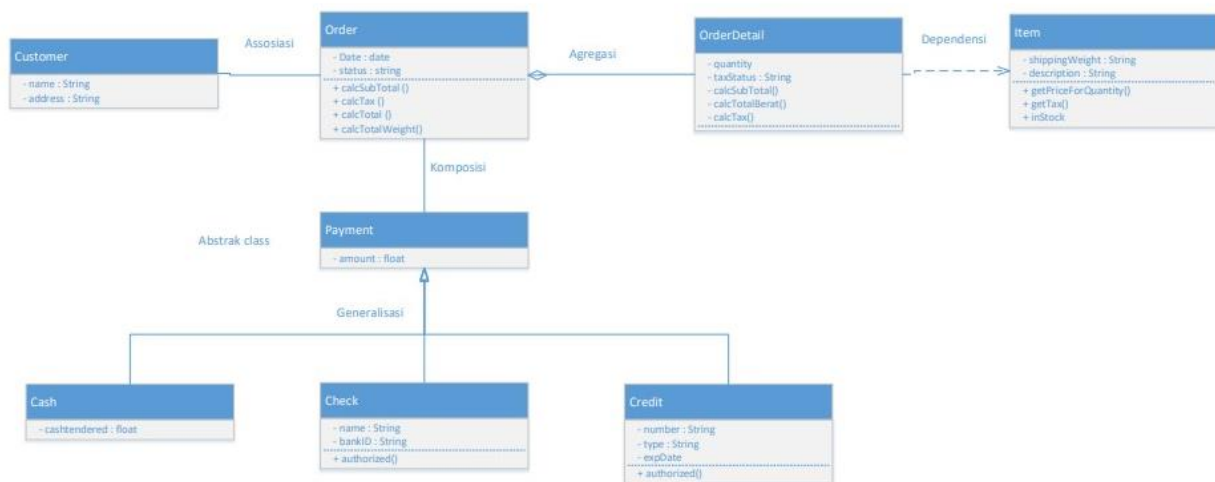


## DIAGRAM CLASS PEMESANAN



Membuat program java dari Diagram Class Pemesanan diatas.

### 1. Class Customer

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

```

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 customer;
7
8
9  import java.util.ArrayList;
10 import java.util.List;
11
12 public class Customer {
13     private String name;
14     private String address;
15     private final List<Order> orderList = new ArrayList<Order>();
16
17
18
19     public void setName(String name) {
20         this.name = name;
21     }
22
23     public String getName() {
24         return name;
25     }
26
27     public void setAddress(String address) {
28         this.address = address;
29     }
30     public String getAddress() {
31         return address;
32     }
33
34     public void addOrder(Order order) {
35         this.orderList.add(order);
36     }
37
38 }

```

## 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  /*
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 customer;
7
8  import java.time.LocalDate;
9  import java.util.ArrayList;
10 import java.util.List;
11
12
13 public class Order {
14     private LocalDate date;
15     private String status;
16     private Customer customer;
17     private List<Cash> cashList;
18     private List<Check> checkList;
19     private List<Credit> creditList;
20     private List<OrderDetail> orderDetailList;
21
22     public Order() {
23         this.cashList = new ArrayList<Cash>();
24         this.checkList = new ArrayList<Check>();
25         this.creditList = new ArrayList<Credit>();
26         this.orderDetailList = new ArrayList<OrderDetail>();
27         this.customer = new Customer();
28     }
29
30     public LocalDate getDate() {
31         return date;
32     }
33     public void setDate(LocalDate date) {
34         this.date = date;
35     }
36     public String getStatus() {
37         return status;
38     }
39     public void setStatus(String status) {
40         this.status = status;
41     }
42     public void addCashList(Cash cash) {
43         this.cashList.add(cash);
44     }
45     public void addCheckList(Check check) {
46         this.checkList.add(check);
47     }
48     public void addCreditList(Credit credit) {
49         this.creditList.add(credit);
50     }
51     public void addOrderDetail(OrderDetail orderDetail) {
52         this.orderDetailList.add(orderDetail);
53     }
54     public void calcSubTotal() {
55         System.out.println("Menghitung Sub Total...");
56     }
57     public void calcTax() {
58         System.out.println("Menghitung Tax...");
59     }
60     public void calcTotal() {
61         System.out.println("Menghitung Total...");
62     }
63     public void calcTotalWeight() {
64         System.out.println("Menghitung Total Berat...");
65     }
66
67 }
68
```

### 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   * 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 customer;
7
8  public class OrderDetail {
9      private int quantity;
10     private String taxStatus;
11     private Order order;
12     private Item item;
13
14     public OrderDetail(Order order) {
15         this.order = order;
16         this.item = new Item();
17     }
18
19     public int getQuantity() {
20         return quantity;
21     }
22
23     public void setQuantity(int quantity) {
24         this.quantity = quantity;
25     }
26
27     public String getTaxStatus() {
28         return taxStatus;
29     }
30
31     public void setTaxStatus(String taxStatus) {
32         this.taxStatus = taxStatus;
33     }
34
35     public void calcSubTotal() {
36         System.out.println("Menghitung Sub Total...");
37     }
38
39     public void calcWeight() {
40         System.out.println("Menghitung Berat...");
41     }
42
43     public void calcTax() {
44         System.out.println("Menghitung Tax...");
45     }
46 }
```

#### 4. Class Item

Class item merupakan relasi class dari OrderDetail, relasi ini dinamakan relasi dependency. Merupakan relasi antar kelas dimana satu kelas membutuhkan atau tergantung kepada kelas lainnya. Tapi ketergantungan tersebut tidak timbal balik.

```
4  * and open the template in the editor.
5  */
6  package customer;
7
8  import java.util.ArrayList;
9  import java.util.List;
10
11  public class Item {
12      private double shippingWeight;
13      private String description;
14      private List<OrderDetail> orderDetailList = new ArrayList<OrderDetail>();
15
16      public double getShippingWeight() {
17          return shippingWeight;
18      }
19
20      public void setShippingWeight(double shippingWeight) {
21          this.shippingWeight = shippingWeight;
22      }
23
24      public String getDescription() {
25          return description;
26      }
27
28      public void setDescription(String description) {
29          this.description = description;
30      }
31
32      public void addOrderDetail(OrderDetail orderDetail) {
33          this.orderDetailList.add(orderDetail);
34      }
35
36      public void getPriceForQuantity() {
37          System.out.println("Getting price for quantity...");
38      }
39
40      public void getTax() {
41          System.out.println("Getting Tax...");
42      }
43
44      public void inStock() {
45          System.out.println("In Stock!");
46      }
47  }
```

## 5. Class Payment

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

```
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 customer;
7
8
9  public class Cash extends Payment {
10     private float cashTendered;
11
12     public float getCashTendered() {
13         return cashTendered;
14     }
15
16     public void setCashTendered(float cashTendered) {
17         this.cashTendered = cashTendered;
18     }
19 }
```

## 6. Class Cash

Class Cash ini merupakan class turunan dari class payment.

```
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 customer;
7
8
9
10
11 public abstract class Payment {
12     private float amount;
13     private final Order order;
14
15     public Payment() {
16         order = new Order();
17     }
18 }
```

## 7. Class Check

Class Check ini merupakan class turunan dari class payment.

```
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 customer;
7
8  public class Check extends Payment {
9      private String name;
10     private String bankID;
11
12     public void authorized() {
13         System.out.println("Proses Pembayaran");
14     }
15
16     public String getName() {
17         return name;
18     }
19
20     public void setName(String name) {
21         this.name = name;
22     }
23
24     public String getBankID() {
25         return bankID;
26     }
27
28     public void setBankID(String bankID) {
29         this.bankID = bankID;
30     }
31 }
```

## 8. Class Credit

Class Credit ini merupakan class turunan dari class payment.

```
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 customer;
7
8
9  import java.time.LocalDate;
10
11  public class Credit extends Payment {
12      private String number;
13      private String type;
14      private LocalDate expDate;
15
16      public void authorized() {
17          System.out.println("Proses Pembayaran");
18      }
19
20      public String getNumber() {
21          return number;
22      }
23
24      public void setNumber(String number) {
25          this.number = number;
26      }
27
28      public String getType() {
29          return type;
30      }
31
32      public void setType(String type) {
33          this.type = type;
34      }
35
36      public LocalDate getExpDate() {
37          return expDate;
38      }
39
40      public void setExpDate(LocalDate expDate) {
41          this.expDate = expDate;
42      }
43  }
```



## 9. Class Main

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

```
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 customer;
7
8  import java.time.LocalDate;
9
10 public class Main {
11     public static void main(String[] args) {
12
13         Customer imam = new Customer();
14         imam.setName("Imam Solehudin");
15         imam.setAddress("Perum BCA Blok. E5 No. 30");
16         System.out.println("Pesanan : " + imam.getName());
17         System.out.println("Alamat Pesanan : " + imam.getAddress());
18
19         System.out.println("=====");
20
21         Order jnt = new Order();
22         imam.addOrder(jnt);
23         jnt.calcSubTotal();
24         jnt.calcTax();
25         jnt.calcTotal();
26         jnt.calcTotalWeight();
27         jnt.setDate(LocalDate.now());
28         jnt.setStatus("Kurir mengambil Pesanan");
29         System.out.println("Tanggal Pesanan : " + jnt.getDate());
30         System.out.println("Status pesanan : " + jnt.getStatus());
31
32         System.out.println("=====");
33
34         OrderDetail detailOrder = new OrderDetail(jnt);
35         detailOrder.calcSubTotal();
36         detailOrder.calcWeight();
37         detailOrder.calcTax();
38         detailOrder.setQuantity(5);
39         detailOrder.setTaxStatus("Pajak 2%");
40         jnt.addOrderDetail(detailOrder);
41         System.out.println("Jumlah Pesanan: " + detailOrder.getQuantity());
42         System.out.println("Status Pajak: " + detailOrder.getTaxStatus());
43
44         System.out.println("=====");
45
46         Item item = new Item();
47         item.getPriceForQuantity();
48         item.getTax();
49         item.inStock();
50         item.setShippingWeight(10.0);
51         item.setDescription("Barang Elektronik");
52         item.addOrderDetail(detailOrder);
53         System.out.println("Total berat pengiriman sebesar: " + item.getShippingWeight() + " Kg.");
54         System.out.println("Deskripsi barang: " + item.getDescription());
55
56         Cash cash = new Cash();
57         cash.setCashTendered((float) 10000.0);
58         jnt.addCashList(cash);
59         System.out.println("Pembayaran ongkos kirim : Rp" + (int) cash.getCashTendered());
60
61         System.out.println("=====");
62
63         Check check = new Check();
64         check.setName("Imam Solehudin");
65         check.setBankID("7896541230");
66         check.authorized();
67         jnt.addCheckList(check);
68         System.out.println("Pembayaran atas nama " + check.getName() + " dengan IDBank " + check.getBankID());
69
70         System.out.println("=====");
71
72         Credit credit = new Credit();
73         credit.authorized();
74         credit.setNumber("4567891230123658");
75         credit.setType("Debit");
76         credit.setExpDate(LocalDate.now());
77         jnt.addCreditList(credit);
78         System.out.println("Nomor kartu kredit " + credit.getNumber() + " dengan tipe " + credit.getType() + " akan berakhir pada " + credit.getExpDate());
79
80         System.out.println("Terima kasih telah berbelanja ditoko kami.");
81     }
82 }
83 }
```



## 10. Hasil Program

```
Output - Customer (run) X
run:
Pesanan : Imam Solehudin
Alamat Pesanan : Perum BCA Blok. E5 No. 30
=====
Menghitung Sub Total...
Menghitung Tax...
Menghitung Total...
Menghitung Total Berat...
Tanggal Pesanan : 2023-01-13
Status pesanan : Kurir mengambil Pesanan
=====
Menghitung Sub Total...
Menghitung Berat...
Menghitung Tax...
Jumlah Pesanan: 5
Status Pajak: Pajak 2%
=====
Getting price for quantity...
Getting Tax...
In Stock!
Total berat pengiriman sebesar: 10.0Kg.
Deskripsi barang: Barang Elektronik
Pembayaran ongkos kirim : Rp10000
=====
Proses Pembayaran
Pembayaran atas nama Imam Solehudin dengan IDBank 7896541230
=====
Proses Pembayaran
Nomor kartu kredit 4567891230123658 dengan tipe Debit akan berakhir pada 2023-01-13
Terima kasih telah berbelanja ditoko kami.
BUILD SUCCESSFUL (total time: 0 seconds)
```