**PRAKTIKUM ALGORITMA DAN STRUKTUR DATA**

**JOBSHEET PERTEMUAN KE-2**

****

**NAMA : ALVINO VALERIAN D.R**

**KELAS : 1A**

**NO. ABSEN : 05**

**NIM : 2341720027**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

**2024**

**2.1.3 Pertanyaan**1. Sebutkan dua karakteristik class atau object!  
2. Perhatikan class **Buku** pada Praktikum 1 tersebut, ada berapa atribut yang dimiliki oleh class  
Buku? Sebutkan apa saja atributnya!  
3. Ada berapa method yang dimiliki oleh class tersebut? Sebutkan apa saja methodnya!  
4. Perhatikan method **terjual()** yang terdapat di dalam class **Buku**. Modifikasi isi method tersebut  
sehingga proses pengurangan hanya dapat dilakukan jika stok masih ada (lebih besar dari 0)!  
5. Menurut Anda, mengapa method **restock()** mempunyai satu parameter berupa bilangan int?  
6. **Commit dan push kode program ke Github**

**Jawaban**

*1.Class masih berupa rancangan atau template yang bersifat umum sedangkan objek merupakan penggamabaran dari objek yang ada di dunia nyata dan sudah berbentuk class. Objek bersifat lebih spesifik.*

*2. halaman,stok,harga; judul,pengarang;*

*3. Terjual,restock,ganti harga*

**4.** void terjual(int jml){

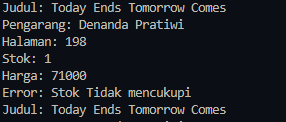
if(jml <= stok){

stok -= jml;

}else{

System.out.println("Error: Stok Tidak mencukupi ");

}



*5. Parameter int digunakan untuk menentukan berapa banyak stok yang ingin ditambahkan.*

**2.2.3 Pertanyaan**1. Pada class **BukuMain**, tunjukkan baris kode program yang digunakan untuk proses instansiasi!  
Apa nama object yang dihasilkan?  
2. Bagaimana cara mengakses atribut dan method dari suatu objek?  
3. Mengapa hasil output pemanggilan method **tampilInformasi()** pertama dan kedua berbeda?

Jawaban

1. Nama objek yang dihasilkan adalah bk1

buku05 bk1 = new buku05();

2.

**Atribut:** Gunakan nama objek diikuti dengan tanda titik (.) dan nama atribut.

namaObject.namaAtribut = nilai;

**Method:** Gunakan nama objek diikuti dengan tanda titik (.) dan nama method, diikuti dengan tanda kurung dan argumen (jika ada).

namaObject.namaMethod();

3. Perbedaan output pada pemanggilan method tampilInformasi() pertama dan kedua disebabkan oleh perubahan nilai atribut stok dan harga yang terjadi setelah pemanggilan pertama.

**2.3.3 Pertanyaan**1. Pada class **Buku** di Percobaan 3, tunjukkan baris kode program yang digunakan untuk  
mendeklarasikan konstruktor berparameter!  
2. Perhatikan class **BukuMain**. Apa sebenarnya yang dilakukan pada baris program berikut?  
3. Hapus konstruktor default pada class **Buku**, kemudian compile dan run program. Bagaimana  
hasilnya? Jelaskan mengapa hasilnya demikian!  
4. Setelah melakukan instansiasi object, apakah method di dalam class **Buku** harus diakses  
secara berurutan? Jelaskan alasannya!  
5. Buat object baru dengan nama **buku<NamaMahasiswa>** menggunakan konstruktor  
berparameter dari class **Buku**!  
6. **Commit dan push kode program ke Github**

Jawaban

1.

public buku05(String jud, String pg, int hal, int stok, int har){}

2. Baris program tersebut membuat objek baru dari class buku05 dengan nama bk2 dan menginisialisasi nilai atribut judul, pengarang, halaman, stok, dan harga dengan nilai yang diberikan.

3. jika konstruktor default dihapus, tidak ada cara untuk membuat objek baru dari class Buku tanpa memberikan nilai parameter. program tidak dapat menemukan cara untuk membuat objek bk1 dan akan menghasilkan error



4. Tidak, method di dalam class Buku tidak harus diakses secara berurutan. Anda dapat mengakses method apa pun dalam urutan apa pun yang Anda inginkan. Setiap method dalamclass adalah fungsi indpenden yang dapat dipanggil secara individual urutan pemanggilan tidak mempengaruhi cara kerja program dan bisa memanggil sesuai kebutuhan program

5.

buku05 bukuAlvino = new buku05("Dilan Ancika","Alvino",200,12,50000);

bukuAlvino.terjual(4);

bukuAlvino.tampilInformasi();



1.

public class buku05 {

String judul,pengarang;

int halaman,stok,harga,hargaTotal,bayar;

double diskon;

void tampilInformasi() {

System.out.println("Judul: " + judul);

System.out.println("Pengarang: " +pengarang);

System.out.println("Halaman: " +halaman);

System.out.println("Stok: " + stok);

System.out.println("Harga: "+ harga);

System.out.println("Harga Total: "+ hargaTotal);

System.out.println("Diskon: "+ diskon );

System.out.println("Harga Bayar: "+ bayar);

}

void terjual(int jml){

if(jml <= stok){

stok -= jml;

hitungHargaTotal(jml);

}else{

System.out.println("Error: Stok Tidak mencukupi ");

}

}

void restock(int jml){

stok += jml;

}

void gantiHarga(int hrg){

harga = hrg;

}

public buku05(){}

public buku05(String jud, String pg, int hal, int stok, int har){

judul = jud;

pengarang = pg;

halaman = hal;

this.stok = stok;

harga = har;

}

void hitungHargaTotal(int jml){

hargaTotal = harga \*jml;

hitungDiskon();

}

void hitungDiskon(){

if (hargaTotal > 150000){

diskon = (int) (hargaTotal\*0.12);

}else if(hargaTotal >= 75000){

diskon =(int) (hargaTotal\*0.5);

}

hitungHargaBayar();

}

void hitungHargaBayar(){

bayar= (int)(hargaTotal-diskon);

tampilInformasi();

}

}

public class bukuMain {

public static void main(String[] args) {

// mengakses atribut

buku05 bk1 = new buku05();

bk1.judul = "Today Ends Tomorrow Comes";

bk1.pengarang = "Denanda Pratiwi";

bk1.halaman = 198;

bk1.stok = 13;

bk1.harga = 71000;

// mengakses method

bk1.tampilInformasi();

bk1.terjual(5);

bk1.gantiHarga(60000);

bk1.tampilInformasi();

buku05 bk2 =new buku05("Self Reward", "Maheera Ayesha", 160, 29, 59000);

bk2.terjual(11);

bk2.tampilInformasi();

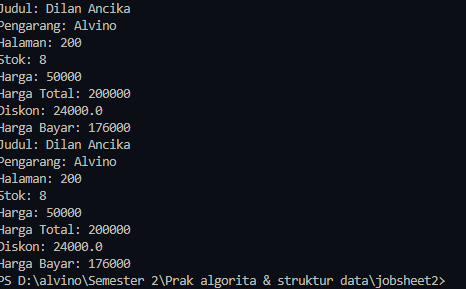
buku05 bukuAlvino = new buku05("Dilan Ancika","Alvino",200,12,50000);

bukuAlvino.terjual(4);

bukuAlvino.tampilInformasi();

}

}



2.

public class dragon {

int x, y, width, height;

public dragon(int x, int y, int width, int height){

this.x = x;

this.y = y;

this.width = width;

this.height = height;

}

void moveLeft() {

if (x > 0 && x < width) {

x--;

}

}

void moveRight() {

if (x > 0 && x < width) {

x++;

}

}

void moveUp() {

if (y > 0 && y < height) {

y++;

}

}

void moveDown() {

if (y > 0 && y < height) {

y--;

}

}

void printPosition() {

System.out.println("Letak Dragon: (" + x + ", " + y + ")");

if (x >= width || y >= height || x == 0 || y == 0) {

detectCollision();

}

}

void detectCollision() {

if (x == width || y == height || x == 0 || y == 0) {

System.out.println("Game Over!");

System.exit(0);

}

}

}

public class dragonMain {

public static void main(String[] args) {

dragon console = new dragon(2,5,5,10);

console.detectCollision();

console.moveRight();

console.printPosition();

console.moveUp();

console.printPosition();

console.moveLeft();

console.printPosition();

console.moveLeft();

console.printPosition();

console.moveLeft();

console.printPosition();

}

}

