**LAPORAN PRAKTIKUM**

**PERTEMUAN-12**

Diajukan untuk memenuhi salat satu tugas praktikum Mata kuliah Pemrograman Berorientasi Objek



**Disusun Oleh:**

**Daiva Raditya Pradipa (231511039)**

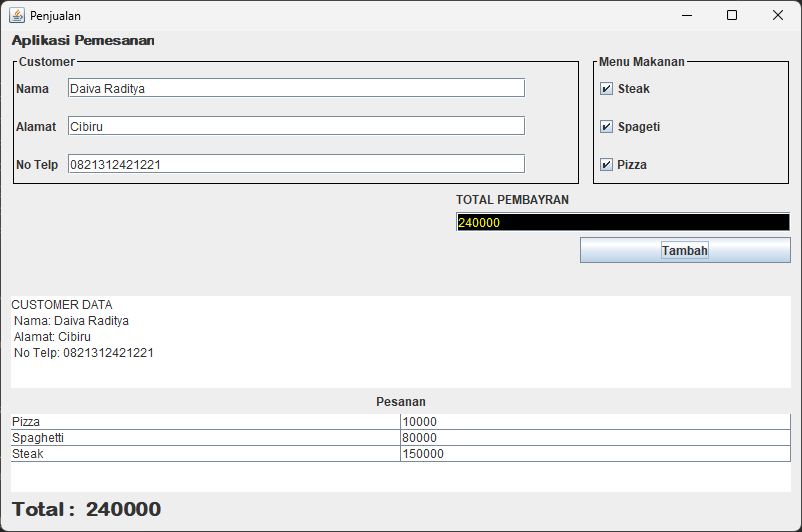
**Jurusan Teknik Komputer dan Informatika**

**Program Studi D-3 Teknik Informatika**

**Politeknik Negeri Bandung**

**2024**

1. **Aplikasi Pemesanan Menggunakan GUI**
2. Screenshot hasil program

****

1. Penjelasan

Program ini merupakan program aplikasi pemesanan makanan yang dibuat menggunakan GUI (Graphical User Interface) dengan menggunakan library java swing. Pada program ini user dapat memasukkan identitas customer berupa nama, Alamat, dan no telp yang dapat diinputkan pada textfield nama, alamat, dan no telp pada section customer. Selain itu user juga dapat memasukkan menu yang dipesan dengan menceklis 1 atau lebih makanan pada section menu makanan. Kemudian saat user menekan tombol tambah, maka program secara otomatis akan mengkalkulasi total harga yang harus dibayar user pada text field total pembayaran dan pada label total di pojok kiri bawah. Selain itu identias customer juga akan ditampilkan pada textarea dengan judul “CUSTOMER DATA”. Terakhir user juga bisa melihat menu menu yang dipesan beserta rincian harga masing-masing makanan pada tabel dibawah text area customer data.

Program ini dibuat menggunakan GUI form dengan library java swing yang melibatkan frame dan beberapa komponen ui berikut

1. JPanel

JPanel digunakan sebagai wadah atau windows panel untuk memungkinkan komponen komponen seperti jButton, Jlabel, JTextField, dan komponen lainnya ditampilkan pada program. Selain itu JPanel dapat digunakan pula untuk melakukan grouping atau pengelompokan beberapa komponen.

1. JTextField

JTextField digunakan sebagai komponen untuk menerima input dari user. Contoh pada program adalah textfield nama, alamat, dan no telp

1. JLabel

Digunakan untuk menampilkan text pada GUI. Contoh pada program adalah label text nama, alamat, dan no telp.

1. JTextArea

Digunakan untuk menerima input dari user namun dalam jumlah yang banyak dan memungkinkan user untuk melihat seluruh inputannya tanpa harus melakukan pergesaran view seperti pada JTextField. Contoh pada program adalah text area untuk menampilkan customer data. Disini saya melakukan konfigurasi JTextArea agar tidak bisa menerima input karena disini saya hanya menggunakan JTextArea untuk menampilkan data identitas yang diinputkan cutomer.

1. JTable

Digunakan untuk membuatkan komponen tabel yang akan digunakan untuk mevisualisasikan data dalam bentuk tabel. Contoh pada program adalah tabel rincian pesanan customer.

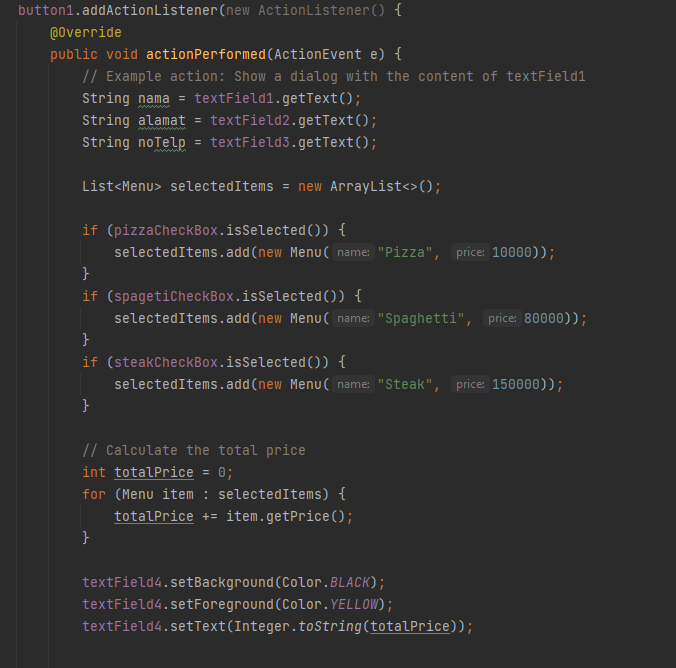
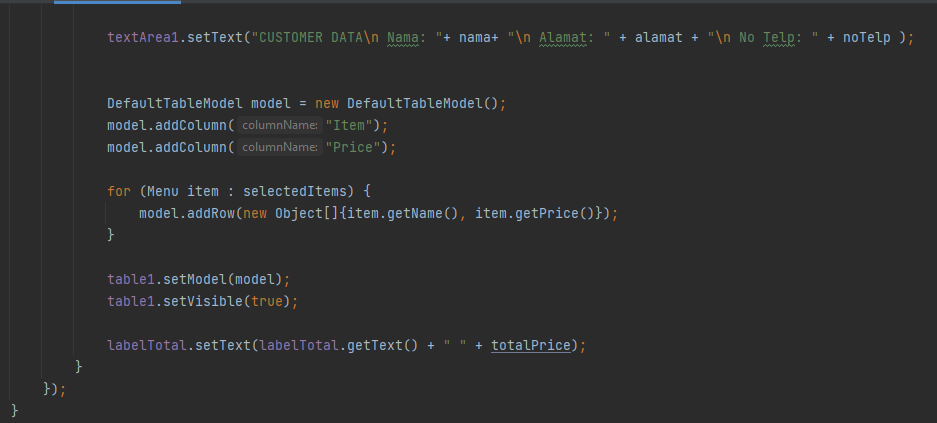
1. JButton

Digunakan untuk membuat komponen GUI Button sekaligus sebagai komponen melakukan action kalkulasi total harga.

1. JCheckbox

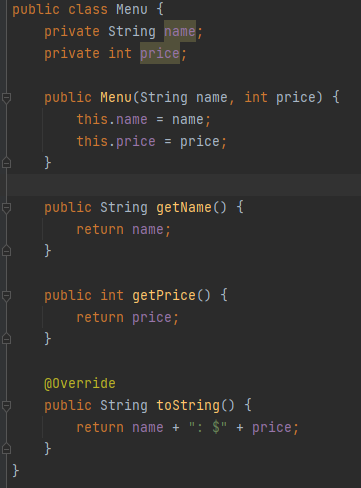
Digunakan untuk membuatkan komponen GUI checkbox yang memungkinkan pengguna untuk memilih atau membatalkan pilihan pada opsi tertentu. Contoh pada program adalah opsi menu yang dapat dipilih atau dipesan oleh customer.

Kemudian pada program ini selain GUI, saya juga menambahkan logika program sehingga program dapat menjalankan fungsinya sesuai tujuannya. Untuk menampung logika program disini saya memafaatkan actionEventListener yang diterapkan pada komponen JButton tambah. Untuk logika yang saya gunakan adalah sebagai berikut

Untuk logikanya sendiri cukup sederhana, dimana saya akan melakukan get value menggunakan method getText() pada masing masing objek text field nama, alamat, dan no telp yang diwakili dengan textField 1, 2, dan 3 yang kemudian disimpan ke variable string nama, alamat, dan noTelp. Method getText() disini begunakan untuk mendapatkan nilai yang diinputkan oleh pengguna pada text field.

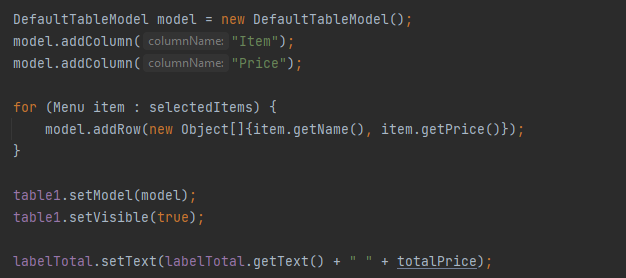
Kemudian saya akan melakukan proses get value pada checkbox untuk mengetahui menu mana saja yang customer pesan. Untuk melakukan ini saya menggunakan methos isSelected() pada masing masing objek checkbox dengan syntax if. Jika method isSelected ini mengembalikan nilai true, maka menu akan dibuat objek Menu dengan parameter nama dan harga yang akan dimasukkan pada collection list dengan tipe list Menu. Untuk class Menu sendiri dapat dilihat pada gambar berikut



Kemudian program akan mengkalkulasi total harga menu yang dipilih user dengan melooping List yang berisi menu yang dipesan customer dan menjumlahkan harga dari masing masing menu yang dipesan customer ke dalam variabel totalPrice. Kemudian program akan menampilkan data identitas cutomer pada textarea menggunakan method setText() pada objek text area dengan berisikan data nama, alamat, dan no telp yang tersimpan pada variabel nama, alamat, dan noTelp yang telah kita assign sebelumnya.

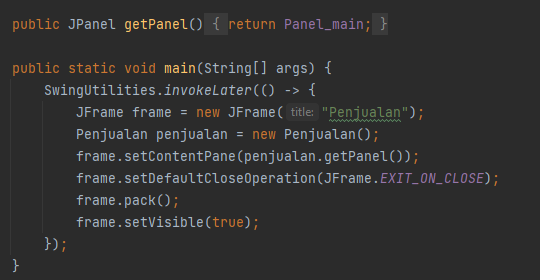
Pada logika terakhir adalah menampilkan rincian pesanan cutomer yang tersimpan pada collection List yang telah saya assign sebelumnya. Untuk melakukan proses assign data pada tabel komoponen Jtabel, kita perlu membuat terlebih dahulu objek DefaultTableModel dengan mengimport library menggunakan syntax “import javax.swing.table.DefaultTableModel;

” sehingga kita dapat melakukan configurasi jumlah kolom tabel dan row pada tabel.



Seperti yang dapat kita lihat pada kode diatas, kita dapat menambah kolom atau menset jumlah kolom dengan menggunakan method addColumn() dengan parameter nama kolom pada objek DefaultTableModel yang dalam kode atas di assign pada variabel model. Kemudian untuk menambah data pada tabel (Row) dengan menggunakan method addRow() denan parameter objek yang dimana objek disini berisi key nama makanan dan harga makanan pada List pesanan makanan yang diambil menggunakan method getName() dan getPrice() pada class Menu melalui looping list.

Terakhir untuk menampilkan GUI yang telah kita buat kita dapat menggunakan dua cara. Cara pertama adalah dengan menggunakan fitur preview yang ada pada form swing yang kita gunakan untuk melakukan design ui dan menambah komponen dan frame dengan drag and drop. Untuk fitur preview sendiri dapat diakses dengan melakukan klik kanan pada mouse di workspace GUI form. Namun, fiur ini hanya menjalankan GUI saja tanpa logika program yang telah kita buat pada class. Sedangkan cara kedua adalah dengan membuat main function pada class sebagai berikut.



SwingUtilities.invokeLater(() -> { ... }); digunakan untuk memastikan bahwa kode yang mengatur GUI pengguna berjalan di thread khusus yang aman untuk Swing, sehingga menghindari masalah yang mungkin timbul dari pemrosesan GUI secara langsung di thread utama.

Pada baris berikutnya, JFrame frame = new JFrame("Penjualan"); membuat jendela utama aplikasi dengan judul "Penjualan". Kemudian, Penjualan penjualan = new Penjualan(); membuat objek Penjualan yang dimana Penjualan merupakan class yang berisi komponen komponen serta frame java swing yang telah kita design atau tambahkan pada GUI form.

Setelah objek Penjualan dibuat, frame.setContentPane(penjualan.getPanel()); digunakan untuk menempatkan panel utama dari objek Penjualan (Panel\_main) sebagai isi dari JFrame, sehingga seluruh elemen antarmuka yang ada di dalam Panel\_main akan tampil di jendela tersebut. Selanjutnya, frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE) menentukan agar program otomatis berhenti saat jendela utama ini ditutup oleh pengguna.

Kemudian frame.pack() berfungsi untuk menyesuaikan ukuran jendela secara otomatis agar sesuai dengan ukuran komponen yang ada di dalamnya, sehingga tampilan lebih rapih tanpa perlu mengatur ukuran secara manual. Terakhir, frame.setVisible(true) digunakan untuk menampilkan jendela JFrame di layar sehingga aplikasi terlihat oleh pengguna.

1. Masalah
2. Solusi
3. Teman Yang Membantu