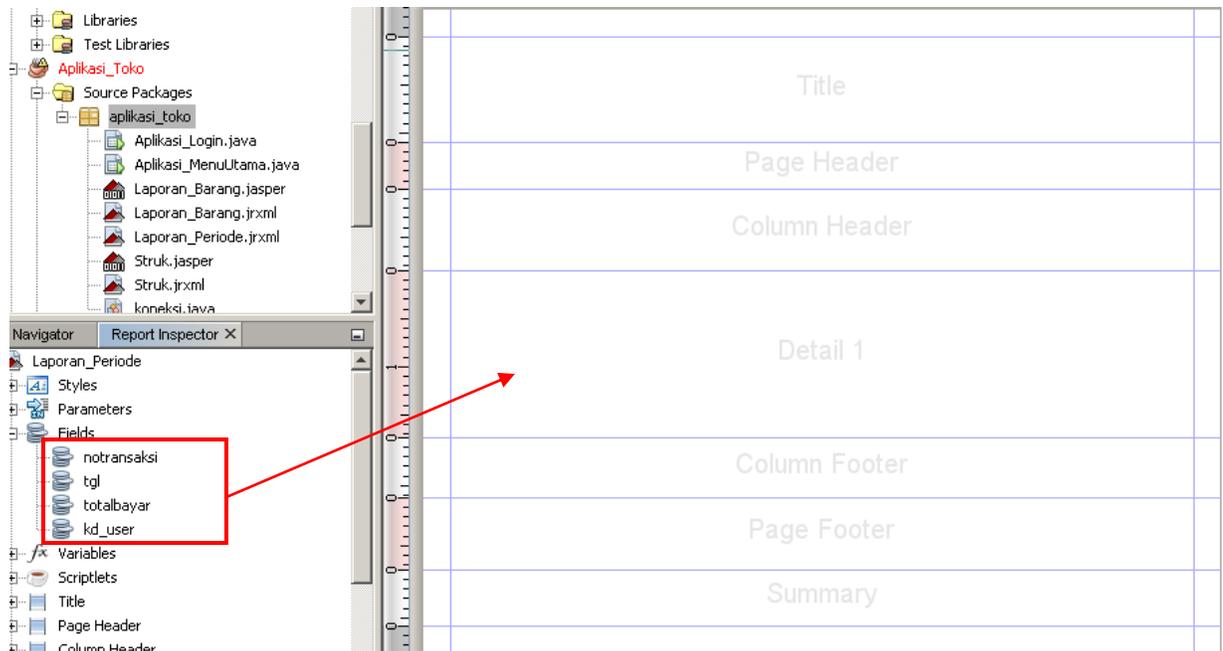
Nama report : **Laporan_Periode**

Tempat penyimpanan disamakan dengan penyimpanan laporan barang

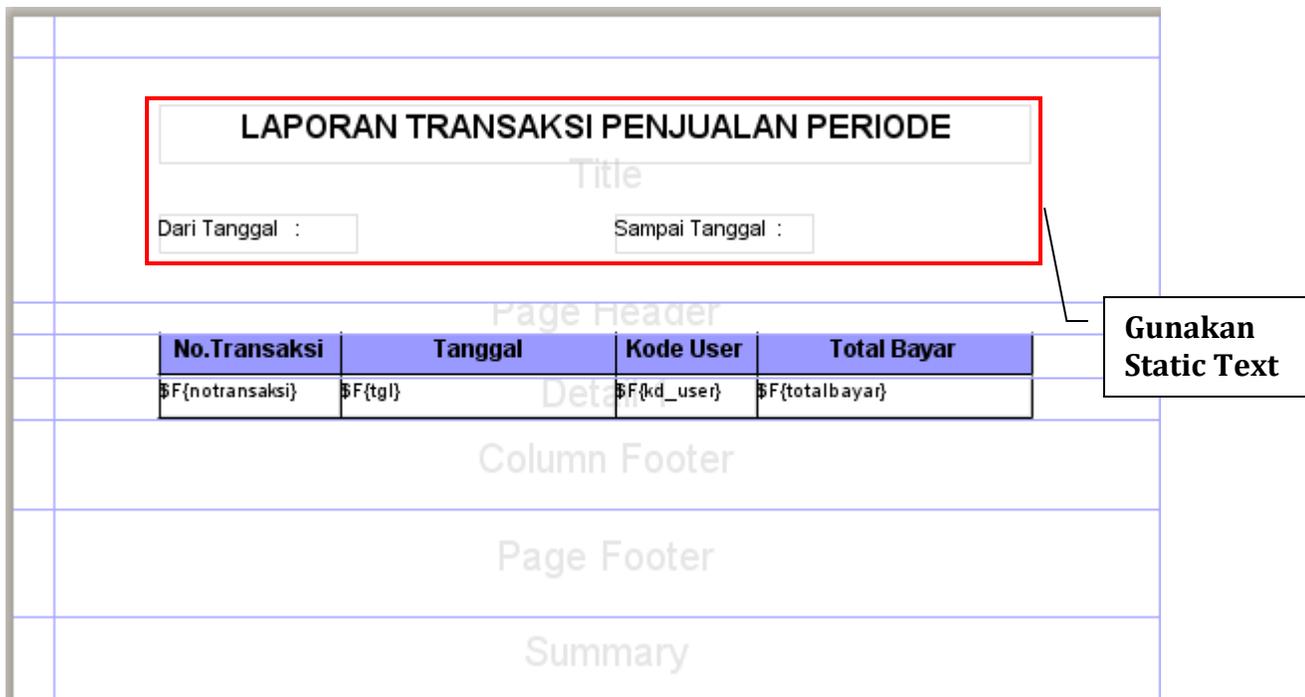
2. Kaena kita akan membuat laporan transaksi, Tuliskan query berikut, || **Next >**



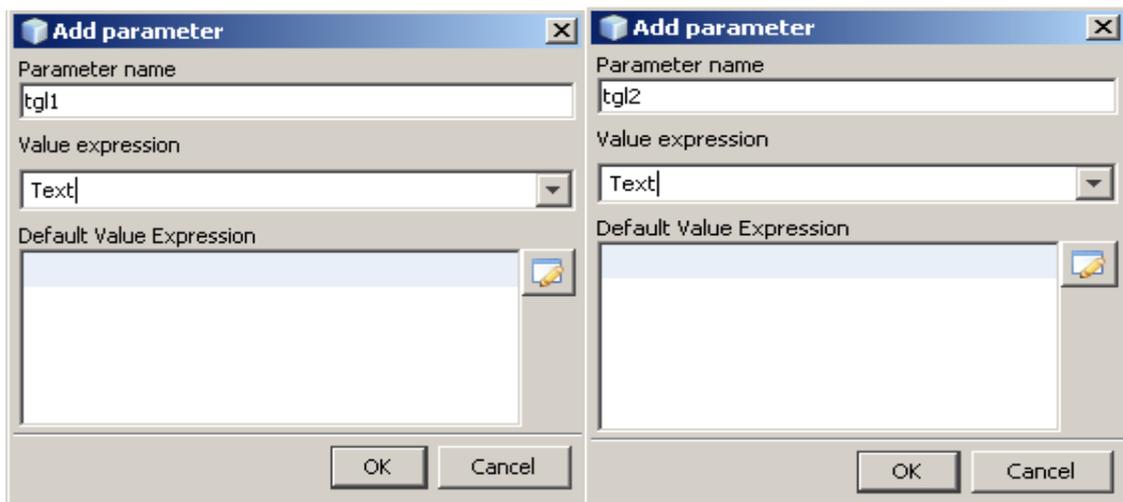
3. Pilih field yang akan ditampilkan di laporan (Pilih semua)
4. Pada jendela **Group By** klik Next saja. Setelah itu **Finish**
5. Masukkan field kedalam design report



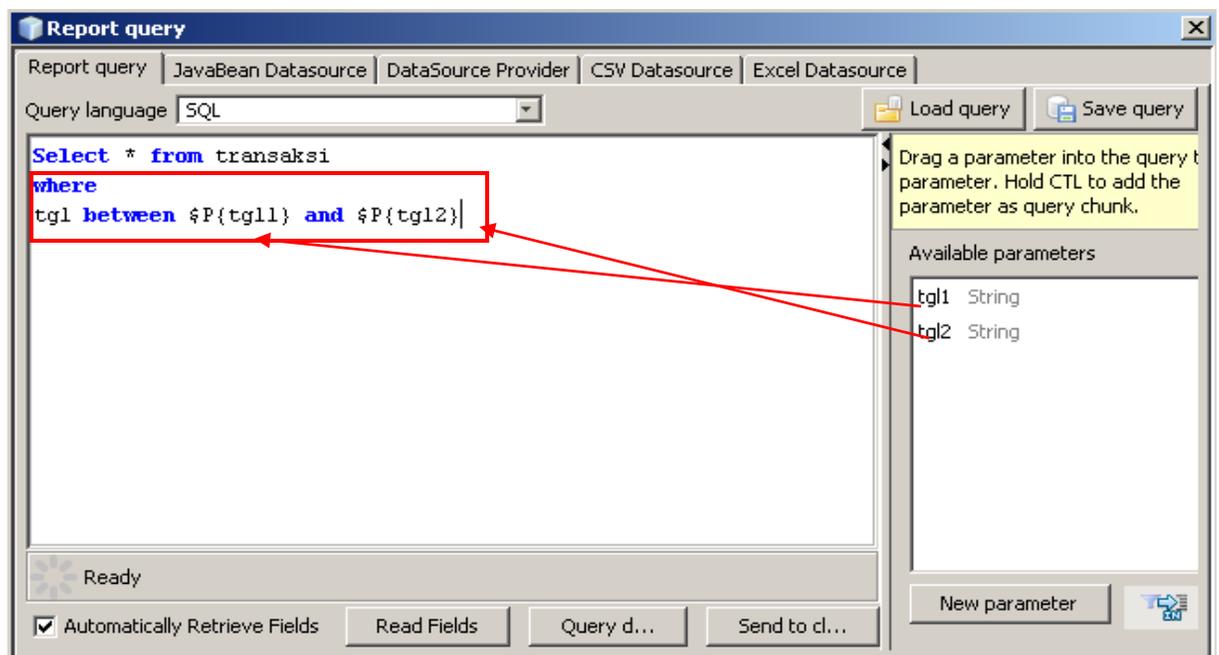
6. Desain tampilan report seperti di bawah ini



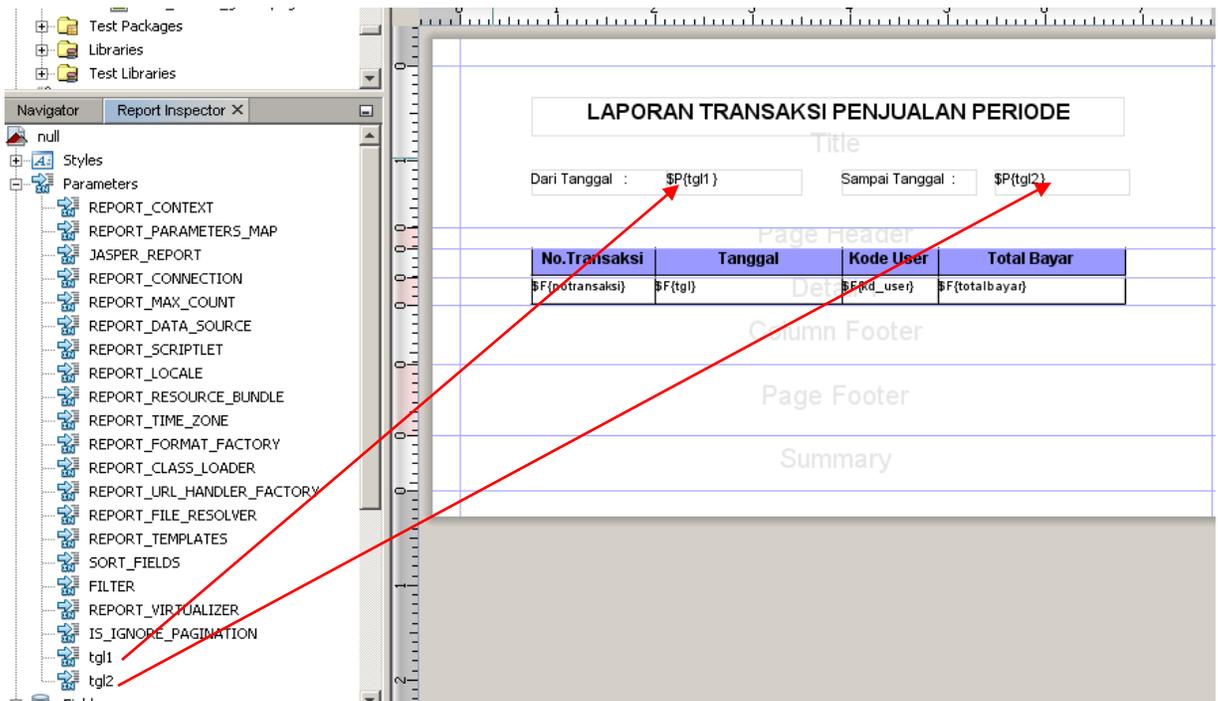
7. Langkah berikutnya adalah kita membuat parameter. Berbeda dengan sebelumnya pada pembuatan Struk, kita hanya menggunakan 1 parameter yaitu **notrans**. Kali ini kita membutuhkan 2 buah parameter, karena ada 2 buah tanggal. Yaitu tanggal awal dan tanggal akhir. Untuk tanggal awal beri nama parameter dengan **tgl1**, dan untuk tanggal akhir beri nama parameter dengan **tgl2**. Untuk Value Expression beri nilai kedua parameter dengan **Text**. Cara membuat parameter sudah dibahas saat pembuatan Struk (langkah 10), jika lupa silakan kembali lagi ke pertemuan sebelumnya



8. Edit query yang ada pada report query seperti dibawah ini, setelah itu **OK**



9. Masukkan parameter **tgl1** dan **tgl2** kedalam laporan (Click – Drag)



10. Klik Preview, maka akan muncul jendela **Parameter Prompt**, masukkan tanggal awal (tgl1) dan tanggal akhir (tgl2) yang akan kita lihat laporannya. Contoh saya masukkan **tgl1** 2018-01-01 dan **tgl2** 2018-01-30 || **OK**
11. Hasil laporan penjualan per priode

LAPORAN TRANSAKSI PENJUALAN PERIODE

Dari Tanggal : 2018-01-01

Sampai Tanggal : 2018-01-30

No.Transaksi	Tanggal	Kode User	Total Bayar
1801232	1/23/18 12:00 AM	MRU	9500000.0
1801233	1/23/18 12:00 AM		1.8E7
18012910	1/29/18 12:00 AM	MRU	7000000.0
1801294	1/29/18 12:00 AM	MRU	5000000.0
1801295	1/29/18 12:00 AM	MRU	3500000.0
1801296	1/29/18 12:00 AM	MRU	9000000.0
1801297	1/29/18 12:00 AM	MRU	3500000.0
1801298	1/29/18 12:00 AM	MRU	2750000.0
1801299	1/29/18 12:00 AM		4500000.0

12. Untuk script pemanggilan laporan per priode ada di Aplikasi_Laporan_Penjualan pada pembahasan selanjutnya.

B. Membuat Laporan Penjualan Per Bulan

Pada dasarnya membuat laporan penjualan perbulan dan laporan penjualan per priode adalah sama. Langkah-langkah pembuatan laporan penjualan perbulan adalah sebagai berikut:

- Ikuti langkah 1-5 saat pembuatan laporan per priode

Template : **Blank A4**

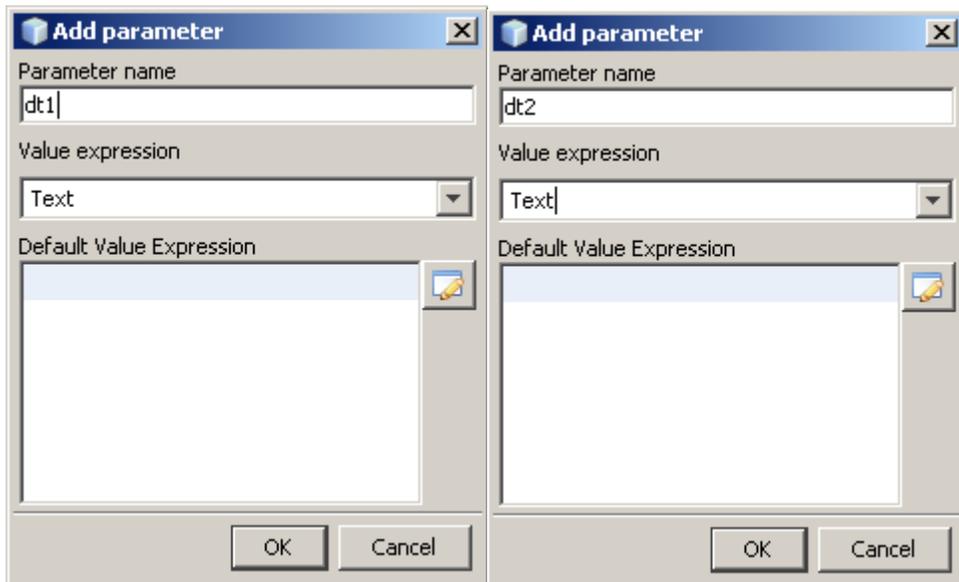
Nama report : **Laporan_Perbulan**

- Desain laporan seperti berikut:

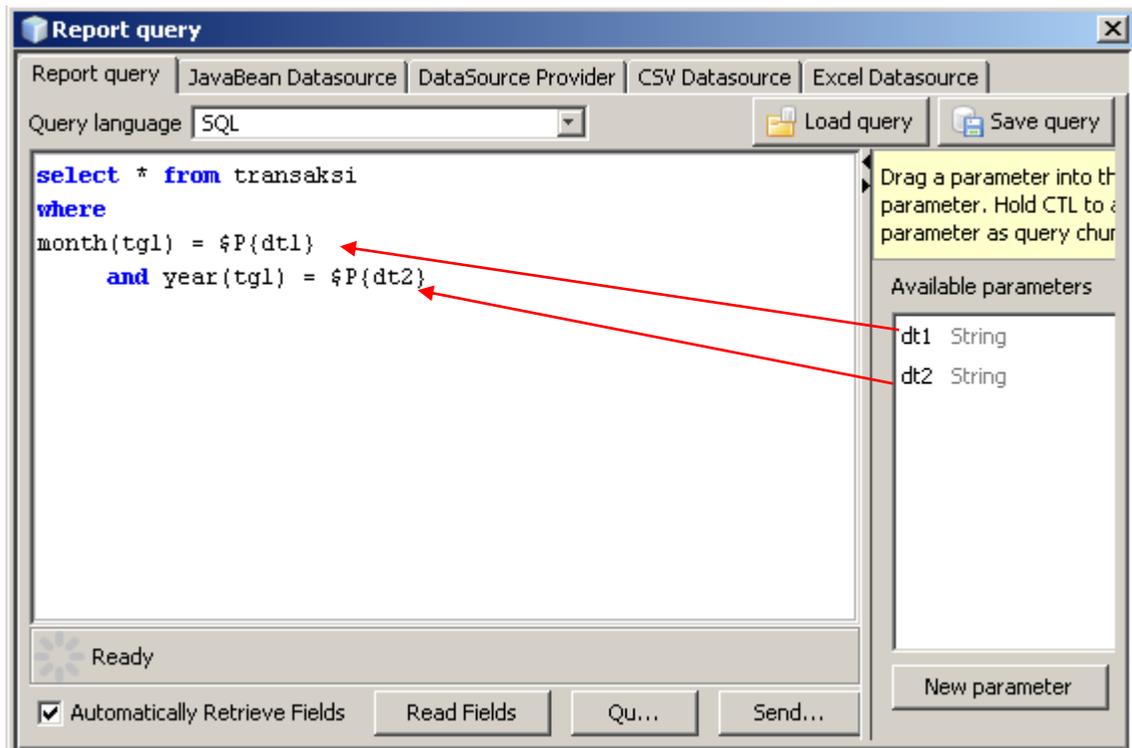
LAPORAN TRANSAKSI PENJUALAN BULANAN			
Bulan : Tahun :			
No.Transaksi	Tanggal	Kode User	Total Bayar
#{notransaksi}	#{tgl}	#{kd_user}	#{totalbayar}

Gunakan Static Text

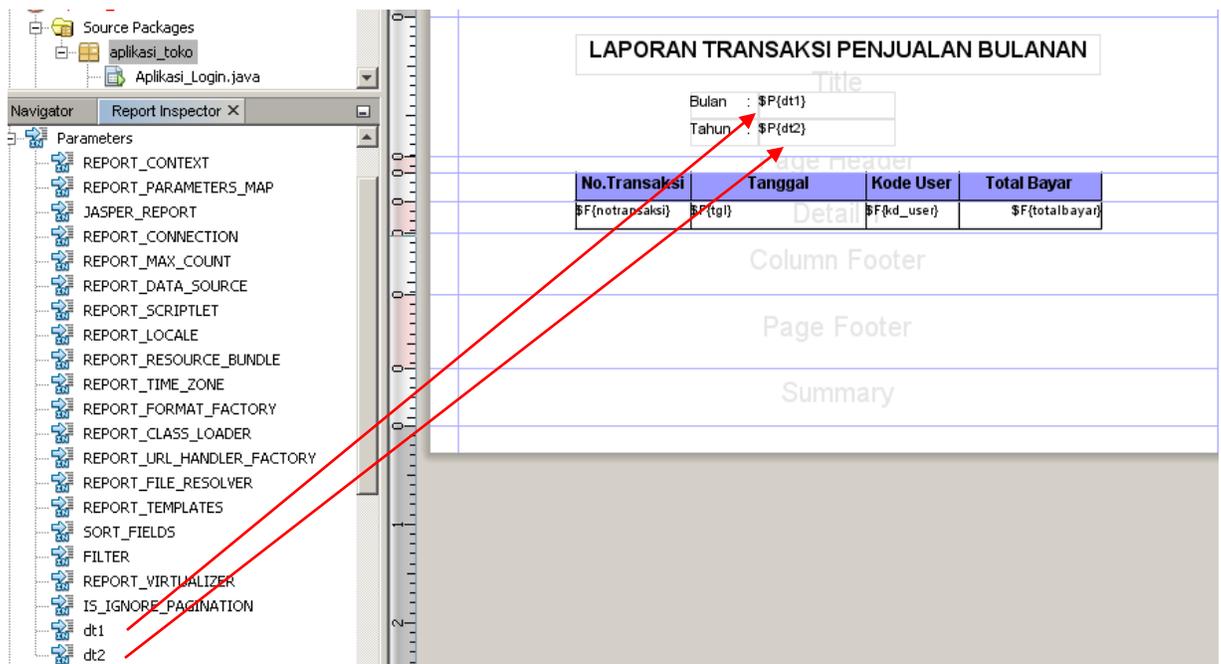
3. Buat 2 buah parameter dengan nama dt1 dan dt2. (dt1 untuk bulan, dt2 untuk tahun)



4. Edit query yang ada pada report query seperti dibawah ini, setelah itu **OK**



5. Masukkan parameter **dt1** dan **dt2** kedalam laporan (Click - Drag)



6. Klik Preview, maka akan muncul jendela **Parameter Prompt**, masukkan Bulan (dt1) dan Tahun (dt2) yang akan kita lihat laporannya. Contoh saya masukkan **dt1: 1 , dt2: 2018 || OK**
7. Hasil laporan penjualan Perbulan:

LAPORAN TRANSAKSI PENJUALAN BULANAN

Bulan : 2
Tahun : 2018

No.Transaksi	Tanggal	Kode User	Total Bayar
18020510	2/5/18 12:00 AM	MRU	8500000.0
FK-1802511	2/5/18 12:00 AM	MRU	8500000.0
FK-1802512	2/5/18 12:00 AM	MRU	9500000.0
FK-1802513	2/6/18 12:00 AM	MRU	7750000.0
FK-1802514	2/7/18 12:00 AM	MRU	9500000.0

8. Untuk script pemanggilan laporan per Bulan ada di Aplikasi_Laporan_Penjualan pada pembahasan selanjutnya

C. Membuat Form Laporan Penjualan

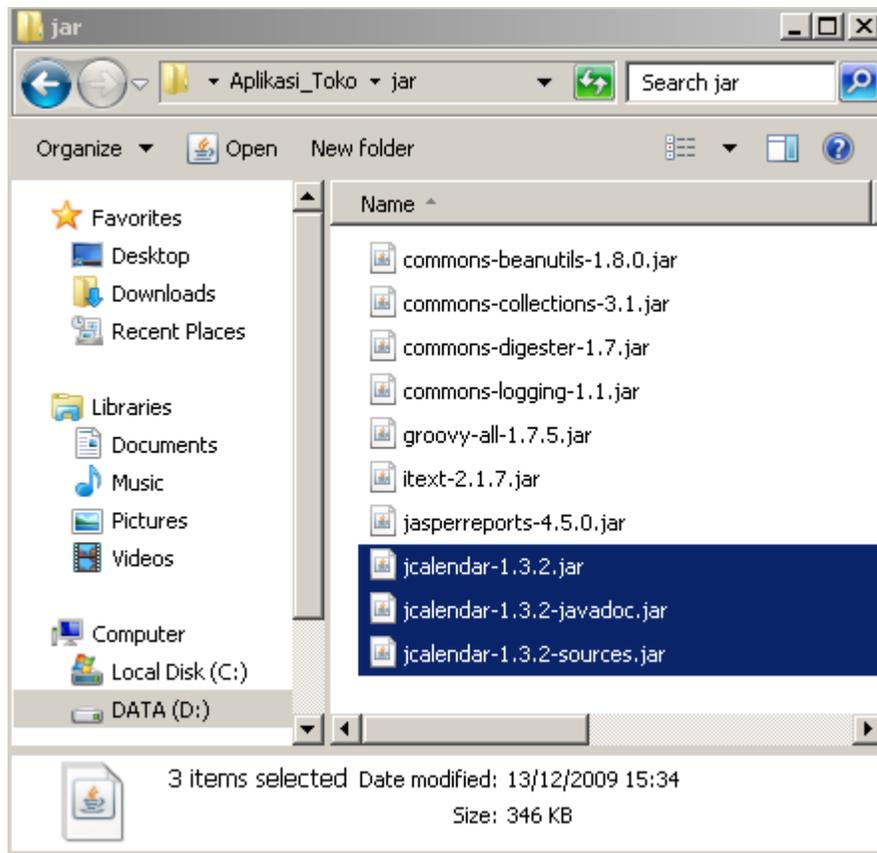
Desain Form Laporan Penjualan

Running Form Laporan Penjualan

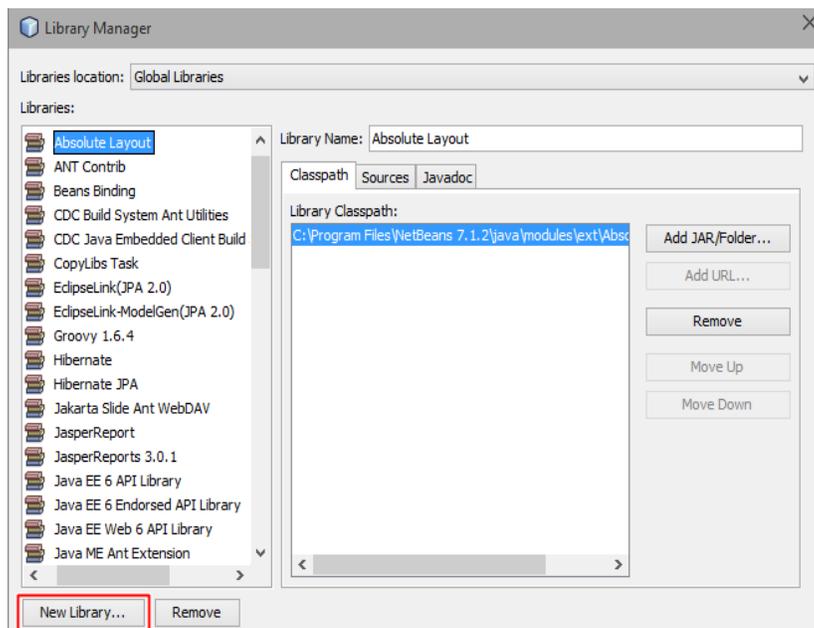
Untuk membuat form laporan transaksi, pada program ini menggunakan komponen **jDateChooser** dan **jYearChooser**. Untuk menambahkan komponen-komponen tersebut pada Palette, kita membutuhkan library **JCalendar**. Silakan anda cari library JCalendar atau silakan klik link [disini](#).

Jika sudah selesai didownload library JCalendar nya, berikutnya kita akan menambahkan library tersebut agar bisa masuk kedalam Palette. Berikut langkah-langkahnya

1. Extract file JCalendar yang sudah didownload. Copy file yang ada di dalamnya ke dalam folder **jar**, pada project **Aplikasi_Toko**

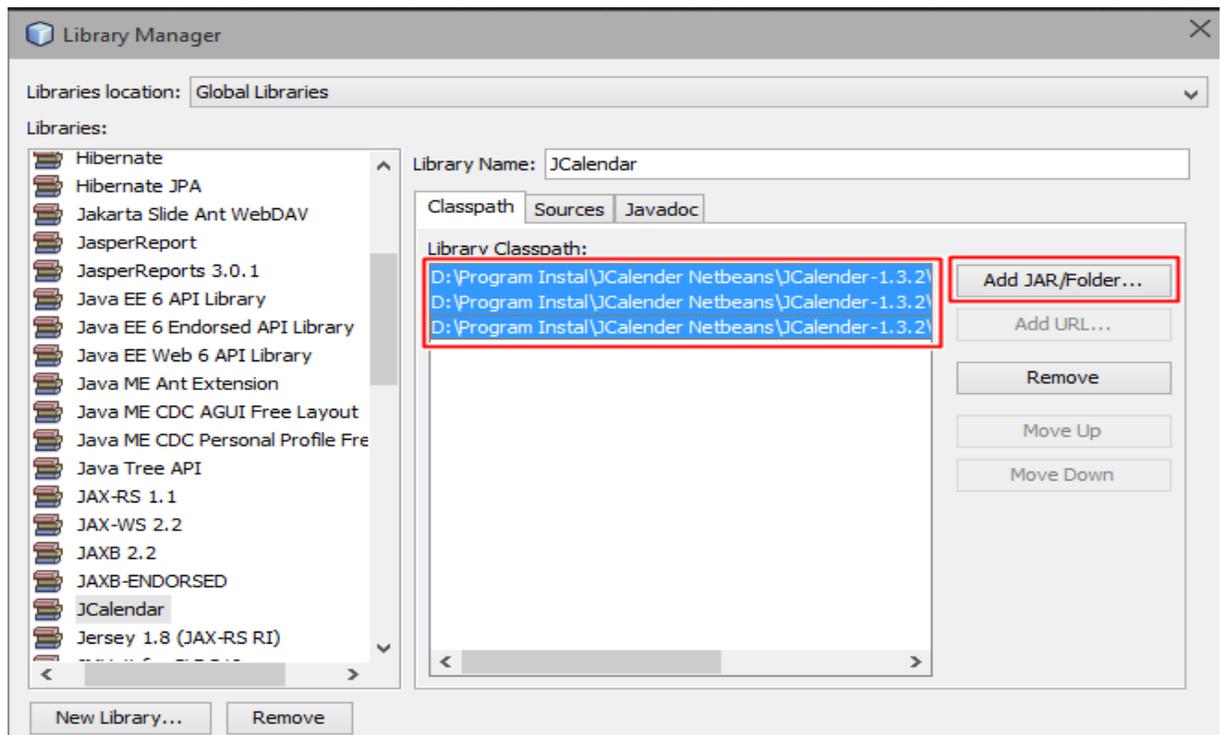


2. Pastikan netbeans dalam keadaan terbuka, pilih Tools - Libraries
3. Klik **New Library**

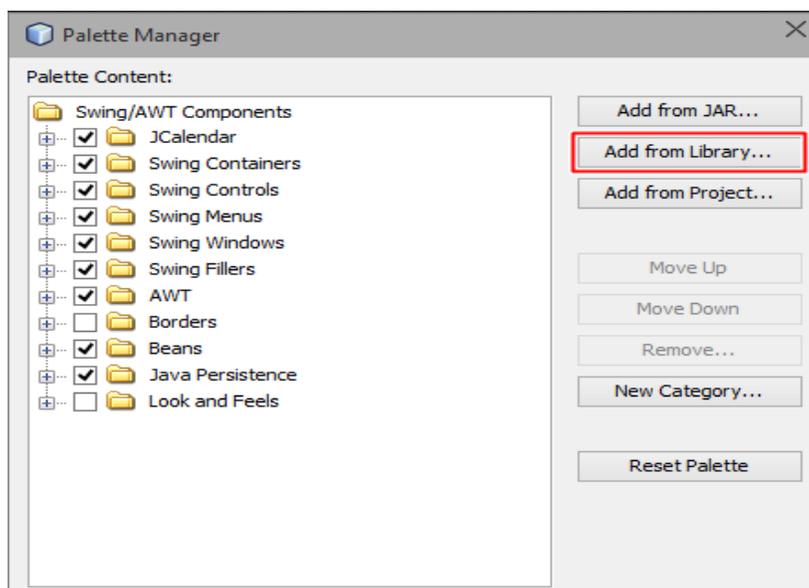


4. Beri nama library dengan **JCalendar**

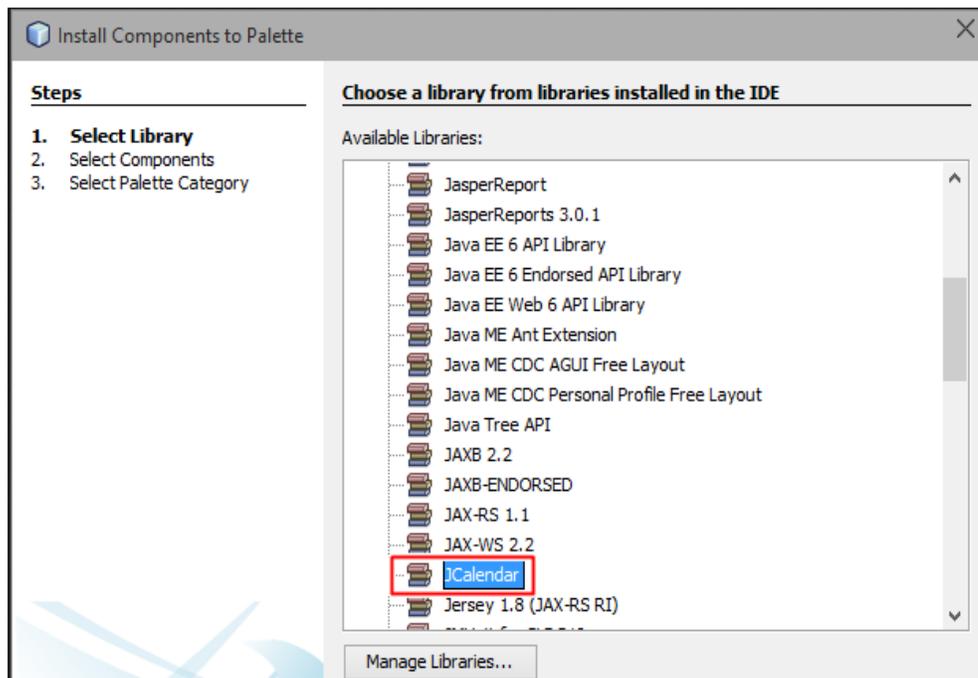
- Arahkan pada library **JCalendar**. Klik **Add Jar/Folder**. Cari file library JCalendar yang sudah diextract sebelumnya dan masukkan ketiga file .jar tersebut || **OK**



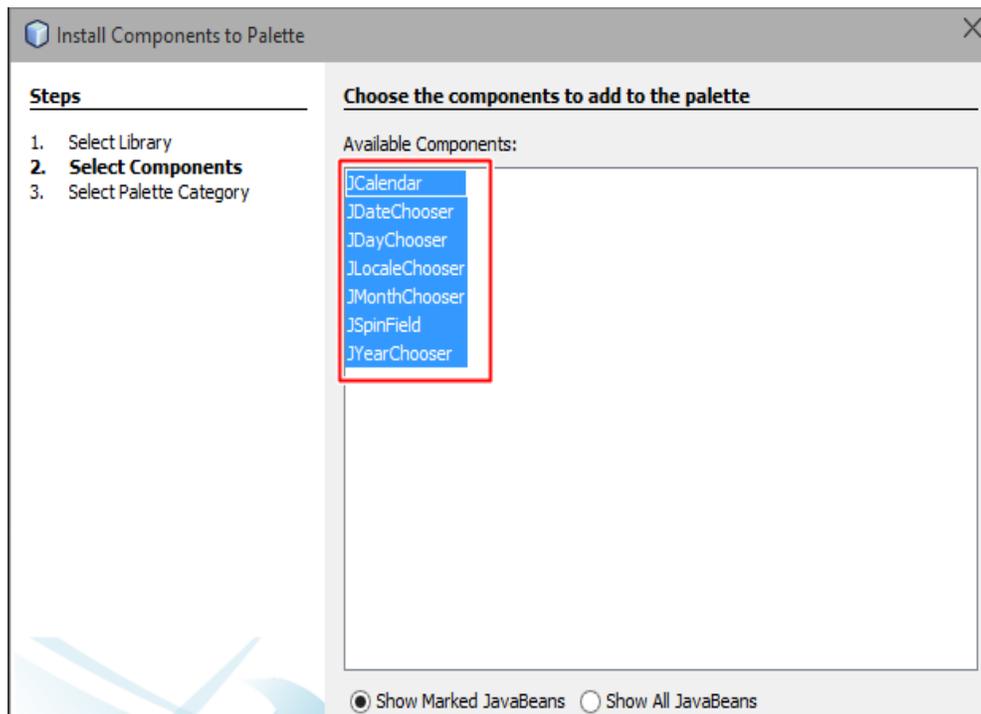
- Klik menu **Tools – Palette – Swing/AWTComponents**
- Klik **New Category** untuk membuat kategori baru untuk JCalendar, beri nama kategori dengan **JCalendar**
- Klik **Add from Library**



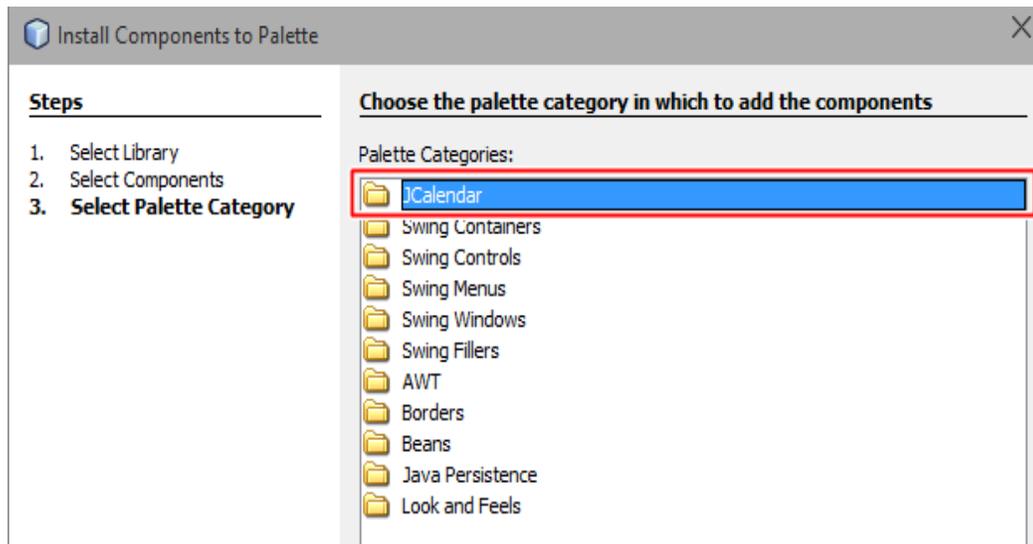
9. Pilih Library JCalendar yang sudah kita buat || Next



10. Pilih semua komponen || Next



11. Pilih JCalendar pada Palette Category || Finish



12. Periksa palette, jika berhasil akan muncul komponen JCalendar



Setelah kita berhasil menambahkan komponen JCalendar, langkah berikutnya adalah kita desain form laporan penjualan. Silakan anda desain form menggunakan JFrame Form di dalam project Aplikasi_Toko. Beri nama for tersebut dengan **Aplikasi_Laporan_Penjualan**.

Spesifikasi properties komponen form
Aplikasi_Laporan_Penjualan

No	Komponen	Properties
1	JDateChooser1	Nama Variabel : dctglawal
2	JDateChooser2	Nama Variabel : dctglakhir
3	JComboBox1	Nama Variabel : cmbbulan Model : 1,2,3,.....,12
4	JYearChooser1	Nama Variabel : yctahun
5	JButton1	Nama Variabel : bt cetakperiode Text : Cetak
6	JButton2	Nama Variabel : bt cetakbulan Text : Cetak
7	JButton3	Nama Variabel : bt_keluar Text : Keluar

Script program untuk Form Aplikasi_Laporan_Penjualan adalah sebagai berikut:

1. Atur koneksi dan file-file yang harus di import

```

package aplikasi_toko;
import javax.swing.JOptionPane;
import net.sf.jasperreports.engine.JasperCompileManager;
import net.sf.jasperreports.engine.design.JasperDesign;
import net.sf.jasperreports.engine.xml.JRXmlLoader;
import net.sf.jasperreports.view.JasperViewer;
import java.text.SimpleDateFormat;
import java.io.File;
import java.util.HashMap;
import javax.swing.DefaultListModel;
import net.sf.jasperreports.engine.JasperFillManager;
import net.sf.jasperreports.engine.JasperPrint;
import net.sf.jasperreports.engine.JasperReport;

/**
 *
 * @author Maruloh
 */
public class Aplikasi_Laporan_Penjualan extends javax.swing.JFrame {
    koneksi kon=new koneksi();
    JasperReport jasperReport;
    JasperDesign jasperDesign;
    JasperPrint jasperPrint;

    /**
     * Creates new form Aplikasi_Laporan_Penjualan
     */
    public Aplikasi_Laporan_Penjualan() {
        initComponents();
        kon.setKoneksi();
    }

```

2. Klik kanan Pada tombol **btcetakperiode**, pilih **Event – Action – ActionPerformed**, ketikkan script berikut:

```
private void btcetakperiodeActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String tgl1 = new SimpleDateFormat("yyyy-MM-dd").format(dctglawal.getDate());
    String tgl2 = new SimpleDateFormat("yyyy-MM-dd").format(dctglakhir.getDate());
    try {
        File file = new File("src/aplikasi_toko/Laporan_Periode.jrxml");
        jasperDesign = JRXmlLoader.load(file);
        HashMap param = new HashMap();
        param.put("tgl1", tgl1);
        param.put("tgl2", tgl2);
        jasperReport = JasperCompileManager.compileReport(jasperDesign);
        jasperPrint = JasperFillManager.fillReport(jasperReport, param, kon.setKoneksi());
        JasperViewer.viewReport(jasperPrint, false);
    } catch (Exception e) {
        JOptionPane.showMessageDialog(null, e.getMessage());
    }
}
```

3. Klik kanan Pada tombol **btcetakbulan**, pilih **Event – Action – ActionPerformed**, ketikkan script berikut:

```
private void btcetakbulanActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try {
        String NamaFile = "src/aplikasi_toko/Laporan_Bulanan.jasper";
        HashMap parameter = new HashMap();
        parameter.put("dt1", cmbbulan.getSelectedItemAt());
        parameter.put("dt2", yctahun.getYear());
        JasperPrint print = JasperFillManager.fillReport(NamaFile, parameter, kon.setKoneksi());
        JasperViewer.viewReport(print, false);
    } catch (Exception ex) {
        System.out.println(ex);
    }
}
```

4. Klik kanan Pada tombol **btkeluar**, pilih **Event – Action – ActionPerformed**, ketikkan script berikut:

```
private void btkeluarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    dispose();
}
```

D. Setting IP untuk Pemakaian Database Bersama

Untuk dapat melakukan pemakaian database bersama, maka kita memerlukan sebuah jaringan lokal. Bisa menggunakan kabel jaringan, atau bisa juga menggunakan intranet dalam 1 ruangan, bisa juga menggunakan share wifi menggunakan Handphone (HP). Oke, dalam kasus ini agar mudah kita gunakan share wifi dengan menggunakan HP. Ada 2 buah komputer/laptop yang akan kita pakai (sebagai contoh). Satu komputer sebagai server (punya database dan program aplikasi penjualan), satu komputer sebagai client (cukup memiliki program aplikasi penjualannya saja).

Jadi untuk dapat menggunakan database bersama dalam jaringan local di java, yang harus dipersiapkan adalah:

- 2 buah komputer/laptop (bisa juga lebih), satu sebagai server, selebihnya sebagai client
- Jaringan local (bisa menggunakan kabel jaringan, intranet, atau share wifi lewat HP)

Langkah-langkah setting Jaringan dan IP dalam penggunaan Database bersama adalah sebagai berikut:

1. Silakan Share wifi melalui HP.
2. Koneksikan 2 komputer ke wifi tersebut, jika sudah terkoneksi maka masing-masing komputer akan memiliki IP. Sebagai contoh komputer 1 sebagai server memiliki IP **192.168.1.1**, sedangkan komputer 2 sebagai client memiliki IP **192.168.1.2**
3. Masuk kedalam PhpMyAdmin (untuk komputer server), dan buat user Privileges agar MySQL dapat diakses oleh komputer client.
4. Pilih **Privileges/User**, kemudian **Add a new User/Add user**



Users overview

	User	Host	Password	Global privileges	Grant	Action
<input type="checkbox"/>	Any	%	--	USAGE	No	Edit Privileges Export
<input type="checkbox"/>	Any	linux	No	USAGE	No	Edit Privileges Export
<input type="checkbox"/>	Any	localhost	No	USAGE	No	Edit Privileges Export
<input type="checkbox"/>	pma	localhost	No	USAGE	No	Edit Privileges Export
<input type="checkbox"/>	root	linux	No	ALL PRIVILEGES	Yes	Edit Privileges Export
<input type="checkbox"/>	root	localhost	No	ALL PRIVILEGES	Yes	Edit Privileges Export

Check All / Uncheck All

Add user

5. Isi data yang disediakan, User name : **root**, Host : **%**, password : **admin**, Database for user : **Grant all privileges**, Global Privileges : **Check All** || **Add User**

Add user

Add user

Login Information

User name: Use text field:

Host: Use text field:

Password: Use text field:

Re-type:

Generate password:

Database for user

None

Create database with same name and grant all privileges

Grant all privileges on wildcard name (username_%)

Global privileges

6. Buka class koneksi, ubah localhost menjadi alamat IP server yaitu **192.168.1.1**, passwordnya : **admin**

```
package aplikasi_toko;
import java.sql.*;
import javax.swing.JOptionPane;

/**
 *
 * @author Maruloh
 */
public class koneksi {
    Connection conn;
    Statement st;
    ResultSet rs;

    public Connection setKoneksi()
    {
        try{
            Class.forName("com.mysql.jdbc.Driver");
            conn=DriverManager.getConnection("jdbc:mysql://192.168.1.1/toko_db","root","admin");
            st=conn.createStatement();
        }
        catch(Exception e){
            JOptionPane.showMessageDialog(null,"Koneksi Gagal : " +e);
        }
        return conn;
    }
}
```

7. Silakan coba jalankan program melalui komputer client. Jika berhasil maka database pada komputer server akan dapat kita akses.

PERTEMUAN 12

QUIS INDIVIDU

Pada pertemuan ini, dosen pengajar memberikan tes individu berupa studi kasus kepada mahasiswa untuk mengetahui kemampuan mahasiswa. Nilai individu ini akan mempengaruhi nilai akhir dari project. Minimal studi kasus memiliki **2 tabel (1 Master, 1 Transaksi)** dan **1 form transaksi**

PERTEMUAN 13-16
Presentasi Final Project

Pada pertemuan ini dilakukan presentasi PROJECT yang dilakukan secara kelompok