**LAPORAN HASIL PRAKTIKUM**

**MATA KULIAH :**

**TEKNIK PEMROGRAMAN – PERTEMUAN 4**

Logo, company name

Description automatically generated

Disusun Oleh :

|  |  |  |
| --- | --- | --- |
| **NIM** | **NAMA** | **KELAS** |
| 221524041 | Ferdi Ahmad Ariesta | 1B |

**SARJANA TERAPAN PROGRAM STUDI TEKNIK INFORMATIKA JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA POLITEKNIK NEGERI BANDUNG**

Daftar Isi

[A. Studi Kasus 1 3](#_Toc130769998)

[1.1 Source Code Program 3](#_Toc130769999)

[1.2 Permasalahan yang dihadapi 6](#_Toc130770000)

[1.3 Solusi permasalahan yang dihadapi 6](#_Toc130770001)

[1.4 Nama teman yang membantu 6](#_Toc130770002)

[B. Studi Kasus 2 6](#_Toc130770003)

[2.1 Source Code Program 6](#_Toc130770004)

[2.2 Permasalahan yang dihadapi 14](#_Toc130770005)

[2.3 Solusi permasalahan yang dihadapi 15](#_Toc130770006)

[2.4 Nama teman yang membantu 15](#_Toc130770007)

# Studi Kasus 1

Restaurant dan Menu

## Source Code Program

|  |
| --- |
| Menu.java |
| public class Menu {  private String nama\_makanan;  private double harga\_makanan;  private int stok;    public String getNama\_makanan() {  return nama\_makanan;  }  public void setNama\_makanan(String nama\_makanan) {  this.nama\_makanan = nama\_makanan;  }  public double getHarga\_makanan() {  return harga\_makanan;  }  public void setHarga\_makanan(double harga\_makanan) {  this.harga\_makanan = harga\_makanan;  }  public int getStok() {  return stok;  }  public void setStok(int stok) {  this.stok = stok;  }  } |

|  |
| --- |
| Restaurant.java |
| public class Restaurant {  private Menu[] menus;  private static byte id = -1;    public Restaurant() {  menus = new Menu[10];  }    public void tambahMenuMakanan(String nama, double harga, int stok) {  this.nextId();  menus[id] = new Menu();  menus[id].setNama\_makanan(nama);  menus[id].setHarga\_makanan(harga);  menus[id].setStok(stok);  }    public void tampilkanMenuMakanan() {  for(int i = 0; i <= id; i++) {  if(!isOutOfStock(i)) {  System.out.println(menus[i].getNama\_makanan() + "[" + menus[i].getStok() + "]" + "Rp. " + menus[i].getHarga\_makanan());  }  }  }    public void pesanMakanan(String nama, int jumlah) {  int index = indexMakanan(nama);  if(!(isOutOfStock(index))) {  if(jumlah <= menus[index].getStok()) {  menus[index].setStok(menus[index].getStok() - jumlah);    } else {  System.out.println("Stok Tidak Cukup");  }  } else {  System.out.println("Stok Kosong");  };  }    public int indexMakanan(String nama) {  int index = 0;    while(!(nama.equals(menus[index].getNama\_makanan()))) {  index++;  }    return index;  }    public boolean isOutOfStock(int id) {  if(menus[id].getStok() == 0) {  return true;  } else {  return false;  }  }    public Menu[] getMenus() {  return menus;  }  public void nextId() {  id++;  }  } |

|  |
| --- |
| RestaurantMain.java |
| public class RestaurantMain {  public static void main(String[] args) {  Restaurant menu = new Restaurant();    menu.tambahMenuMakanan("Gehu", 1000, 20);  menu.tambahMenuMakanan("Tahu", 1000, 0);  menu.tambahMenuMakanan("Molen", 1000, 20);  menu.pesanMakanan("Gehu", 19);  menu.tampilkanMenuMakanan();  }  } |

|  |
| --- |
| Output Program |
|  |

## Permasalahan yang dihadapi

a. Amati desain setiap class, Apakah desain class sudah memenuhi konsep OOP yang

benar?

b. 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

## Solusi permasalahan yang dihadapi

a. Dalam class tersebut belum memenuhi aturan oop yang tepat karena tidak menggunakan sebuah aturan always keep data private. Caranya adalah dengan mengubah

statement menjadi private. Di dalam kode terdapat sebuah statement statement yang

merupakan sebuah menu sehingga kita dapat memisahkannya menjadi sebuah class

menu, kita perlu menggunakan setter and getter tapi tetap saja kita perlu memilahnya

sehingga hanya setter dan getter yang akana digunakan saja yang dibuat

b. Dapat. Kita dapat mengembangkan aplikasi tersebut dengan menambah fitur

pemesanan dan mengurangi stok makanan yang dipesan, jika makanan yang dipesan

kosong maka akan menampilkan pesan “Stok Kosong”, jika makanan yang dipesan

stok nya kurang maka akan menampilkan pesan "Stok Tidak Cukup”. Untuk

membuatnya ktia perlu membuat sebuah modul baru yang bernama

tambahkanPesanan().

## Nama teman yang membantu

Muhammd Agim 221524046

# Studi Kasus 2

Modifikasi Restaurant dan Menu

## 2.1 Source Code Program

|  |
| --- |
| Penjualan.java |
| public class Penjualan extends Produk {  private int[] totalPerMakanan;  private int[] jumlahPerMakanan;    public Penjualan() {  super();  totalPerMakanan = new int[10];  jumlahPerMakanan = new int[10];  }    public void tambahMenuMakanan(String nama, int harga, int stok) {  nextId();;  super.setNamaMakanan(nama);  super.setHargaMakanan(harga);  super.setStok(stok);  }    public void hapusPesanan(String nama, int jumlah) {  int index = super.indexMakanan(nama.toLowerCase());  if(jumlahPerMakanan[index] != 0) {  if(jumlah <= super.getStok()[index]) {  super.setStok(super.getStok()[index] - jumlah, index);  jumlahPerMakanan[index] -= jumlah;  totalPerMakanan[index] -= (super.getHargaMakanan()[index] \* jumlah);  System.out.println("Menghapus Pesanan Berhasil");  } else {  System.out.println("Gagal Menhapus Pesan");  }  } else {  System.out.println("Pesanan Tidak Ada");  }  }    public void tambahkanPesanan(String nama, int jumlah) {  int index = super.indexMakanan(nama.toLowerCase());  if(!(super.isOutOfStock(index))) {  if(jumlah <= super.getStok()[index]) {  super.setStok(super.getStok()[index] - jumlah, index);  jumlahPerMakanan[index] += jumlah;  totalPerMakanan[index] += (super.getHargaMakanan()[index] \* jumlah);  System.out.println("Menambah Pesanan Berhasil");  } else {  System.out.println("Stok Tidak Cukup");  }  } else {  System.out.println("Stok Kosong");  }  }    public void tampilkanMenuTersedia() {  int i;  for(i = 0;i <= super.getId();i++) {  System.out.printf("%-2d. %-15s Rp. %d\n", i + 1, super.getNamaMakanan()[i], super.getHargaMakanan()[i]);  }  }    public void tampilkanMenuYangDiPesan() {  int i;  int angka = 0;  System.out.println("No | Nama Produk | Jumlah | Harga |");  for(i = 0;i < 10;i++) {  if(totalPerMakanan[i] != 0) System.out.printf("%-3d|%-20s | %-11d| Rp. %-5d |\n",++angka, super.getNamaMakanan()[i], jumlahPerMakanan[i], totalPerMakanan[i]);  }  System.out.printf("Total Harga | %d |\n",hitungTotalHarga());  }    public void kosongkanPesanan() {  for(int i = 0; i < 10;i++) {  totalPerMakanan[i] = 0;  jumlahPerMakanan[i] = 0;  }  }    public int hitungTotalHarga() {  int total = 0;  for(int i = 0; i < 10;i++) {  if(totalPerMakanan[i] != 0) total += totalPerMakanan[i];  }    return total;  }    } |

|  |
| --- |
| PenjualanMain.java |
| import java.util.Scanner;  public class PenjualanMain {  public static void main(String[] args) {  Penjualan resto = new Penjualan();  Scanner in = new Scanner(System.in);  String pesanan = "";  String lagi;  int jumlah;  int pilih;  int uang;    resto.tambahMenuMakanan("Siomay", 17000, 20);  resto.tambahMenuMakanan("Mie", 12000, 34);  resto.tambahMenuMakanan("Batagor", 15000, 0);  resto.tambahMenuMakanan("Gehu", 10000, 7);  resto.tambahMenuMakanan("Ayam", 25000, 15);  resto.tambahMenuMakanan("Cappucino", 15000, 21);  resto.tambahMenuMakanan("Matcha", 18000, 13);  resto.tambahMenuMakanan("ThaiTea", 17000, 22);  resto.tambahMenuMakanan("Chocolatte", 20000, 3);  resto.tambahMenuMakanan("MilkTea", 19000, 10);    do {    do {  resto.tampilkanMenuTersedia();  System.out.println("Ingin Memesan Apa ?");  pilih = in.nextInt();  switch(pilih) {  case 1: pesanan = new String("Siomay");break;  case 2: pesanan = new String("Mie");break;  case 3: pesanan = new String("Batagor");break;  case 4: pesanan = new String("Gehu");break;  case 5: pesanan = new String("Ayam");break;  case 6: pesanan = new String("Cappucino");break;  case 7: pesanan = new String("Matcha");break;  case 8: pesanan = new String("ThaiTea");break;  case 9: pesanan = new String("Chocolatte");break;  case 10: pesanan = new String("MilkTea");break;  }  System.out.println("Ingin Memesan Berapa ?");  jumlah = in.nextInt();  resto.tambahkanPesanan(pesanan, jumlah);  System.out.printf("Apakah Ingin Memesan Makanan Lagi ? (Y/N)" );  lagi = in.next();  } while(lagi.toUpperCase().equals("Y"));  resto.tampilkanMenuYangDiPesan();  System.out.println("Apakah Ingin Menghapus Pesanan ? (Y/N)");  lagi = in.next();  while(lagi.toUpperCase().equals("Y")) {  System.out.println("Ingin Menghapus Pesanan Apa ? Ketikan Nama Pesanan!");  pesanan = in.next();  System.out.println("Ingin Menhapus Berapa ?");  jumlah = in.nextInt();  resto.hapusPesanan(pesanan, jumlah);  resto.tampilkanMenuYangDiPesan();  System.out.println("Apakah Ingin Menghapus Pesanan Lagi ? (Y/N)");  lagi = in.next();  }  for(;;) {  resto.tampilkanMenuYangDiPesan();  System.out.println("Masukan Uang Anda...");  uang = in.nextInt();  if(uang >= resto.hitungTotalHarga()) {  System.out.println("Pembayaran Sukses!!!");  break;  }  System.out.println("Pembayaran Gagal, Uang Kurang!!!");  }  System.out.println("Apakah Ingin Memesan Pesanan Lagi ? (Y/N)");  lagi = in.next();  if(lagi.toUpperCase().equals("Y")) resto.kosongkanPesanan();  } while(lagi.toUpperCase().equals("Y"));  }  } |

|  |
| --- |
| Produk.java |
| public class Produk {  private String[] namaMakanan;  private int[] hargaMakanan;  private int[] stok;  private int id = -1;    public Produk() {  namaMakanan= new String[10];  hargaMakanan = new int[10];  stok = new int[10];  }      public void tampilkanMenuTersedia() {  int i;  for(i = 0;i <= id;i++) {  System.out.printf("%d. %s Rp. %d\n", i + 1, namaMakanan[i], hargaMakanan[i]);  }  }    public int indexMakanan(String nama) {  int index = 0;    while(!(nama.equals(namaMakanan[index].toLowerCase()))) {  index++;  }    return index;  }      public boolean isOutOfStock(int id) {  if(stok[id] == 0) {  return true;  } else {  return false;  }  }    public void nextId() {  id++;  }  public String[] getNamaMakanan() {  return namaMakanan;  }  public void setNamaMakanan(String namaProduk) {  this.namaMakanan[id] = namaProduk;  }  public int[] getHargaMakanan() {  return hargaMakanan;  }  public void setHargaMakanan(int hargaProduk) {  this.hargaMakanan[id] = hargaProduk;  }  public int[] getStok() {  return stok;  }  public void setStok(int stok) {  this.stok[id] = stok;  }    public void setStok(int stok, int index) {  this.stok[index] = stok;  }  public int getId() {  return id;  }    } |

|  |
| --- |
| Output Program |
|  |

## 2.2 Permasalahan yang dihadapi

- Membuat tampilan menu

- Membuat modul input dari user

- Variabel yang dibutuhkan untuk membentuk program pesan

## 2.3 Solusi permasalahan yang dihadapi

- Membuat class produk sebagai menu yang tersedia dengan statement setter dan getter

- Membuat main untuk main program dan menggunakan case

- Membuat algoritma penjualan di class penjualan agar penghitungan stok dan uang dapat dilakukan.

## 2.4 Nama teman yang membantu

Muhammad Agim 221524046