

**LAPORAN PRAKTIKUM  
PEMROGRAMAN LANJUT**



**Progres Kelompok 1 (Coffe Shop Sarena)**

[alvinsyahri/Kelompok1\\_CoffeShopSarena](#)

**Oleh :**

Nama : Alvin Syahri (2020573010006)  
: Al Khaidar (2020573010005)  
: Adillah (2020573010002)  
: Saskia Febrianti (2020573010064)  
Kelas : TI 2B  
Dosen Pembimbing : Musta'inul Abdi, S.S.T.,M.Kom

**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNOLOGI INFORMASI DAN KOMPUTER  
POLITEKNIK NEGERI LHOKEUMAWA  
TAHUN 2022**

# BAB 1

## KARYAWAN

### A. Rancangan

- GUI



- DATABASE

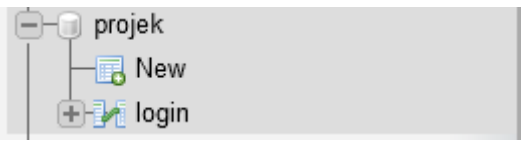


Table	Action	Rows	Type	Collation	Size	Overhead
login	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_general_ci	16.0 KiB	-
1 table	Sum	1	InnoDB	utf8mb4_general_ci	16.0 KiB	0 B

### B. Fitur

- Selesai

#### 1. Menampilkan proses login

```
6 package koneksi;
7 import java.sql.Connection;
8 import java.sql.ResultSet;
9 import java.sql.Statement;
10 import javax.swing.JOptionPane;
11 /**
12  *
13  * @author hp
14  */
```

```

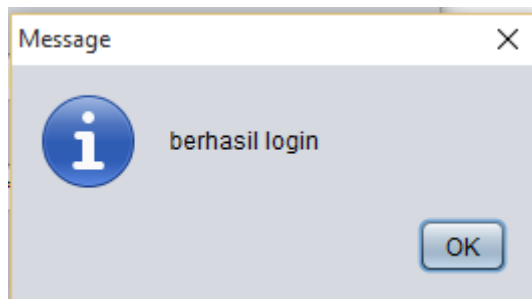
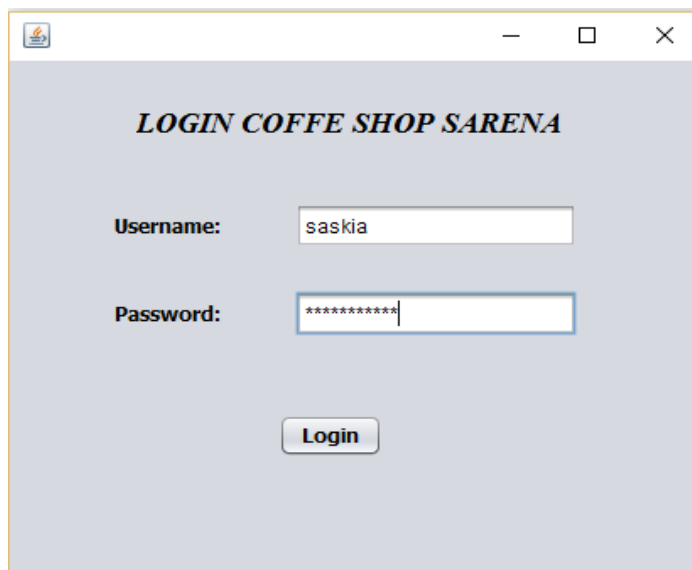
15 public class login extends javax.swing.JFrame {
16     Connection con;
17     Statement stat;
18     ResultSet rs;
19     String sql;
20
21     /**
22      * Creates new form login
23      */
24     public login() {
25         initComponents();
26         coneK DB = new coneK();
27         DB.config();
28         con = DB.con;
29         stat = DB.stm;
30     }
31
32     /**
33      * This method is called from within the constructor to initialize the form.
34      * WARNING: Do NOT modify this code. The content of this method is always
35      * regenerated by the Form Editor.
36      */
37     @SuppressWarnings("unchecked")
38     Generated Code
110
111     private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
112         // TODO add your handling code here:
113
114         try {
115             sql = "SELECT * FROM login WHERE username='"+txt_name.getText()+"' AND password='"+txt_pass.getText()+"'";
116             rs = stat.executeQuery(sql);
117             if(rs.next()){
118                 if(txt_name.getText().equals(rs.getString("username")) && txt_pass.getText().equals(rs.getString("password"))){
119                     JOptionPane.showMessageDialog(null, "berhasil login");
120                 }
121             }
122             else{
123                 JOptionPane.showMessageDialog(null, "username atau password salah");
124             }
125         } catch (Exception e) {
126             JOptionPane.showMessageDialog(this, e.getMessage());
127         }
128
129     /**
130      * @param args the command line arguments
131      */
132     public static void main(String args[]) {
133         /* Set the Nimbus look and feel */
134         Look and feel setting code (optional)
135
136         /* Create and display the form */
137         java.awt.EventQueue.invokeLater(new Runnable() {
138             public void run() {
139                 new login().setVisible(true);
140             }
141         });
142     }
143
144     // Variables declaration - do not modify
145     private javax.swing.JButton jButton1;
146     private javax.swing.JLabel jLabel1;
147     private javax.swing.JLabel jLabel2;
148     private javax.swing.JLabel jLabel3;
149     private javax.swing.JTextField txt_name;
150     private javax.swing.JPasswordField txt_pass;
151     // End of variables declaration
152 }

```

## 2. Menampilkan koneksi login

```
6 package koneksi;
7 import java.sql.Connection;
8 import java.sql.DriverManager;
9 import java.sql.Statement;
10 import javax.swing.JOptionPane;
11 /**
12  *
13  * @author hp
14  */
15 public class konek {
16     Connection con;
17     Statement stm;
18
19     public void config(){
20         try {
21             Class.forName("com.mysql.jdbc.Driver");
22             con=DriverManager.getConnection("jdbc:mysql://localhost/projek", "root", "");
23             stm = con.createStatement();
24         } catch (Exception e) {
25             JOptionPane.showMessageDialog(null, "koneksi gagal "+e.getMessage());
26         }
27     }
28 }
```

- **Belum Selesai**
  1. Membuat seluruh data karyawan
- **Hasil Sementara Projek**



Masih login saja yang bisa ditampilkan

## C. Lampiran

Dikumpulkan pada tanggal 14 April 2022 sekitaran jam 11.00

main

[Kelompok1\\_CoffeShopSarena](#) / [Login\\_CoffeShopSarena](#) /

Go to file

Add file

SaskiaFebrianti pertama

a1ceaf3 2 hours ago 

History

..

build/classes

pertama

2 hours ago

nbproject

pertama

2 hours ago

src

pertama

2 hours ago

build.xml

pertama

2 hours ago

manifest.mf

pertama

2 hours ago

## BAB 2

### STOK BARANG

#### A. Rancangan

- **GUI**

Untuk GUI belum selesai karena dalam proses pembuatan

- **DATABASE**

Columns: <span>+</span> Add <span>-</span> Remove <span>▲</span> Up <span>▼</span> Down											
#	Name	Datatype	Length/Set	Unsign...	Allow N...	Zerofill	Default	Comment	Collation	Expression	Virtuality
1	Kode Barang	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No default				
2	Nama Barang	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL		latin1_swedish_ci		
3	Harga Beli	INT	11	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL				
4	harga Jual	INT	11	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL				
5	Tanggal Masuk	DATETIME		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL				

- **KONEKSI DATABASE**

```
package stokbarang;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.Statement;

public class DBKoneksi {
    public static Connection con;
    public static Statement stm;
    public static void main(String args[]){
        try {
            String url ="jdbc:mysql://localhost/stokbarang";
            String user="root";
            String pass="";
            Class.forName("com.mysql.jdbc.Driver");
            con =DriverManager.getConnection(url,user,pass);
            stm = con.createStatement();
            System.out.println("koneksi berhasil;");
        } catch (Exception e) {
            System.err.println("koneksi gagal" +e.getMessage());
        }
    }
}

Object DBConnection;
Object Connection;
class connection {

    public connection() {

    }

}
```

#### Output - Stokbarang (run)

```
run:
koneksi berhasil;
BUILD SUCCESSFUL (total time: 2 seconds)
```

## B. Fitur

- **Selesai**

1. Menampilkan Data barang

```
private void select(String sql){  
  
    String header[]={"Kode Barang","Nama Barang","Harga Beli","Harga Jual","Tanggal Masuk"};  
    DefaultTableModel defaultTableModel=new DefaultTableModel(null, header);  
    jTable1.setModel(defaultTableModel);  
  
    for (int i = 0; i < jTable1.getRowCount(); i++) {  
        defaultTableModel.removeRow(i);  
    }  
  
    try {  
        resultSet=statement.executeQuery(sql);  
  
        while (resultSet.next()) {  
            String id_barang=resultSet.getString(1);  
            String kategori=resultSet.getString(2);  
            String nama_barang=resultSet.getString(3);  
            int stok=resultSet.getInt(4);  
  
            Object baris[]={id_barang,kategori,nama_barang,stok};  
            defaultTableModel.addRow(baris);  
        }  
    } catch (Exception e) {  
        JOptionPane.showMessageDialog(null, e.getMessage());  
    }  
}
```

- **Belum Selesai**

1. Menampilkan stok barang
2. Membuat method untuk dipanggil
3. Pembuatan GUI

- Hasil Sementara Proyek

```

public stokbarang () {
    initComponents();
    setTitle("STOK BARANG");
    koneksi = new DBKoneksi();
}

private void select(String sql){

    String header[]={"Kode Barang","Nama Barang","Harga Beli","Harga Jual","Tanggal Masuk"};
    DefaultTableModel defaultTableModel=new DefaultTableModel(null, header);
    jTable1.setModel(defaultTableModel);

    for (int i = 0; i < jTable1.getRowCount(); i++) {
        defaultTableModel.removeRow(i);
    }

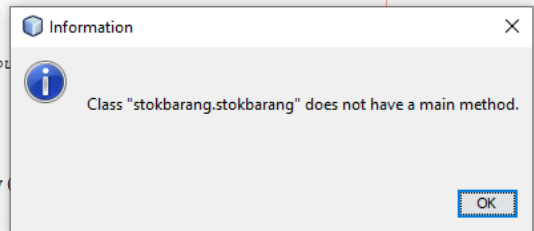
    try {
        resultSet=statement.executeQuery(sql);

        while (resultSet.next()) {
            String id_barang=resultSet.getString(1);
            String kategori=resultSet.getString(2);
            String nama_barang=resultSet.getString(3);
            int stok=resultSet.getInt(4);

            Object baris[]={id_barang, kategori, nama_barang, stok};
            defaultTableModel.addRow(baris);
        }
    } catch (Exception e) {
        JOptionPane.showMessageDialog(null, e.getMessage());
    }
}

private void initComponents() {
    throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods,
}

```



Belum bisa menampilkan stokbarang dikarenakan pembuatan method belum ada

### C. Lampiran

Dikumpulkan pertama kali pada tanggal 14 April 2022 pada sekitaran jam 10:45

utama ▾ Kelompok1_CoffeeShopSarena / Stok Barang /		Buka file	Tambah berkas ▾	...
Alkhaidar Koneksi Database		900b507	1 jam yang lalu	Sejarah
..				
membangun	Koneksi Basis Data			1 jam yang lalu
proyek nb	Koneksi Basis Data			1 jam yang lalu
src/ stok barang	Koneksi Basis Data			1 jam yang lalu
applet.policy	Koneksi Basis Data			1 jam yang lalu
build.xml	Koneksi Basis Data			1 jam yang lalu
manifest.mf	Koneksi Basis Data			1 jam yang lalu



## BAB 3

### Penjualan / Kasir

#### A. Rancangan

- **GUI**

The screenshot shows a web application interface titled "PENJUALAN". It includes input fields for "No Transaksi", "Tanggal", "ID Costomer", "Nama Customer", "ID Barang", "Nama Barang", "Harga", and "Jumlah". Below these is a table with four columns: "Title 1", "Title 2", "Title 3", and "Title 4". To the right of the table are "Tambah" and "Hapus" buttons. At the bottom, there are fields for "Total Bayar", "Bayar", and "Kembalian", along with a "Simpan" button and a large teal button.

- **DATABASE**

The screenshot shows a database management tool interface. On the left, a tree view displays the database structure: "dbpenjualan" (containing "Baru", "pembayaran", "penjualan", and "dbperpustakaan") and "dbperpustakaan". The main area shows a table list for "dbpenjualan" with the following data:

Tabel	Tindakan	Baris	Jenis	Penyortiran	Ukuran	Beban
<input type="checkbox"/> pembayaran	[Icons]	0	InnoDB	latin1_swedish_ci	16.0 KB	-
<input type="checkbox"/> penjualan	[Icons]	0	InnoDB	latin1_swedish_ci	16.0 KB	-
2 tabel	Jumlah	0	InnoDB	latin1_swedish_ci	32.0 KB	0 B

#### B. Fitur

- **Selesai**

1. Menampilkan menghitung total biaya

```

1  /*
2   * To change this license header, choose License Headers in Project Properties.
3   * To change this template file, choose Tools | Templates
4   * and open the template in the editor.
5   */
6   package kasir;
7
8   import java.sql.Connection;
9   import java.sql.PreparedStatement;
10  import java.sql.ResultSet;
11  import java.sql.Statement;
12  import java.sql.Date;
13  import java.text.SimpleDateFormat;
14  import javax.swing.table.DefaultTableModel;
15
16  /**
17   *
18   * @author user
19   */
20  public class penjualan extends javax.swing.JFrame {
21
22      private DefaultTableModel model;
23
24      public void totalBiaya() {
25          int jumlahBaris = jTable.getRowCount();
26          int totalBiaya = 0;
27          int jumlahBarang, hargaBarang;
28          for (int i = 0; i < jumlahBaris; i++) {
29              jumlahBarang = Integer.parseInt(jTable.getValueAt(i, 3).toString());
30              hargaBarang = Integer.parseInt(jTable.getValueAt(i, 4).toString());
31              totalBiaya = totalBiaya + (jumlahBarang * hargaBarang);
32          }
33          txTotalBayar.setText(String.valueOf(totalBiaya));
34          txTampil.setText("Rp" + totalBiaya + ",00");
35      }
36  }

```

## 2. Menampilkan membuat automember

```

37  private void automember() {
38      try {
39          Connection c = koneksi.getKoneksi();
40          Statement s = c.createStatement();
41          String sql = "SELECT * FROM dbpenjualan ORDER BY NoFaktur DESC";
42          ResultSet r = s.executeQuery(sql);
43          if (r.next()) {
44              String NoFaktur = r.getString("NoFaktur").substring(2);
45              String TR = "" + (Integer.parseInt(NoFaktur) + 1);
46              String No1 = "";
47              if (TR.length() == 1) {
48                  No1 = "000";
49              } else if (TR.length() == 2) {
50                  No1 = "00";
51              } else if (TR.length() == 3) {
52                  No1 = "0";
53              } else if (TR.length() == 4) {
54                  No1 = "";
55              }
56              txNoTransaksi.setText("TR" + No1 + TR);
57          } else {
58              txNoTransaksi.setText("TR0001");
59          }
60          r.close();
61          s.close();
62      } catch (Exception e) {
63          System.out.println("Automember Error");
64      }
65  }
66  }

```

### 3. Menampilkan LoadData dan Kosong

```
66
67 public void loadData() {
68     DefaultTableModel model = (DefaultTableModel) jTable.getModel();
69     model.addRow(new Object[]{
70         txNoTransaksi.getText(),
71         txIDBarang.getText(),
72         txNamaBarang.getText(),
73         txJumlah.getText(),
74         txHarga.getText(),
75         txTotalBayar.getText(),});
76 }
77
78 public void kosong() {
79     DefaultTableModel model = (DefaultTableModel) jTable.getModel();
80     while (model.getRowCount() > 0) {
81         model.removeRow(0);
82     }
83 }
```

### 4. Membuat Koneksi

```
6 package kasir;
7
8 /**
9  *
10  * @author user
11  */
12 import java.sql.Connection;
13 import java.sql.DriverManager;
14 import java.sql.SQLException;
15
16 public class koneksi {
17
18     private static Connection koneksi;
19
20     public static Connection getKoneksi() {
21         if (koneksi == null) {
22             try {
23                 String url = "jdbc:mysql://localhost:3306/dbpenjualan";
24                 String user = "root";
25                 String Password = "";
26                 DriverManager.registerDriver(new com.mysql.jdbc.Driver());
27                 koneksi = DriverManager.getConnection(url, user, Password);
28                 System.out.println("Berhasil");
29             } catch (Exception e) {
30                 System.out.println("Error");
31             }
32         }
33         return koneksi;
34     }
35
36     public static void main(String args[]) {
37         getKoneksi();
38     }
39 }
```

- **Belum Selesai**

1. Menampilkan data yang dibeli
2. Mengisi table NoFaktur
3. Membuat method tambah pesanan

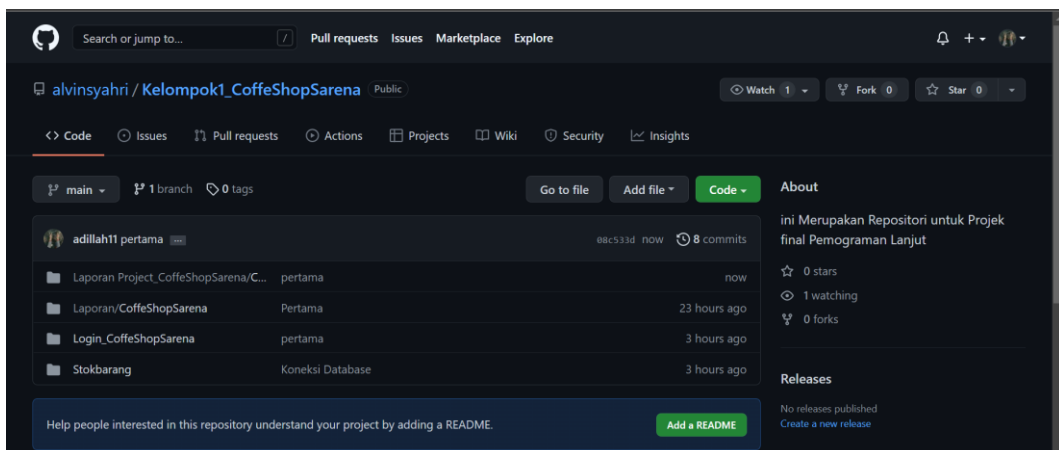
- **Hasil Sementara Projek**

[illegible]

Karena belum ada memasukkan barang, nama barang, harga barang, dan jumlah yang dipesan maka dari itu totalbyar, bayar, dan kembalian belum dapat ditampilkan.

### C. Lampiran

Dikumpulkan pertama kali pada tanggal 14 April 2022 pada sekitaran jam 14:50



## BAB 4

### LAPORAN

#### A. Rancangan

- **GUI**

Laporan Penjualan

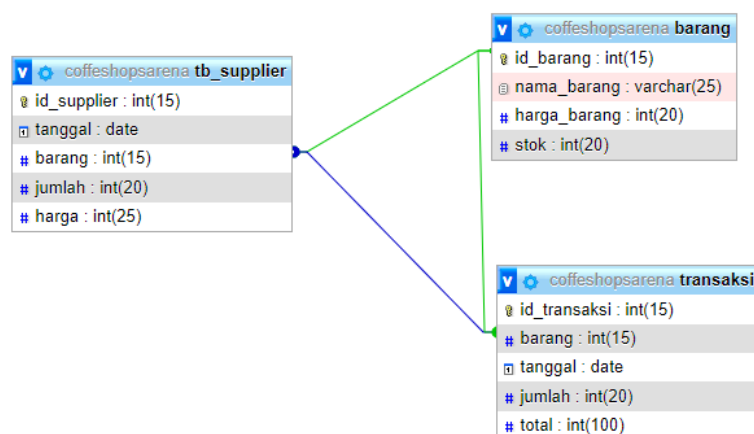
Bulan : 01 Tahun :

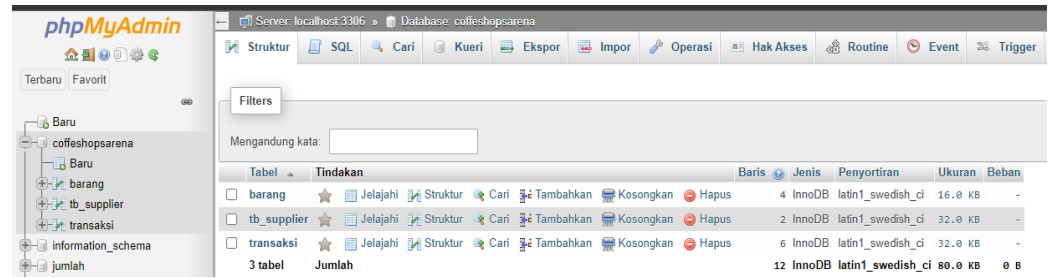
Title 1	Title 2	Title 3	Title 4	Title 5	Title 6

Jumlah Transaksi : 0

Total Belanjaan  
Total Transaksi  
Pendapatan

- **DATABASE**





## B. Fitur

- **Selesai**

1. Menampilkan record laporan transaksi sesuai dengan bulan dan tahun sekarang

```
public void TampilData() {
    DefaultTableModel tabel = new DefaultTableModel();
    tabel.addColumn("id_transaksi");
    tabel.addColumn("nama_barang");
    tabel.addColumn("tanggal");
    tabel.addColumn("harga");
    tabel.addColumn("jumlah");
    tabel.addColumn("total");
    jTable1.setModel(tabel);

    DateFormat FormatTahun = new SimpleDateFormat("yyyy");
    DateFormat FormatBulan = new SimpleDateFormat("MM");
    Calendar kalender = Calendar.getInstance();

    try {
        Connection c = Koneksi.getKoneksi();
        Statement s = c.createStatement();

        String sql = "SELECT t.id_transaksi, b.nama_barang, t.tanggal, b.harga, "
            + "t.jumlah, t.total from transaksi t LEFT JOIN barang b ON t.barang = "
            + "b.id_barang where year(t.tanggal) = '"
            + FormatTahun.format(kalender.getTime()) + "' & month(t.tanggal) = '" + FormatBulan.format(kalender.getTime()) + "'";
        ResultSet r = s.executeQuery(sql);

        while(r.next()) {
            tabel.addRow(new Object[] {
                r.getString("id_transaksi"),
                r.getString("nama_barang"),
                r.getString("tanggal"),
                r.getString("harga"),
                r.getString("jumlah"),
                r.getString("total")
            });
        }
        hitung(FormatTahun.format(kalender.getTime()), FormatBulan.format(kalender.getTime()));
    } catch (SQLException e) {
        System.out.println("terjadi kesalahan");
    }
}
```

2. Menampilkan record laporan transaksi sesuai dengan bulan dan tahun yang diinginkan

```
private void txtTahunActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    DefaultTableModel tabel = new DefaultTableModel();  
    tabel.addColumn("id_transaksi");  
    tabel.addColumn("nama_barang");  
    tabel.addColumn("tanggal");  
    tabel.addColumn("harga");  
    tabel.addColumn("Jumlah");  
    tabel.addColumn("total");  
    jTable1.setModel(tabel);  
  
    try{  
        Connection c = Koneksi.getConnection();  
        String sql = "Select t.id_transaksi, b.nama_barang, t.tanggal, "  
        + "b.harga, t.jumlah, t.total from transaksi t LEFT JOIN "  
        + "barang b ON t.barang = b.id_barang where year(t.tanggal) = '"  
        + txtTahun.getText() + "' && month(t.tanggal) = '" + cmbBulan.getSelectedItem() + "'";  
        Statement stat = c.createStatement();  
        ResultSet rs = stat.executeQuery(sql);  
  
        while(rs.next()){  
            tabel.addRow(new Object[]{  
                rs.getString("id_transaksi"),  
                rs.getString("nama_barang"),  
                rs.getString("tanggal"),  
                rs.getString("harga"),  
                rs.getString("jumlah"),  
                rs.getString("total")  
            });  
        }  
        String a = String.valueOf(cmbBulan.getSelectedItem());  
        hitung(txtTahun.getText(), a);  
    } catch (Exception e) {  
        System.out.println(e);  
    }  
}
```

3. Menampilkan jumlah record dari tampilan record laporan transaksi pada point 1 dan 2

```
public void hitung(String a, String b){  
    try{  
        Connection c = Koneksi.getConnection();  
        Statement s = c.createStatement();  
  
        String sql = "SELECT COUNT(id_transaksi) as I FROM transaksi where year(tanggal) = '"  
        + a + "' && month(tanggal) = '" + b + "'";  
        ResultSet r = s.executeQuery(sql);  
        if(r.next()){  
            txtJumlahTransaksi.setText(String.valueOf(r.getString("I")));  
        }  
    } catch (Exception e) {  
        System.out.println(e);  
    }  
}
```

- **Belum Selesai**

1. Pengoutputan data jumlah total belanjaan barang yang data ini diperoleh dari pejumlahan pembelian stok barang

2. Pengoutputan data jumlah total transaksi barang yang mana data ini diperoleh dari record pada tampilan record laporan yang ditampilkan sesuai dengan pilihannya baik sekarang maupun bulan dan tahun yang diinginkan

- **Hasil Sementara Projek**

id_transaksi	nama_barang	tanggal	harga	Jumlah	total
1	Capucino	2022-04-13	8000	1	8000
5	Sanger	2022-04-14	10000	2	20000

Jumlah Transaksi : 2

Total Belanjaan

Total Transaksi

Pendapatan

Kenapa hanya dua record itu dikarenakan untuk transaksi pada bulan dan tahun sekarang program berjalan hanya record ini lah yang termasuk begitu juga dengan jumlah transaksi dikarenakan Cuma 2 record yang tampil makna jumlah transaksi tersebut adalah 2

### C. Kendala

1. Untuk pencarian bulan dengan metode inputan angka tidak bisa dikarenakan harus ada dua fungsi yang sama pada bulan tahun  
Solusi:  
Mebuat button combo box untuk inputan bulan
2. Banyaknya pendeklarasian variable yang sama / objek yang sama dengan fungsi yang berbeda yang menyebabkan coding yang terlalu Panjang
3. Pembuatan database untuk menghitung jumlah belanjaan yang masih ambigu.



## D. Lampiran

Dikumpulkan pertama kali pada tanggal 13 April 2022 pada sekitaran jam 16:05

