

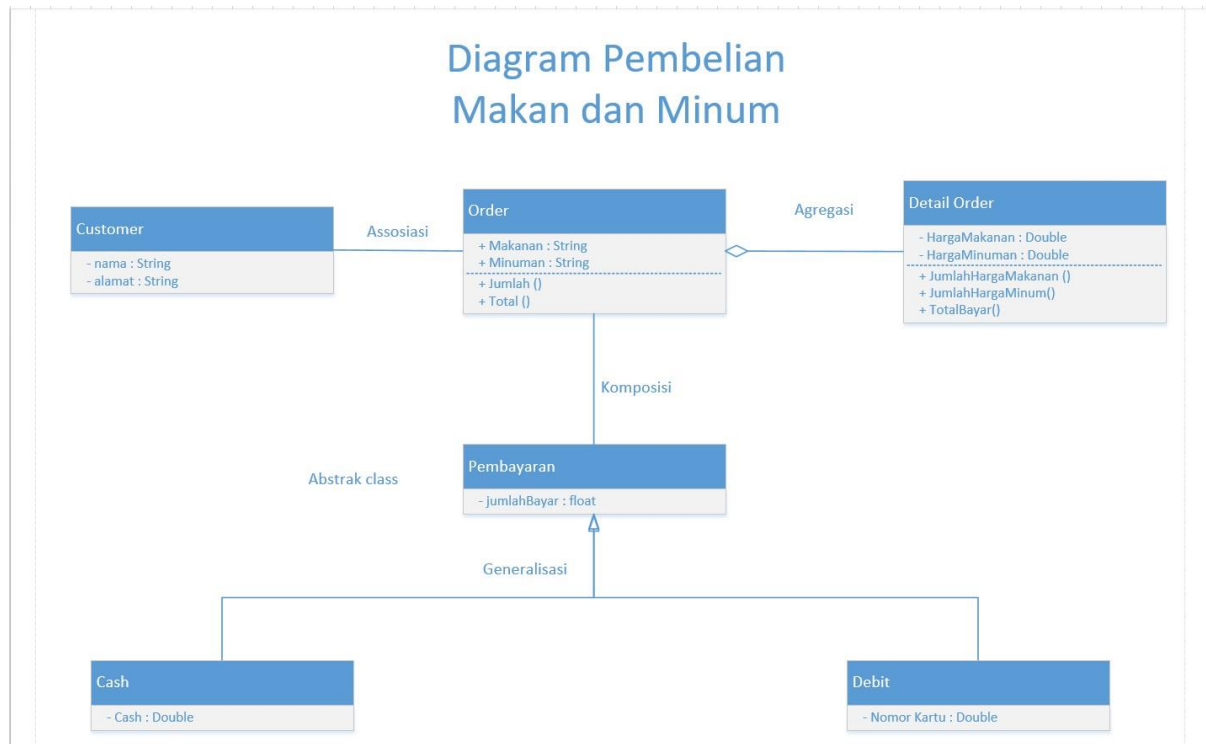
NAMA : IMAM SOLEHUDIN

NIM : 312110290

KELAS : TI.21.C.1

MATA KULIAH : PEMROGRAMAN ORIENTASI OBJEK

1. DIAGRAM CLASS

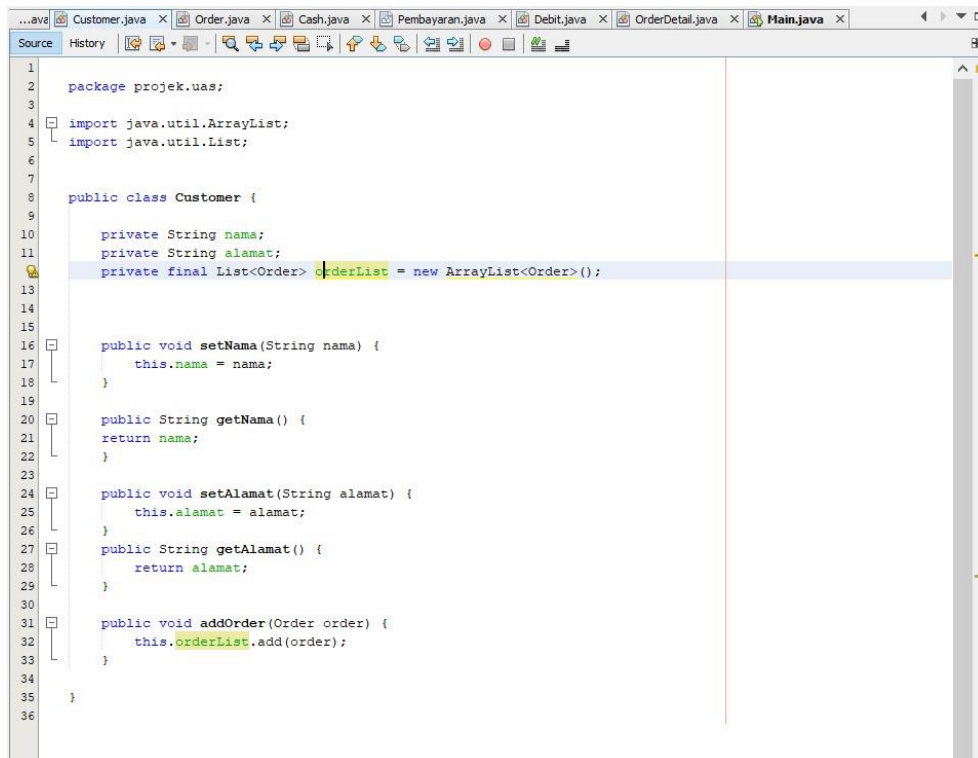


2. Kode Program Java dari Diagram diatas

Membuat program java dari Diagram Class diatas.

1. Class Customer

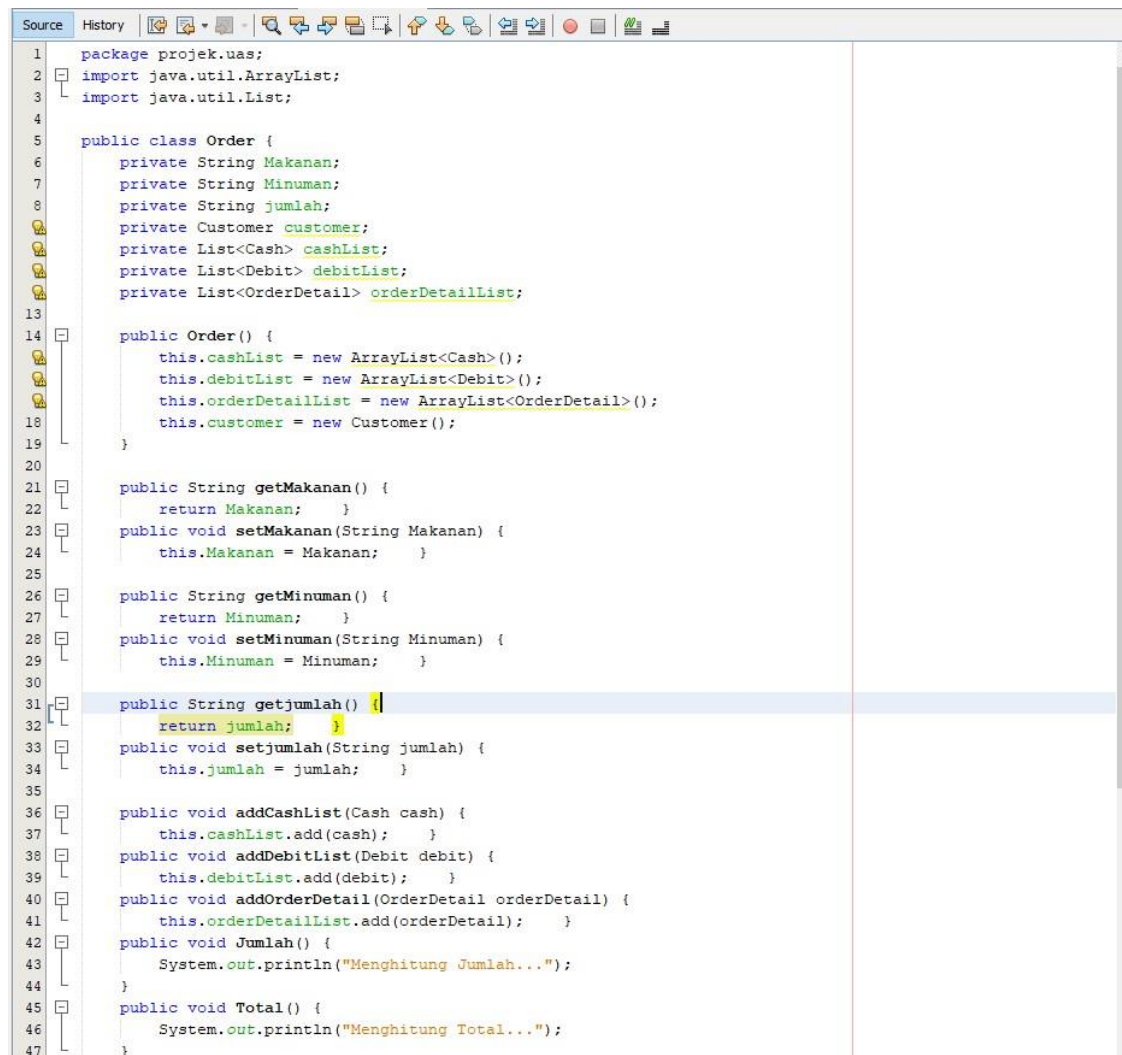
Class customer ini saling berhubungan assosiasi dengan class order, relasi class ini dinamakan assosiasi.



```
1 package projek.uas;
2
3
4 import java.util.ArrayList;
5 import java.util.List;
6
7
8 public class Customer {
9
10     private String nama;
11     private String alamat;
12     private final List<Order> orderList = new ArrayList<Order>();
13
14
15
16     public void setName(String nama) {
17         this.nama = nama;
18     }
19
20     public String getName() {
21         return nama;
22     }
23
24     public void setAddress(String alamat) {
25         this.alamat = alamat;
26     }
27     public String getAddress() {
28         return alamat;
29     }
30
31     public void addOrder(Order order) {
32         this.orderList.add(order);
33     }
34
35 }
36
```

2. Class Order

Class Order ini merupakan relasi class dari class customer mereka saling berhubungan, hanya saja class order tidak akan ada jika class customer tidak ada.



```
1 package projek.uas;
2 import java.util.ArrayList;
3 import java.util.List;
4
5 public class Order {
6     private String Makanan;
7     private String Minuman;
8     private String jumlah;
9     private Customer customer;
10    private List<Cash> cashList;
11    private List<Debit> debitList;
12    private List<OrderDetail> orderDetailList;
13
14    public Order() {
15        this.cashList = new ArrayList<Cash>();
16        this.debitList = new ArrayList<Debit>();
17        this.orderDetailList = new ArrayList<OrderDetail>();
18        this.customer = new Customer();
19    }
20
21    public String getMakanan() {
22        return Makanan;
23    }
24    public void setMakanan(String Makanan) {
25        this.Makanan = Makanan;
26    }
27
28    public String getMinuman() {
29        return Minuman;
30    }
31    public void setMinuman(String Minuman) {
32        this.Minuman = Minuman;
33    }
34
35    public String getjumlah() {
36        return jumlah;
37    }
38    public void setjumlah(String jumlah) {
39        this.jumlah = jumlah;
40    }
41
42    public void addCashList(Cash cash) {
43        this.cashList.add(cash);
44    }
45    public void addDebitList(Debit debit) {
46        this.debitList.add(debit);
47    }
48    public void addOrderDetail(OrderDetail orderDetail) {
49        this.orderDetailList.add(orderDetail);
50    }
51    public void Jumlah() {
52        System.out.println("Menghitung Jumlah...");
53    }
54    public void Total() {
55        System.out.println("Menghitung Total...");
56    }
57 }
```

3. Class OrderDetail

Class OrderDetail merupakan relasi class dari class order, relasi ini dinamakan agregasi. agregasi serupa dengan asosiasi, yaitu memiliki sebuah atribut dengan tipe dari class lain. Namun hubungan diperkuat dengan adanya dependensi pada konstruktornya.

```
1
2 package projek.uas;
3
4
5 public class OrderDetail {
6     private int HargaMakanan;
7     private int HargaMinuman;
8     private final Order order;
9
10
11     public OrderDetail(Order order) {
12         this.order = order;
13     }
14
15
16     public int getHargaMakanan() {
17         return HargaMakanan;
18     }
19
20     public void setHargaMakanan(int HargaMakanan) {
21         this.HargaMakanan = HargaMakanan;
22     }
23
24
25     public int getHargaMinuman() {
26         return HargaMinuman;
27     }
28
29     public void setHargaMinuman(int HargaMinuman) {
30         this.HargaMinuman = HargaMinuman;
31     }
32
33
34     public void JumlahHargaMakanan() {
35         System.out.println("Menghitung Jumlah Harga Makanan...");
36     }
37
38     public void JumlahHargaMinuman() {
39         System.out.println("Menghitung Jumlah Harga Minuman...");
40     }
41
42
43 }
```

4. Class Pembayaran

Class Pembayaran ini merupakan abstract class, karena class tersebut terdapat method yg tidak memiliki implementasi (kode program), hanya judul method dan parameternya

```
1
2 package projek.uas;
3
4
5 public abstract class Pembayaran {
6     private float jumlahBayar;
7     private final Order order;
8
9     public Pembayaran() {
10         order = new Order();
11     }
12 }
```

5. Class Cash

Class Cash ini merupakan class turunan dari class pembayaran.

```
1
2 package projek.uas;
3
4
5 public class Cash extends Pembayaran {
6     private float cash;
7
8     public float getCash() {
9         return cash;
10    }
11
12    public void setCash(float cash) {
13        this.cash = cash;
14    }
15 }
```

6. Class Debit

Class Debit ini merupakan class turunan dari class pembayaran

```
1
2 package projek.uas;
3
4
5 public class Debit extends Pembayaran {
6     private String NomorKartu;
7
8
9     public void prosesBayar() {
10         System.out.println("Proses Pembayaran");
11     }
12
13     public String getNomorKartu() {
14         return NomorKartu;
15     }
16
17     public void setNomorKartu(String NomorKartu) {
18         this.NomorKartu = NomorKartu;
19     }
20
21
22 }
```

7. Class Main

Class main ini digunakan untuk menjalankan program yang telah kita buat sebelumnya.

```

1 package projek.uas;
2
3
4 public class Main {
5
6     public static void main(String[] args) {
7
8         Customer imam = new Customer();
9         imam.setName("Imam Solehudin");
10        imam.setAlamat("Perum BCA Blok. E5 No. 30");
11        System.out.println("Pesanan : " + imam.getName());
12        System.out.println("Alamat Pesanan : " + imam.getAlamat());
13
14        System.out.println("=====");
15
16        Order jnt = new Order();
17        imam.addOrder(jnt);
18
19        jnt.setMakanan("Nasi Goreng");
20        jnt.setMinuman("Es Teh Manis");
21        jnt.setjumlah("1 Paket");
22        System.out.println("Pesanan : " + jnt.getMakanan()+"+" + jnt.getMinuman());
23        jnt.Jumlah();
24        jnt.Total();
25        System.out.println("Total pesanan : " + jnt.getjumlah());
26
27        System.out.println("=====");
28
29        Cash cash = new Cash();
30        cash.setCash((float) 20000.0);
31        jnt.addCashList(cash);
32        System.out.println("Total Pembayaran : Rp" + (int) cash.getCash());
33
34        System.out.println("=====");
35
36        Debit debit = new Debit();
37        debit.prosesBayar();
38        debit.setNomorKartu("4567891230123658");
39        jnt.addDebitList(debit);
40        System.out.println("Nomor kartu debit : " + debit.getNomorKartu());
41
42        System.out.println("Terima kasih telah berbelanja");
43
44    }
45 }

```


8. Hasil Program

```
18         imam.addOrder(jnt);
19
20         jnt.setMakanan("Nasi Goreng");
21         jnt.setMinuman("Es Teh Manis");
22         jnt.setjumlah("1 Paket");
```

Output - projek uas (run) X

```
run:
Pesanan : Imam Solehudin
Alamat Pesanan : Perum BCA Blok. E5 No. 30
=====
Pesanan : Nasi Goreng+Es Teh Manis
Menghitung Jumlah...
Menghitung Total...
Total pesanan : 1 Paket
=====
Total Pembayaran : Rp20000
=====
Proses Pembayaran
Nomor kartu debit : 4567891230123658
Terima kasih telah berbelanja
BUILD SUCCESSFUL (total time: 0 seconds)
```

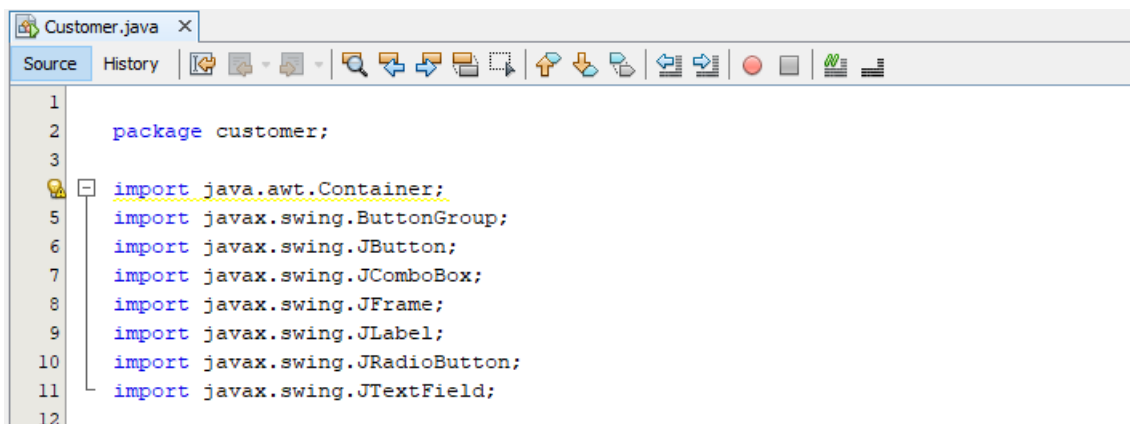
3. JAVA GUI

Java GUI (Graphic User Interface). Graphic User Interface (GUI) adalah pemrograman dengan bahasa Java yang dibuat menggunakan aplikasi yang berbasis GUI. Tujuannya adalah menambahkan beberapa komponen yang tidak bisa dibuat dalam basis text.

Disini saya akan membuat program java GUI dari diagram cals yang sudah kita buat diatas.

- Import Java

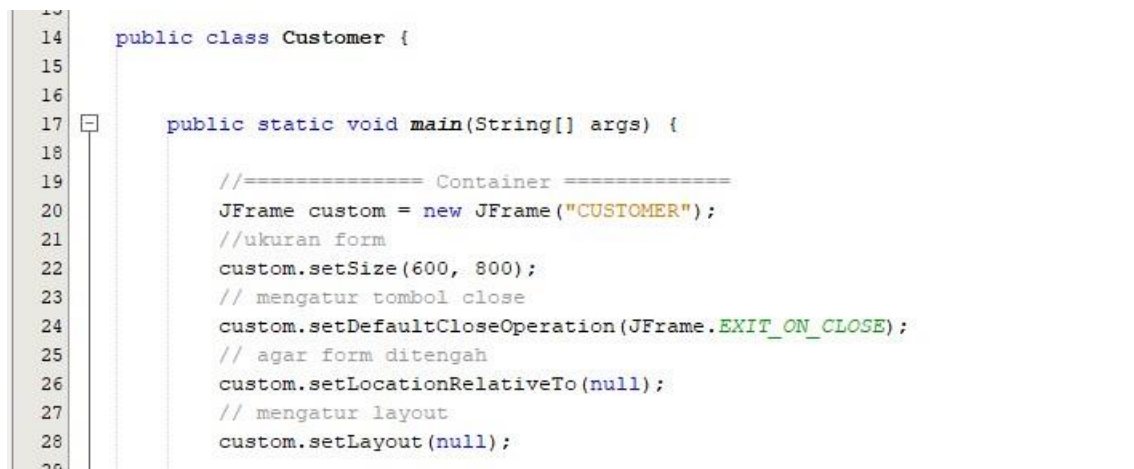
Import Pada Java Merupakan Suatu Perintah Untuk Memasukan suatu Method atau perintah dalam Bahasa Pemrograman Java sehingga perintah tersebut dapat Aktif dan digunakan atau berfungsi.



```
1
2 package customer;
3
4 import java.awt.Container;
5 import javax.swing.ButtonGroup;
6 import javax.swing.JButton;
7 import javax.swing.JComboBox;
8 import javax.swing.JFrame;
9 import javax.swing.JLabel;
10 import javax.swing.JRadioButton;
11 import javax.swing.JTextField;
12
```

- Container

Membuat class dan membuat objek menggunakan JFrame, JFrame merupakan turunan dari kelas java.awt.Frame dengan penambahan dukungan sebagai wadah atau tempat untuk komponen swing yaitu obyek dari kelas-kelas yang terletak di paket javax.



```
14 public class Customer {
15
16
17     public static void main(String[] args) {
18
19         //===== Container =====
20         JFrame custom = new JFrame("CUSTOMER");
21         //ukuran form
22         custom.setSize(600, 800);
23         // mengatur tombol close
24         custom.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
25         // agar form ditengah
26         custom.setLocationRelativeTo(null);
27         // mengatur layout
28         custom.setLayout(null);
29     }
```

- Membuat Komponen

Pada bagian ini kita membuat komponen objek yang akan kita tampilkan pada objek custom yang sudah kita buat sebelumnya.

```
29
30 // ===== Bagian Componen =====
31
32 JLabel labelcustom = new JLabel("CUSTOMER");
33 labelcustom.setBounds(180, 30, 100, 30);
34 custom.add(labelcustom);
35
36 JLabel labelNama = new JLabel("Nama");
37 labelNama.setBounds(20, 80, 150, 25);
38 custom.add(labelNama);
39
40 JTextField txtNama = new JTextField();
41 txtNama.setBounds(150, 80, 150, 25);
42 custom.add(txtNama);
43
44 JLabel labelAlamat = new JLabel("Alamat");
45 labelAlamat.setBounds(20, 110, 150, 25);
46 custom.add(labelAlamat);
47
48 JTextField txtAlamat = new JTextField();
49 txtAlamat.setBounds(150, 110, 150, 25);
50 custom.add(txtAlamat);
51
52
53 JLabel labelOrder = new JLabel("ORDER");
54 labelOrder.setBounds(180, 140, 150, 50);
55 custom.add(labelOrder);
56
57 JLabel labelPesanan = new JLabel("Pilih Pesanan");
58 labelPesanan.setBounds(40, 190, 150, 25);
59 custom.add(labelPesanan);
60
61 JLabel labelMakanan = new JLabel("Makanan");
62 labelMakanan.setBounds(40, 220, 150, 25);
63 custom.add(labelMakanan);
64
65 String Makan[] = {"Nasi Goreng", "Mie Goreng", "Ayam Goreng", "Ikan Goreng"};
66 JComboBox cbMakan = new JComboBox(Makan);
67 cbMakan.setBounds(150, 220, 100, 30);
68 custom.add(cbMakan);
69
```

```
Customer.java x
Source History
70 JLabel labelJPesan1 = new JLabel("Jumlah Pesanan");
71 labelJPesan1.setBounds(280, 190, 150, 25);
72 custom.add(labelJPesan1);
73
74 String Jumlah1[] = {"1", "2", "3", "4", "5"};
75 JComboBox cbJumlah1 = new JComboBox(Jumlah1);
76 cbJumlah1.setBounds(280, 220, 50, 30);
77 custom.add(cbJumlah1);
78
79
80 JLabel labelMinuman = new JLabel("Minuman");
81 labelMinuman.setBounds(40, 270, 150, 25);
82 custom.add(labelMinuman);
83
84 String Minum[] = {"Es Teh Manis", "Es Jeruk", "Cappucionno", "Air Putih"};
85 JComboBox cbMinum = new JComboBox(Minum);
86 cbMinum.setBounds(150, 270, 100, 30);
87 custom.add(cbMinum);
88
89 String Jumlah2[] = {"1", "2", "3", "4", "5"};
90 JComboBox cbJumlah2 = new JComboBox(Jumlah2);
91 cbJumlah2.setBounds(280, 270, 50, 30);
92 custom.add(cbJumlah2);
93
94 JLabel labelOrderDetail = new JLabel("ORDER DETAIL");
95 labelOrderDetail.setBounds(180, 320, 150, 50);
96 custom.add(labelOrderDetail);
97
98 JLabel labelHargaMakan = new JLabel("Harga Makanan");
99 labelHargaMakan.setBounds(20, 390, 150, 25);
100 custom.add(labelHargaMakan);
101
102 JTextField txtHargaMakan = new JTextField();
103 txtHargaMakan.setBounds(150, 390, 150, 25);
104 custom.add(txtHargaMakan);
105
106 JLabel labelHargaMinum = new JLabel("Harga Minum");
107 labelHargaMinum.setBounds(20, 420, 150, 25);
108 custom.add(labelHargaMinum);
109
110 JTextField txtHargaMinum = new JTextField();
111 txtHargaMinum.setBounds(150, 420, 150, 25);
112 custom.add(txtHargaMinum);
```

```
Customer.java x
Source History
112 custom.add(txtHargaMinum);
113
114 JTextField txtJuml = new JTextField();
115 txtJuml.setBounds(320, 390, 40, 25);
116 custom.add(txtJuml);
117
118 JLabel Jum1 = new JLabel("x");
119 Jum1.setBounds(310, 390, 40, 25);
120 custom.add(Jum1);
121
122 JTextField txtJum2 = new JTextField();
123 txtJum2.setBounds(320, 420, 40, 25);
124 custom.add(txtJum2);
125
126 JLabel Jum2 = new JLabel("x");
127 Jum2.setBounds(310, 420, 40, 25);
128 custom.add(Jum2);
129
130 JLabel labelJPesan2 = new JLabel("Jml");
131 labelJPesan2.setBounds(325, 350, 150, 50);
132 custom.add(labelJPesan2);
133
134
135 JLabel labelTotalBayar = new JLabel("Total Bayar");
136 labelTotalBayar.setBounds(20, 450, 150, 25);
137 custom.add(labelTotalBayar);
138
139 JTextField txtTotalBayar = new JTextField();
140 txtTotalBayar.setBounds(150, 450, 150, 25);
141 custom.add(txtTotalBayar);
142
143 JLabel labelPembayaran = new JLabel("Pembayaran");
144 labelPembayaran.setBounds(180, 480, 150, 50);
145 custom.add(labelPembayaran);
146
147 JRadioButton radioCash = new JRadioButton();
148 radioCash.setText("Cash");
149 radioCash.setBounds(130, 530, 80, 50);
150 JRadioButton radioDebit = new JRadioButton();
151 radioDebit.setText("Debit");
152 radioDebit.setBounds(250, 530, 80, 50);
153 custom.add(radioCash);
154 custom.add(radioDebit);
155
```



```
155
156     ButtonGroup grouppay = new ButtonGroup();
157     grouppay.add(radioCash);
158     grouppay.add(radioDebit);
159
160     JLabel labelDebit = new JLabel("Masukan Nomor Kartu");
161     labelDebit.setBounds(20, 590, 150, 25);
162     custom.add(labelDebit);
163
164     JTextField txtDebit = new JTextField();
165     txtDebit.setBounds(150, 590, 150, 25);
166     custom.add(txtDebit);
167     // menampilkan form
168
169
170     custom.setVisible(true);
171
172
173     JButton Submit = new JButton("SUBMIT");
174     Submit.setBounds(160, 650, 80, 25);
175     custom.add(Submit);
176
177     JButton Reset = new JButton("RESET");
178     Reset.setBounds(260, 650, 80, 25);
179     custom.add(Reset);
180
181 }
182
183
184 }
185
```

- ```
custom.add(labelAlamat);
```

**CUSTOMER**

Nama

Alamat

**ORDER**

| Pilih Pesanan                                       | Jumlah Pesanan                     |
|-----------------------------------------------------|------------------------------------|
| Makanan <input type="text" value="Nasi Goreng"/> ▼  | 1 <input type="text" value="1"/> ▼ |
| Minuman <input type="text" value="Es Teh Manis"/> ▼ | 1 <input type="text" value="1"/> ▼ |

**ORDER DETAIL**

|                                    | Jml                    |
|------------------------------------|------------------------|
| Harga Makanan <input type="text"/> | x <input type="text"/> |
| Harga Minum <input type="text"/>   | x <input type="text"/> |
| Total Bayar <input type="text"/>   |                        |

**Pembayaran**

☐ Cash
 ☐ Debit

Masukan Nomor Kartu