**Tugas Responsi**

**NAMA : YULIANTO BAYU PRASETYO**

**NIM : 5200411559**

Bab 1

1. Alasan membuat form dengan isian data demikan

Alasan membuat form dengan isi data tersebuta adalah karena kebutuhan data distriutor tersebut pasti membuthkan nama produk, dan penerima adalah yang menerima pasokan produk, tentu juga membutuhkan jumlah produk yang dipasok berserta harga per unitnya, dan wajib memberikan data supliernya dan juga perlu untuk mencatat tanggalnya, jenis dan catatan adalah informasi tambahan.

1. Atribut yang ada di form

Terdapat jpane dan label untuk memberi judul, dan pada bagian form terdapat 6 text fild dan pada tanggal value dari text field tersebut akan di gabungkan , kemudian ada 2 combox yang datanya sudah pasti agar saat input data meminimalisir kesalahan penulisan, dan radio button yang digabungkan dengan radio group menggunakannya karena sudah pasti hanya ada 2 isi, tujuannya sama dengan saat menggunkan combo box agar meminimalisir kesalahan penulisan dan yang terkhir ada text area sebai catatan, karena catatan biasanya bisa sangat panjang akan tidak baik jika menggukan jtextield

Bab 2

Source Code Tiap Tombol

1. Simpan

private void btnSimpanActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

try {

Connection conn = Koneksi.getKoneksi();

String sql = "INSERT INTO siberkom (nama\_barang, jumlah\_unit, jenis\_produk, penerima, supplier, harga\_perunit, catatan, tanggal)"+

" VALUES (?, ?, ?, ?, ?, ?, ?, ?)";

PreparedStatement ps = conn.prepareStatement(sql);

String selectedRadio = "";

if (rdLokal.isSelected()) {

selectedRadio = "lokal";

} else if (rdImpor.isSelected()) {

selectedRadio = "impor";

}

ps.setString(1, txtNamab.getText());

ps.setInt(2, Integer.parseInt(txtJumlahu.getText()));

ps.setString(3, selectedRadio);

ps.setString(4, cmbPenerima.getSelectedItem().toString());

ps.setString(5, cmbSupplier.getSelectedItem().toString());

ps.setInt(6, Integer.parseInt(txtHargap.getText()));

ps.setString(7, txaCatatan.getText());

ps.setString(8, txtTahun.getText()+"-"+txtBulan.getText()+"-"+txtTanggal.getText());

int insertedRow = ps.executeUpdate();

loadTable();

if (insertedRow > 0) {

JOptionPane.showMessageDialog(this, "Berhasil menyimpan data");

} else {

JOptionPane.showMessageDialog(this, "Gagal menyimpan data");

}

bersihkan();

} catch (SQLException e) {

System.out.println("Terjadi masalah : " + e);

}

}

1. Edit

private void btnEditActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

try {

Connection conn = Koneksi.getKoneksi();

String sql = "UPDATE siberkom SET "+

"nama\_barang = ?, "+

"jumlah\_unit = ?, "+

"jenis\_produk = ?, "+

"penerima = ?, "+

"supplier = ? , "+

"harga\_perunit = ?, "+

"catatan = ?, "+

"tanggal = ? "+

"WHERE id = ?";

PreparedStatement ps = conn.prepareStatement(sql);

String selectedRadio = "";

if (rdLokal.isSelected()) {

selectedRadio = "lokal";

} else if (rdImpor.isSelected()) {

selectedRadio = "impor";

}

ps.setString(1, txtNamab.getText());

ps.setInt(2, Integer.parseInt(txtJumlahu.getText()));

ps.setString(3, selectedRadio);

ps.setString(4, cmbPenerima.getSelectedItem().toString());

ps.setString(5, cmbSupplier.getSelectedItem().toString());

ps.setInt(6, Integer.parseInt(txtHargap.getText()));

ps.setString(7, txaCatatan.getText());

ps.setString(8, txtTahun.getText()+"-"+txtBulan.getText()+"-"+txtTanggal.getText());

ps.setInt(9, Integer.parseInt(txtId.getText()));

int insertedRow = ps.executeUpdate();

loadTable();

if (insertedRow > 0) {

JOptionPane.showMessageDialog(this, "Berhasil mengubah data");

} else {

JOptionPane.showMessageDialog(this, "Gagal mengubah data");

}

bersihkan();

} catch (SQLException e) {

System.out.println("Terjadi masalah : " + e);

}

}

1. Delete

private void btnHapusActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

try {

int confirm = JOptionPane.showConfirmDialog(this,"Yakin akan menghapus data ? ","Peringatan !!!",JOptionPane.YES\_NO\_OPTION);

if (confirm == 0 ) {

Connection conn = Koneksi.getKoneksi();

String sql = "DELETE FROM siberkom WHERE id = ?";

PreparedStatement ps = conn.prepareStatement(sql);

ps.setInt(1, Integer.parseInt(txtId.getText()));

int insertedRow = ps.executeUpdate();

ps.close();

loadTable();

if (insertedRow > 0) {

JOptionPane.showMessageDialog(this, "Berhasil menyimpan data");

} else {

JOptionPane.showMessageDialog(this, "Gagal menyimpan data");

}

}

bersihkan();

} catch (SQLException e) {

System.out.println("Terjadi Masalah : " + e);

}

}

1. Bersihkan

private void btnClearActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

bersihkan();

}

private void bersihkan() {

txtId.setText("ID Terisi otomatis");

txtNamab.setText("");

txtJumlahu.setText("");

rdgJenis.clearSelection();

cmbPenerima.setSelectedIndex(0);

cmbSupplier.setSelectedIndex(0);

txtHargap.setText("");

txaCatatan.setText("");

txtTanggal.setText("");

txtBulan.setText("");

txtTahun.setText("");

}

1. Keluar

private void btnKeluarActionPerformed(java.awt.event.ActionEvent evt) {

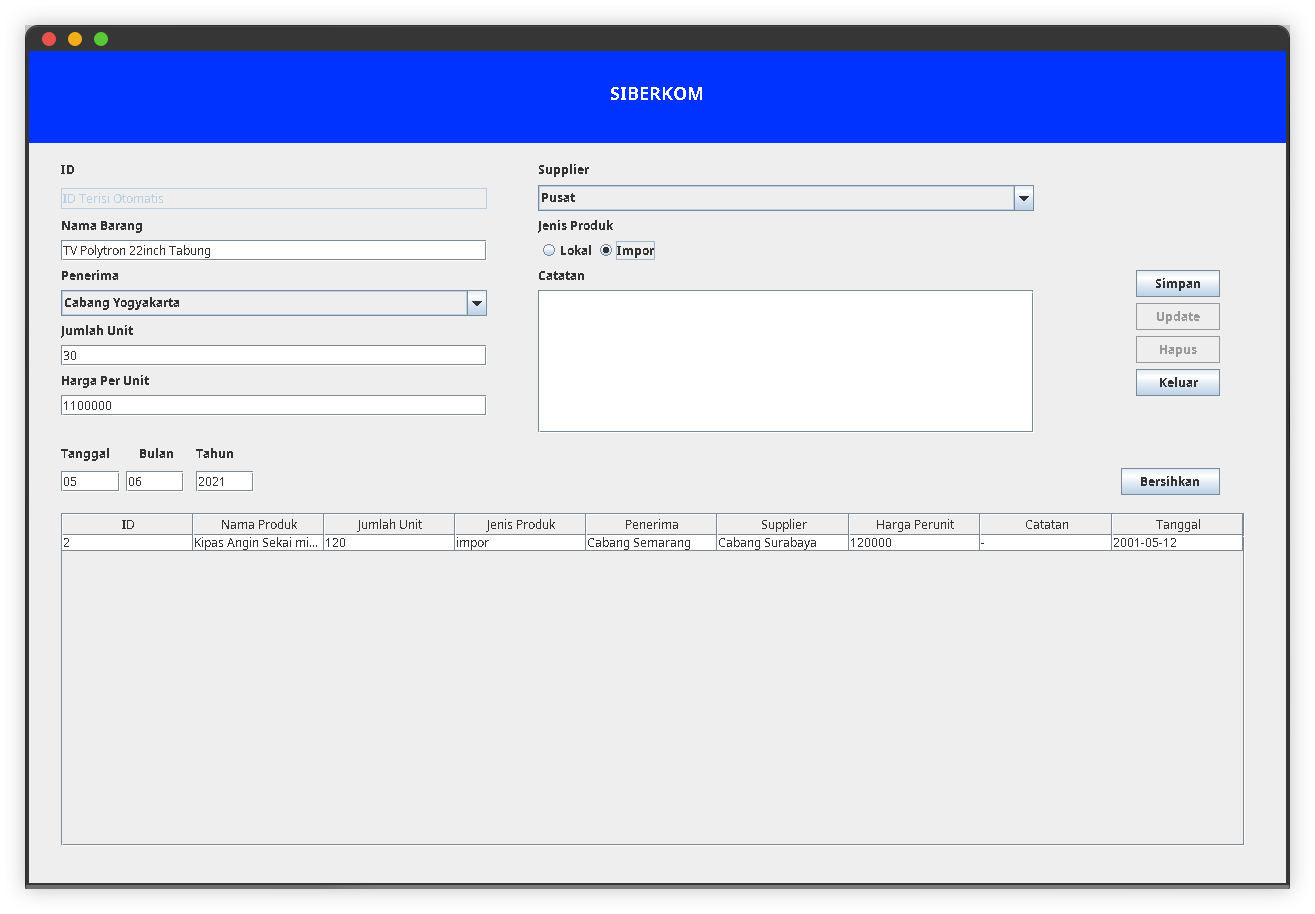
// TODO add your handling code here:

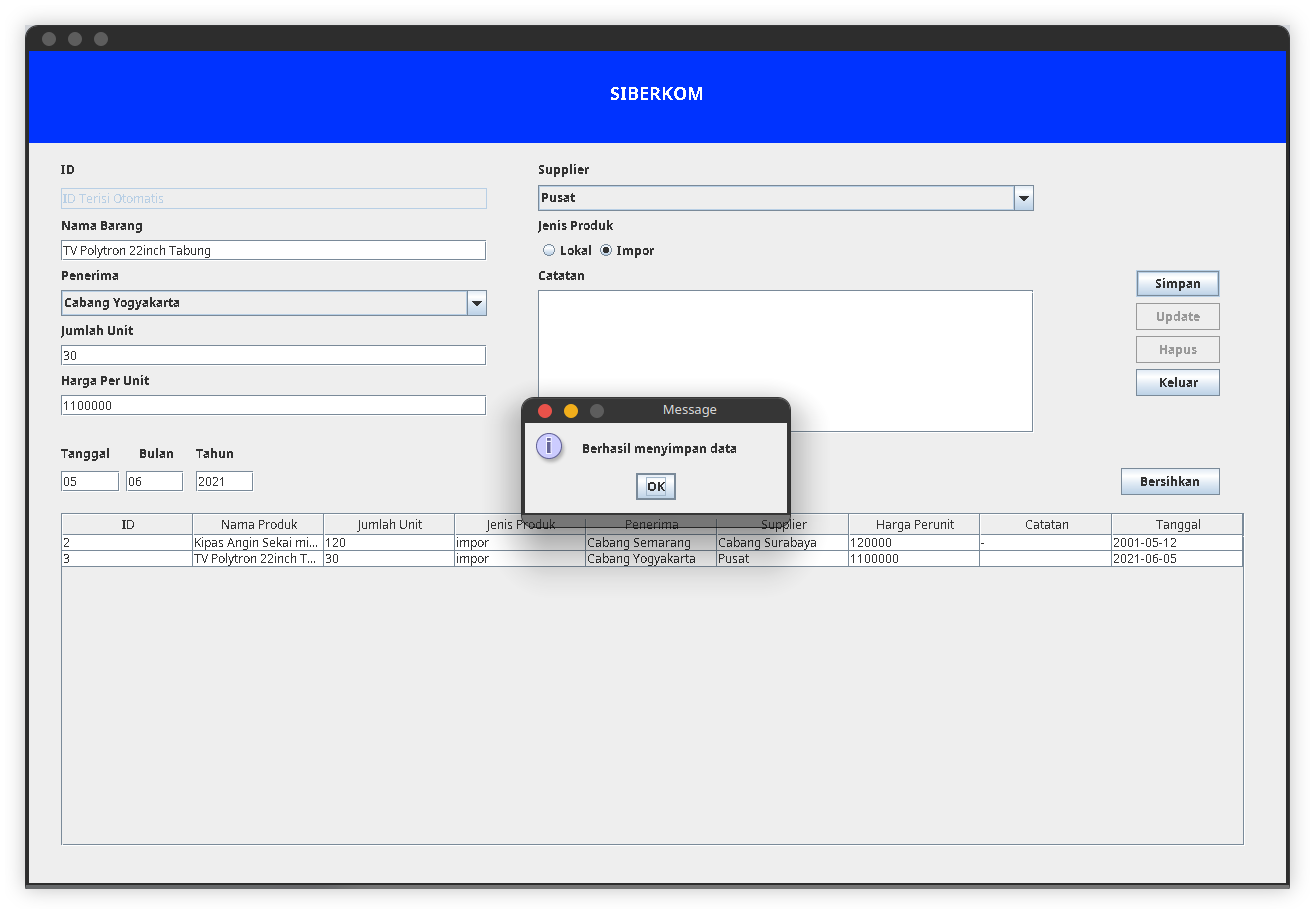
System.exit(0);

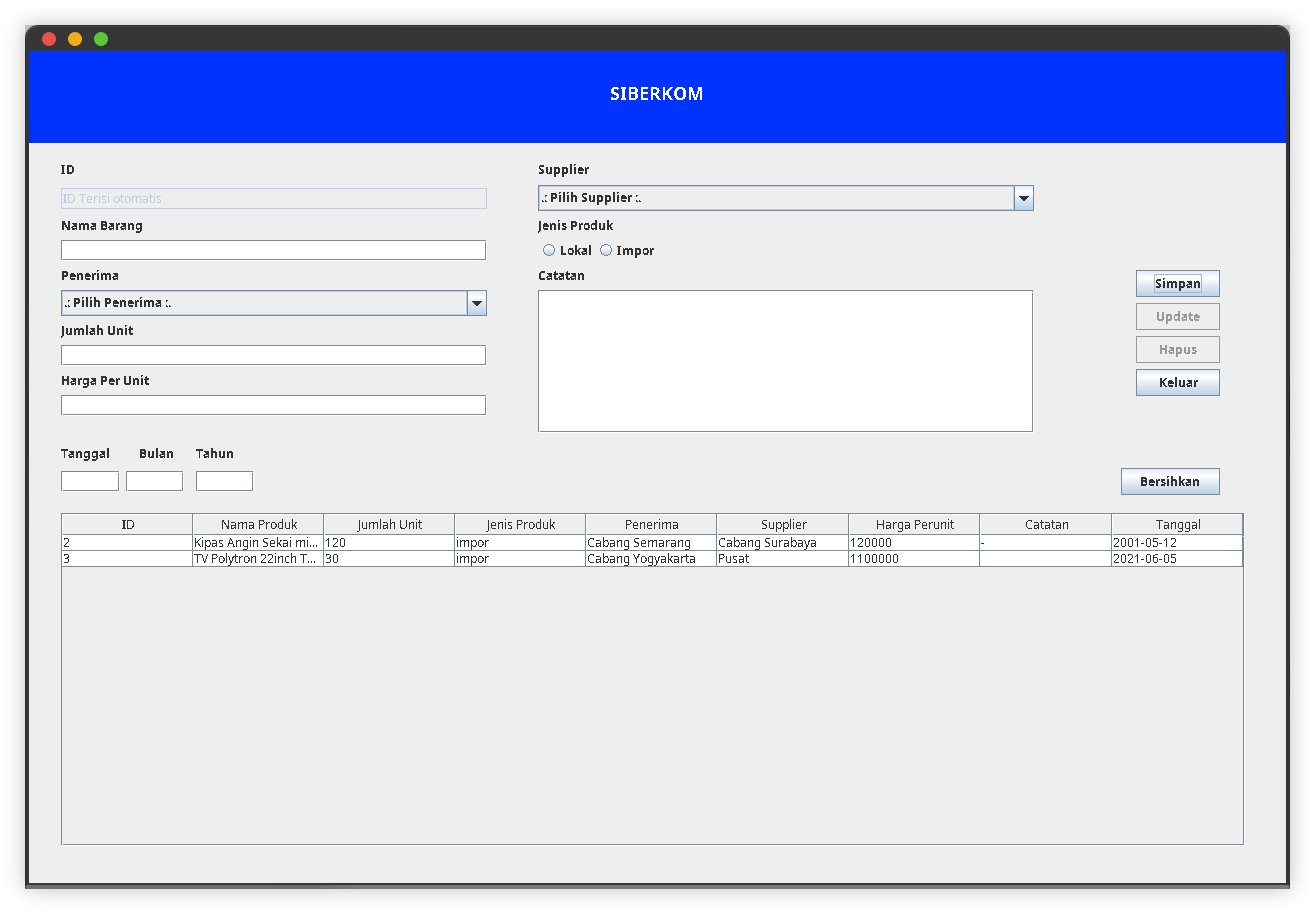
}

Bab 3

1. Tampilan







1. Struktur table

