**PRAKTIKUM ALGORITMA DAN STRUKTUR DATA**

**JOBSHEET PERTEMUAN KE-2**

****

**NAMA : NABEEL NIZAM**

**KELAS : 1A**

**NO. ABSEN : 22**

**NIM : 2341720155**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

**2024**

**PERCOBAAN 1**

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. Ada 5, diantaranya judul,pengarang,halaman,stok,harga
3. Ada 4 yaitu tampilInformasi, terjual, restock, dan gantiHarga

void terjual(int jml){ if (stok<0) { stok -= jml; }

1. Karena parameter tersebut mewakili jumlah tambahan stok buku yang akan ditambahkan ke stok yang ada.

**PERCOBAAN 2**

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 yaitu bk1

buku22 bk1 = new buku22();

1. Menggunakan tanda titik “.” Sebagai penghubung antara namaObjek dengan namaAtribut/Method. Format penulisan seperti dibawah ini

Cara mengakses atribut :

namaObject.namaAtribut = nilai;

Cara mengakses Method

namaObject.namaMethod();

1. Karena diantara pemanggilan pertama dan kedua terjadi pengisian lagi yang mana setelah pemanggilan pertama terjadi pembaruan data.

\

**PERCOBAAN 3**

Pertanyaan

1. Pada class Buku di Percobaan 3, tunjukkan baris kode program yang digunakan untuk

mendeklarasikan konstruktor berparameter!

1. Perhatikan class BukuMain. Apa sebenarnya yang dilakukan pada baris program berikut?



1. Hapus konstruktor default pada class Buku, kemudian compile dan run program. Bagaimana

hasilnya? Jelaskan mengapa hasilnya demikian!

1. Setelah melakukan instansiasi object, apakah method di dalam class Buku harus diakses

secara berurutan? Jelaskan alasannya!

1. Buat object baru dengan nama buku<NamaMahasiswa> menggunakan konstruktor

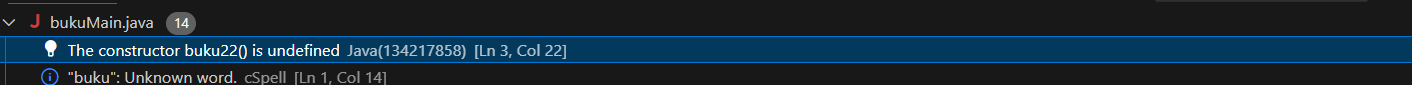
berparameter dari class Buku!

1. Commit dan push kode program ke Github

Jawaban

1. Konstruktor berparameter
2. Melakukan kontstruktor saat melak ukan instansiasi dengan menggunakan konstruktor berparameter

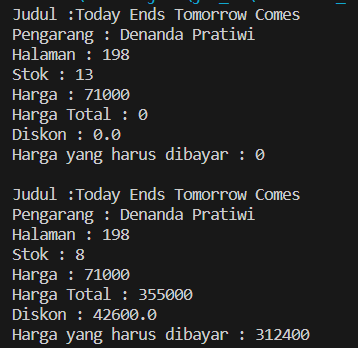
buku22 bukuNabeel = new buku22("The laws of power", "Dion Yulianto", 194, 1, 650000); bukuNabeel.tampilInformasi();

1. Program tidak bisa dirunning dikarenakan kontstruktor tidak didefinisikan pada class Buku. Selain iu hal ini juga terjadi karena kita menggunakan konstruktor tanpa parameter buku22() di tempat lain dalam kode kita, tetapi tidak ada konstruktor tanpa parameter yang didefinisikan dalam kelas buku22.
2. Tidak, method di dalam sebuah class tidak harus diakses secara berurutan.Alasan utama untuk memisahkan dan mengatur method sesuai dengan fungsi atau tujuan masing-masing adalah untuk meningkatkan keterbacaan dan keterorganisasian kode. Dengan cara ini, saat seseorang membaca atau bekerja dengan kode tersebut, mereka dapat dengan mudah menemukan dan memahami apa yang dilakukan oleh setiap method tanpa harus mencari-cari di seluruh kode.

public buku22(String jud,String pg, int hal, int stok, int har){ judul = jud; pengarang = pg; halaman = hal; this.stok = stok; harga = har; }

**LATIHAN PRAKTIKUM 4**

public class buku22{ String judul,pengarang; int halaman,stok,harga, hargaTotal, hargaBayar; 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 yang harus dibayar : " + hargaBayar); System.out.println(); } void terjual(int jml){ if (stok > 0) { stok -= jml; hitungHargaTotal(jml); } } void restock(int jml){ stok += jml; } void gantiHarga(int hrg){ harga = hrg; } void hitungHargaTotal(int jml){ hargaTotal = harga\*jml; hitungDiskon(); } void hitungDiskon(){ if (hargaTotal > 150000) { diskon = (int)(hargaTotal \* 0.12); }else if (hargaTotal >= 75000 && hargaTotal <= 150000 ) { diskon = (int)(hargaTotal \* 0.05); }else if (hargaTotal < 72000) { System.out.println("Tidak ada diskon"); } hitungHargaBayar(); } void hitungHargaBayar(){ hargaBayar = (int)(hargaTotal - diskon); tampilInformasi(); } public buku22(){ } public buku22(String jud,String pg, int hal, int stok, int har){ judul = jud; pengarang = pg; halaman = hal; this.stok = stok; harga = har; }}



public buku22(){ } public buku22(String jud,String pg, int hal, int stok, int har){ judul = jud; pengarang = pg; halaman = hal; this.stok = stok; harga = har; }}

2.

public class Dragon { int x, y, width, height; void moveLeft() { if (x < 0 && x > width) { x -= 1; } else detectCollision(); } void moveRight() { if (x < 0 && x > width) { x += 1; } else detectCollision(); } void moveUp() { if (y < 0 && y > height) { y += 1; } else detectCollision(); } void moveDown() { if (y < 0 && y > height) { y -= 1; } else detectCollision(); } void printPosition() { } void detectCollision() { if (x == width || y == height) { System.out.println("GAME OVER!!!"); } }}