

LAPORAN Pengerjaan Tugas Pertemuan 3

TEKNIK PEMOGRAMAN PRAKTIKUM

Disusun untuk memenuhi salah satu tugas pertemuan ketiga mata kuliah

Teknik Pemograman



Oleh:

Fadilah Akbar (231524041)

Kelas :

Teknik Informatika D4 – 1B

SARJANA TERAPAN PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA

POLITEKNIK NEGERI BANDUNG

STUDI KASUS 1

1. Amati desain setiap class, Apakah desain class tersebut sudah memenuhi konsep OOP yang benar? Jika tidak, coba anda perbaiki dengan mengacu pada Design Hint di Buku Chapter 4.10. Setiap perubahan yang dibuat harus dibubuhi penjelasan serta argumentasi yang jelas.

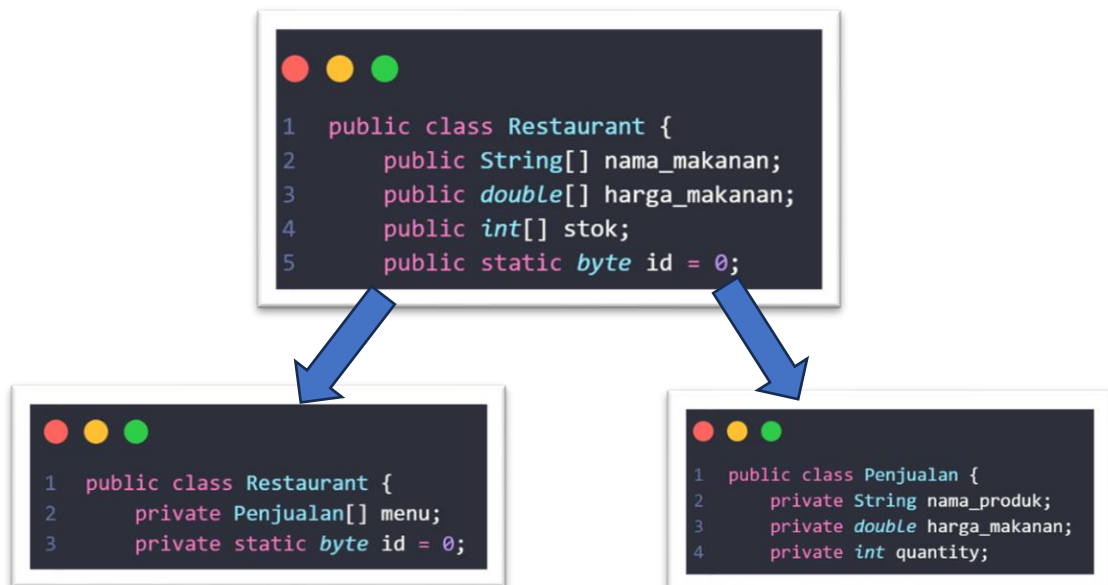
Jawab :

Menurut saya masih ada beberapa yang masih kurang memenuhi konsep OOP sesuai dengan Design Hint di Buku Chapter 4.10. Berikut ini adalah beberapa hal menurut saya yang dimaksud :

- Data atribut yang tidak dibuat private



- Membagi class restaurant menambahkan class penjualan



Berikut adalah perubahan pada class Restaurant, penjelasan berupa coment di dalam program

```
1 public class Restaurant {
2     // atribute/field berupa nama_makanan, harga_makanan, dan stok
3     //dipindahkan ke class Penjualan
4     private Penjualan[] menu;
5     private static byte id = 0;
6
7     public Restaurant() {
8         //perubahan #2 = dari yang awalnya instansiasi per field, tapi karena sudah
9         //diubah ke record menjadi instansiasi record tersebut*/
10        menu = new Penjualan[10];
11    }
12
13    void tambahMenuMakanan(String nama, double harga, int stok) {
14        menu[id] = new Penjualan(nama, harga, stok, 0);
15        nextId();
16    }
17
18    public boolean isOutOfStock(int id) {
19        return this.menu[id] == null || this.menu[id].getStok() == 0;
20    } //return yang awalnya tidak menggunakan getter menjadi menggunakan getter
21    //karena sudah di encapsulasi (private)
22
23    public static void nextId() {
24        id++;
25    }
26
27    public void tampilMenuMakanan() { //merubah tampilan serta pemanggilan menggunakan getter
28        System.out.println("\n|-----|");
29        System.out.println("|          Daftar Produk Makanan          |");
30        System.out.println("|-----|");
31        System.out.println("| No | Nama Produk | Harga Satuan | Stok |");
32        System.out.println("|-----|");
33
34        for (int i = 0; i <= id; i++) {
35            if (!isOutOfStock(i)) {
36                System.out.printf("%-2d | %-12s | " + "Rp. " + " %-6.0f | %-3d |\n", i + 1, menu[i].getNama(), menu[i].getHarga(), menu[i].getStok());
37            }
38        }
39        System.out.println("|-----|");
40    }
41 }
```

Berikut adalah perubahan pada classPenjualan, penjelasan berupa coment di dalam program

```
1 public class Penjualan {
2     //yang sebelumnya ketiga atribut di class Restaurant
3     private String nama_produk;
4     private double harga_makanan;
5     private int quantity;
6
7     //atribut tambahan untuk menyimpan jumlah pesanan yang diinputkan
8     private boolean sudahDipesan;
9     private int jumlah;
10
11    public Penjualan(String nama, double harga, int stok, int jumlah) {
12        this.nama_produk = nama;
13        this.harga_makanan = harga;
14        this.quantity = stok;
15        this.sudahDipesan = false;
16        this.jumlah = jumlah; // Menyimpan jumlah pesanan yang diinputkan
17    }
18
19    // Getter dan setter untuk atribut private
20    public int getJumlahPesanan() {
21        return jumlah;
22    }
23
24    public String getNama() {
25        return nama_produk;
26    }
27
28    public double getHarga() {
29        return harga_makanan;
30    }
31 }
```

2. Ada kebutuhan untuk mengembangkan aplikasi tersebut, dengan menambah fitur pemesanan dan mengurangi setiap stok yang ada. Apakah dengan desain program yang ada dapat dikembangkan? Jika Sulit kemukakan alasannya dan bandingkan dengan desain class hasil modifikasi anda

Jawab :

Pengembangan serta menambahkan fitur pemesanan dan mengurangi stok dapat dilakukan dengan menambahkan atribut baru dan metode/fungsi baru untuk membuat program fitur tersebut. Berikut menampilkan pengembangan pada program ini.

```
1 public void tampilMenuMakanan() { //merubah tampilan serta pemanggilan menggunakan getter
2     System.out.println("\n|-----|");
3     System.out.println("    Daftar Produk Makanan    ");
4     System.out.println("|-----|");
5     System.out.println(" No | Nama Produk | Harga Satuan | Stok |");
6     System.out.println("|-----|");
7
8     for (int i = 0; i <= id; i++) {
9         if (!isOutOfStock(i)) {
10             System.out.printf(" %2d | %-12s | " + "Rp. " + " %6.0f | %3d |\n",
11                 i + 1, menu[i].getNama(), menu[i].getHarga(), menu[i].getStok());
12         }
13     }
14     System.out.println("|-----|");
15 }
16
17 // Metode untuk melakukan pemesanan
18 public void pesanMenu(int idMenu, int quantity) {
19     if (idMenu >= 0 && idMenu <= id && menu[idMenu] != null && menu[idMenu].getStok() >= quantity) {
20         menu[idMenu].setStok(menu[idMenu].getStok() - quantity);
21         menu[idMenu].setSudahDipesan(true); // Menandai produk sebagai sudah dipesan
22         menu[idMenu].setJumlah(quantity); // Menyimpan jumlah pesanan yang diinputkan
23         System.out.println("Pesanan Anda berhasil! Terima kasih telah memesan " + quantity + " " + menu[idMenu].getNama());
24     } else {
25         System.out.println("Maaf, pesanan tidak dapat diproses. Coba lagi.");
26     }
27 }
28
29 // Method untuk menampilkan pesanan dan total harga
30 public void tampilPesanan() {
31     System.out.println("\n|-----|");
32     System.out.println("    Pesanan Konsumen    ");
33     System.out.println("|-----|");
34     System.out.println(" No | Nama Produk | Jumlah | Total Harga |");
35     System.out.println("|-----|");
36
37     double totalHarga = 0;
38
39     for (int i = 0; i <= id; i++) {
40         if (menu[i] != null && menu[i].isSudahDipesan()) {
41             int jumlahDipesan = menu[i].getJumlahPesanan(); // Mengambil jumlah yang benar-benar dipesan
42             double subtotal = menu[i].getHarga() * jumlahDipesan;
43             totalHarga += subtotal;
44             System.out.printf(" %2d | %-12s | %5d | Rp. %6.0f |\n",
45                 i + 1, menu[i].getNama(), jumlahDipesan, subtotal);
46         }
47     }
48
49     System.out.println("|-----|");
50     System.out.println("Total Harga Keseluruhan: Rp. " + totalHarga);
51 }
```

Pada program ini merubah fungsi tampilMenuMakanan yang menjadi proses menampilkan daftar makananya menggunakan perulangan. Serta menambahkan fungsi tampilPesanan yang berisikan jumlah produk yang dipesan serta total harga dari produk dikali dengan jumlah pesanan

Berikut modifikasi/pengembangan pada program utama **RestaurantMain**

```
1 package Revisi;
2 import java.util.Scanner;
3
4 public class RestaurantMain {
5     public static void main(String[] args) {
6         Scanner input = new Scanner(System.in);
7         boolean isSelesai = false;
8
9         Restaurant restoran = new Restaurant(); //daftar produk
10        restoran.tambahMenuMakanan("Bala-bala", 1_000, 20);
11        restoran.tambahMenuMakanan("Gehu", 1_000, 20);
12        restoran.tambahMenuMakanan("Tahu", 1_000, 10);
13        restoran.tambahMenuMakanan("Molen", 1_000, 20);
14        restoran.tambahMenuMakanan("cireng", 3_000, 1);
15
16        do {
17            restoran.tampilMenuMakanan();
18
19            // Pemesanan
20            System.out.print("\nSilakan masukkan ID menu yang ingin dipesan : ");
21            int noMenu = input.nextInt();
22            System.out.print("Masukkan jumlah yang ingin dipesan : ");
23            int jumlah = input.nextInt();
24
25            restoran.pesanMenu(noMenu - 1, jumlah);
26
27            // Tanya apakah ingin memesan lagi
28            System.out.print("\nApakah ada barang lain yang ingin dibeli (y/t) ? : ");
29            String lanjutkan = input.next();
30            if (!lanjutkan.equalsIgnoreCase("y")) {
31                isSelesai = true;
32            }
33        } while (!isSelesai); // Melakukan pengulangan selama isSelesai bernilai false
34
35        restoran.tampilPesanan(); // Menampilkan pesanan setelah pemesanan
36
37    }
38 }
```

Hasil Akhir

Daftar Produk Makanan			
No	Nama Produk	Harga Satuan	Stok
1	Bala-bala	Rp. 1000	20
2	Gehu	Rp. 1000	20
3	Tahu	Rp. 1000	10
4	Molen	Rp. 1000	20
5	cireng	Rp. 3000	1

Silakan masukkan ID menu yang ingin dipesan : 4
Masukkan jumlah yang ingin dipesan : 10
Pesanan Anda berhasil! Terima kasih telah memesan 10 Molen

Apakah ada barang lain yang ingin dibeli (y/t) ? : y

Daftar Produk Makanan			
No	Nama Produk	Harga Satuan	Stok
1	Bala-bala	Rp. 1000	20
2	Gehu	Rp. 1000	20
3	Tahu	Rp. 1000	10
4	Molen	Rp. 1000	10
5	cireng	Rp. 3000	1

Silakan masukkan ID menu yang ingin dipesan : 2
Masukkan jumlah yang ingin dipesan : 5
Pesanan Anda berhasil! Terima kasih telah memesan 5 Gehu

Apakah ada barang lain yang ingin dibeli (y/t) ? : █

Apakah ada barang lain yang ingin dibeli (y/t) ? : t

Pesanan Konsumen			
No	Nama Produk	Jumlah	Total Harga
2	Gehu	5	Rp. 5000
4	Molen	10	Rp. 10000

Total Harga Keseluruhan: Rp. 15000.0

KASUS 2

Pada kasus 2 ini tidak berbeda jauh dengan kasus 1 yaitu membuat sebuah code program yang menampilkan list daftar produk makanan, serta melakukan transaksi

```
1 public class Main {
2     static Produk[] daftarProduk = {
3         new Produk("Batagor", 5000, 20),
4         new Produk("Roti Bakar", 12000, 10),
5         new Produk("Indomi+Telor", 10000, 15),
6         new Produk("Kwetiaw", 12000, 12),
7         new Produk("Nasi Goreng", 12000, 10),
8         new Produk("Air Mineral", 3000, 5),
9         new Produk("Teh Manis", 4000, 15),
10        new Produk("Jus Alpukat", 8000, 5),
11        new Produk("Teh Botol", 5000, 10),
12        new Produk("Kopi", 3000, 11),
```

Pada class main ini terdapat list daftarProduk dengan 10 produk yang disimpan dalam class produk array Produk

```
1 public static void main(String[] args) {
2     Scanner input = new Scanner(System.in);
3     Penjualan transaksi = new Penjualan();
4     boolean isSelesai = false;
5
6     while (!isSelesai) {
7         transaksi.tampilMenuMakanan();
8
9         System.out.print("Silakan masukkan nomor produk yang ingin dibeli : ");
10        int nomorProduk = input.nextInt();
11        System.out.print("Masukkan jumlah produk yang ingin dibeli : ");
12        int jumlah = input.nextInt();
13
14        // proses pemilihan produk
15        if (nomorProduk >= 1 && nomorProduk <= daftarProduk.length) {
16            Produk barangPilihan = daftarProduk[nomorProduk - 1];
17            int stokTersedia = barangPilihan.getStok();
18            if (stokTersedia >= jumlah) {
19                int totalHarga = barangPilihan.getHargaSatuan() * jumlah;
20                transaksi.tambahPesanan(barangPilihan, jumlah, totalHarga);
21                barangPilihan.kurangiStok(jumlah);
22                System.out.println("\nTotal harga : " + totalHarga);
23            }
24        } else {
25            System.out.println("Nomor barang tidak valid.");
26        }
27
28        // Konfirmasi untuk memesan item makanan lain
29        System.out.print("\nApakah ada barang lain yang ingin dibeli (y/t) ? : ");
30        String lanjutkan = input.next();
31        if (lanjutkan.equalsIgnoreCase("t")) {
32            isSelesai = true;
33        }
34    }
35
36    //tampilan untuk pesanan konsumen
37    transaksi.tampilPesananKonsumen(transaksi);
38
39    int totalPembayaran = transaksi.getHarga_total();
40    int diskon = transaksi.hitungDiskon();
41    int totalSetelahDiskon = totalPembayaran - diskon;
42
43    System.out.println("Total pembayaran : " + totalPembayaran);
44    System.out.println("Anda mendapatkan diskon sebesar : " + diskon); // kondisi untuk mendapatkan diskon
45    System.out.println("Total pembayaran setelah diskon : " + totalSetelahDiskon);
46
47    // pengkondisian setelah diskon
48    transaksi.pembayaran(totalSetelahDiskon);
49
50    System.out.println("\n\t\t\tTERIMA KASIH TELAH BERBELANJA\n\n");
51
52    input.close();
53 }
54 }
```


Pada program utama ini adalah alur bagaimana proses cara kerja dan menggunakan fungsi yang dipanggil dari class lain, bagian pertama dibungkus oleh sebuah perulangan WhileDo dengan kondisi benar (!isSelesai) ini ketika konsumen tidak akan melakukan pembelian/transaksi suatu produk dan ingin melanjutkan ke tahap pembayaran sesuai apa apa saja produk yang dipilih oleh konsumen beserta dengan jumlah dari tiap produknya.

Terdapat sebuah objek dari class Penjualan yaitu dengan nama **transaksi**. Selanjutnya objek tersebut memanggil sebuah fungsi untuk menampilkan daftar produk yang disimpan di dalam class Penjualan

```

1 //list dan tampilan daftarProduk
2 public void tampilMenuMakanan(){
3     Produk[] daftarProduk = {
4         new Produk("Batagor", 5000, 20),
5         new Produk("Roti Bakar", 12000, 10),
6         new Produk("Indomie+Telor", 10000, 15),
7         new Produk("Kwetiaw", 12000, 12),
8         new Produk("Nasi Goreng", 12000, 10),
9         new Produk("Air Mineral", 3000, 5),
10        new Produk("Teh Manis", 4000, 15),
11        new Produk("Jus Alpukat", 8000, 5),
12        new Produk("Teh Botol", 5000, 10),
13        new Produk("Kopi", 3000, 11),
14    };
15
16    System.out.println("\n-----|");
17    System.out.println("SELAMAT DATANG DI TOKO ARTHUR");
18    System.out.println("-----|");
19    System.out.println("Daftar Produk Makanan");
20    System.out.println("-----|");
21    System.out.println("No | Nama Produk | Harga Satuan | Stok |");
22    System.out.println("-----|");
23
24    for (int i = 0; i < daftarProduk.length; i++) { //mencetak daftar produk
25        System.out.printf("%-2d | %-20s | %-10s | %-3d |\n", i + 1, daftarProduk[i].getNama_produk(), daftarProduk[i].getHargaSatuan(), daftarProduk[i].getStok());
26    }
27    System.out.println("-----|");
28 }

```

Setelah itu, silahkan memilih produk mana yang ingin dipesan dengan menginputkan nomer dari produk tersebut, serta menginputkan dari produk yang dipilih akan memesan dengan jumlah berapa banyak dari stok yang tersedia.

Selanjutnya terdapat **kondisi apakah konsumen ingin membeli barang lain ?** apabila memasukan kata kunci “y” maka konsumen kembali memilih produk mana yang akan dibeli, jika memasukan kata kunci “t” maka akan lanjut ke proses menampilkan **PesananKonsumen**

```

-----|
SELAMAT DATANG DI TOKO ARTHUR
-----|
Daftar Produk Makanan
-----|
No | Nama Produk | Harga Satuan | Stok |
-----|
1 | Batagor | Rp. 5000 | 20 |
2 | Roti Bakar | Rp. 12000 | 10 |
3 | Indomie+Telor | Rp. 10000 | 15 |
4 | Kwetiaw | Rp. 12000 | 12 |
5 | Nasi Goreng | Rp. 12000 | 10 |
6 | Air Mineral | Rp. 3000 | 5 |
7 | Teh Manis | Rp. 4000 | 15 |
8 | Jus Alpukat | Rp. 8000 | 5 |
9 | Teh Botol | Rp. 5000 | 10 |
10 | Kopi | Rp. 3000 | 11 |
-----|
Silakan masukkan nomor produk yang ingin dibeli : 5
Masukkan jumlah produk yang ingin dibeli : 5

Total harga : 60000

Apakah ada barang lain yang ingin dibeli (y/t) ? : 

```

Gambar di atas adalah contoh ketika konsumen memilih produk, memasukan jumlah produk yang ingin dibeli serta menampilkan total harga dari produk dikali dengan jumlah, dan menampilkan kondisi apakah konsumen ingin membeli barang lain

PESANAN KONSUMEN				
No	Nama barang	Harga Satuan	Banyak Pesanan	Harga Total
1	Batagor	Rp. 5000	10	Rp. 50000
2	Nasi Goreng	Rp. 12000	5	Rp. 60000
Total pembayaran : 110000				
Anda mendapatkan diskon sebesar : 11000				
Total pembayaran setelah diskon : 99000				
Silakan masukkan uang yang dibayarkan : 90000				
Uang yang Anda bayarkan kurang dari total keseluruhan.				
Silakan masukkan uang yang dibayarkan : 99000				
Kembalian: 0				
TERIMA KASIH TELAH BERBELANJA				

Contoh tampilan yang menampilkan pesanan apa saja yang dibeli oleh konsumen serta terdapat banyak pesanan dan harga total setiap produk. Serta menampilkan total pembayaran dari keseluruhan harga total tiap produk

Selain itu terdapat fitur tambahan yaitu mendapatkan **Diskon**, apabila total pembelian lebih dari Rp. 50.000 dan mendapatkan potongan harga sebesar 10%. Serta apabila konsumen memasukan uang kurang atau sama dengan total pembayaran setelah diskon, maka akan kembali memasukan nominal sesuai diskon atau lebih

Berikut Source Code Dari Masing Masing Class

a. Class Main

```
1  public static void main(String[] args) {
2      Scanner input = new Scanner(System.in);
3      Penjualan transaksi = new Penjualan();
4      boolean isSelesai = false;
5
6      while (!isSelesai) {
7          transaksi.tampilMenuMakanan();
8
9          System.out.print("Silakan masukkan nomor produk yang ingin dibeli : ");
10         int nomorProduk = input.nextInt();
11         System.out.print("Masukkan jumlah produk yang ingin dibeli : ");
12         int jumlah = input.nextInt();
13
14         // proses pemilihan produk
15         if (nomorProduk >= 1 && nomorProduk <= daftarProduk.length) {
16             Produk barangPilihan = daftarProduk[nomorProduk - 1];
17             int stokTersedia = barangPilihan.getStok();
18             if (stokTersedia >= jumlah) {
19                 int totalHarga = barangPilihan.getHargaSatuan() * jumlah;
20                 transaksi.tambahPesanan(barangPilihan, jumlah, totalHarga);
21                 barangPilihan.kurangiStok(jumlah);
22                 System.out.println("\nTotal harga : " + totalHarga);
23             }
24
25         } else {
26             System.out.println("Nomor barang tidak valid.");
27         }
28
29         // Konfirmasi untuk memesan item makanan lain
30         System.out.print("\nApakah ada barang lain yang ingin dibeli (y/t) ? : ");
31         String lanjutkan = input.next();
32         if (lanjutkan.equalsIgnoreCase("t")) {
33             isSelesai = true;
34         }
35     }
36
37     //tampilan untuk pesanan konsumen
38     transaksi.tampilPesananKonsumen(transaksi);
39
40     int totalPembayaran = transaksi.getHarga_total();
41     int diskon = transaksi.hitungDiskon();
42     int totalSetelahDiskon = totalPembayaran - diskon;
43
44     System.out.println("Total pembayaran : " + totalPembayaran);
45     System.out.println("Anda mendapatkan diskon sebesar : " + diskon); // kondisi untuk mendaatkan diskon
46     System.out.println("Total pembayaran setelah diskon : " + totalSetelahDiskon);
47
48     // pengkondisian setelah diskon
49     transaksi.pembayaran(totalSetelahDiskon);
50
51     System.out.println("\n\t\tTERIMA KASIH TELAH BERBELANJA\n\n");
52
53     input.close();
54 }
55 }
```

b. Class Penjualan

```
1 public class Penjualan {
2     Scanner input = new Scanner(System.in);
3     private ArrayList<Produk> pesanan;
4     private int harga_total;
5
6     public Penjualan() {
7         pesanan = new ArrayList<>();
8         harga_total = 0;
9     }
10
11     public void tambahPesanan(Produk produkPilihan, int banyakPesanan, int hargaTotal) {
12         produkPilihan.setBanyakPesanan(banyakPesanan); // Set jumlah pesanan pada barang
13         pesanan.add(produkPilihan);
14         harga_total += hargaTotal;
15     }
16
17     public int getHarga_total() {
18         return harga_total;
19     }
20
21     public int hitungDiskon() {
22         if (harga_total > 50000) {
23             return harga_total / 10; // Diskon 10%
24         } else {
25             return 0;
26         }
27     }
28
29     public ArrayList<Produk> getPesanan() { // Getter untuk pesanan
30         return pesanan;
31     }
32
33     // Metode untuk mendapatkan banyak pesanan berdasarkan nama barang
34     public int getBanyakPesananByName(String namaProduk) {
35         int jumlah = 0;
36         for (Produk produk : pesanan) {
37             if (produk.getNama_produk().equals(namaProduk)) {
38                 jumlah += produk.getBanyakPesanan();
39             }
40         }
41         return jumlah;
42     }
43 }
```

c. Class Produk

```
1  package ClassClass;
2
3  public class Produk {
4      private String nama_produk;
5      private int hargaSatuan;
6      private int banyakPesanan;
7      private int stok;
8
9      public Produk(String nama_produk, int hargaSatuan, int stok) {
10         this.nama_produk = nama_produk;
11         this.hargaSatuan = hargaSatuan;
12         this.stok = stok;
13     }
14
15     public void kurangiStok(int jumlah){
16         this.stok -= jumlah;
17     }
18
19     public int getBanyakPesanan() { // Getter untuk banyakPesanan
20         return banyakPesanan;
21     }
22
23     public void setBanyakPesanan(int banyakPesanan) { // Setter untuk banyakPesanan
24         this.banyakPesanan = banyakPesanan;
25     }
26
27     public String getNama_produk() { // Getter untuk nama_produk barang
28         return nama_produk;
29     }
30
31     public int getHargaSatuan() { // Getter untuk harga satuan barang
32         return hargaSatuan;
33     }
34
35     public int getStok() {
36         return stok;
37     }
38
39     public void setStok(int stok) {
40         this.stok = stok;
41     }
42 }
```