**SAFRIZAL RAHMAN  
Laporan Praktikum**

**Algoritma Dan Struktur Data**

Link Github: <https://github.com/safrizalrahman46/JOBSHEET2_SEM2>

1.

public class BUKUSAFRIZAL {

    public static void main(String[] args) {

*// tampilinformasi;*

    }

*//Atribut*

    String jdl, pengarang;

    int halaman, stok, harga;

    void tampilinformasi(){

        System.out.println("Judul BUku Yaw " + jdl);

        System.out.println("===============================================");

        System.out.println("===============================================");

        System.out.println("Pengarang Buku yaw" + pengarang);

        System.out.println("Jumlah Halaman Bukunye bERAPE" + halaman);

        System.out.println("===============================================");

        System.out.println("===============================================");

        System.out.println("Sisa Stok Berapa Bos" + stok);

        System.out.println("HargaNyee" +  harga);

        System.out.println("===============================================");

        System.out.println("===============================================");

    }

    void terjual(int jmlh){

        stok -= jmlh;

    }

    void Kulakan(int jmlh){

        stok += jmlh;

    }

    void gantiHarga(int jmlh){

        stok -= jmlh;

    }

}

1. Dua karakteristik dari class atau object adalah:

2. Encapsulation: Class dalam pemrograman berorientasi objek mengenkapsulasi data dan fungsi yang beroperasi pada data tersebut dalam satu unit. Ini berarti bahwa class mendefinisikan jenis data dan metode (fungsi) yang dapat beroperasi pada jenis data tersebut.

Instantiation: Class adalah blueprint atau template dan objek adalah instance dari class. Ketika class didefinisikan, hanya spesifikasi untuk objek yang ditentukan. Oleh karena itu, tidak ada memori atau penyimpanan yang dialokasikan.

3. Class Buku memiliki lima atribut, yaitu: jdl (judul), pengarang, halaman, stok, dan harga.

4. Class tersebut memiliki empat metode, yaitu: tampilinformasi(), terjual(int jmlh), Kulakan(int jmlh), dan gantiHarga(int jmlh).

Berikut adalah modifikasi pada method terjual() sehingga proses pengurangan hanya dapat dilakukan jika stok masih ada (lebih besar dari 0):

    void terjual(int jmlh){

        if(stok > 0){

            stok -= jmlh;

        } else {

            System.out.println("Stok habis, Belum Kulakan.");

        }

    }

5. Method restock() memiliki satu parameter berupa bilangan int karena metode tersebut digunakan untuk menambah jumlah stok buku. Parameter tersebut menunjukkan jumlah buku yang ditambahkan ke stok. Bilangan int dipilih karena jumlah buku adalah nilai diskrit dan tidak bisa berupa bilangan desima

2.

import java.util.Scanner;

public class BukuMainSAFRIZAL {

    public static void main(String[] args) {

        Buku bk1 = new Buku();

        bk1.judul = "Today Ends Tomorrow Comes";

        bk1.pengarang = "SAFRIZAL RAHMAN";

        bk1.halaman = 198;

        bk1.stok = 13;

        bk1.harga = 71000;

        bk1.tampilInformasi();

        bk1.terjual(5);

        bk1.gantiHarga(60000);

        bk1.tampilInformasi();

    }

}

            class Buku {

                String judul;

                String pengarang;

                int halaman;

                int stok;

                int harga;

*// Konstruktor default*

                Buku() {}

    void tampilInformasi() {

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

        System.out.println("---------------------");

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

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

        System.out.println("---------------------");

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

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

    }

    void terjual(int jumlahTerjual) {

        if (jumlahTerjual <= stok) {

            stok -= jumlahTerjual;

            System.out.println(jumlahTerjual + " buku terjual.");

        } else {

            System.out.println("Stok tidak mencukupi.");

        }

    }

*// Metode untuk mengubah harga buku*

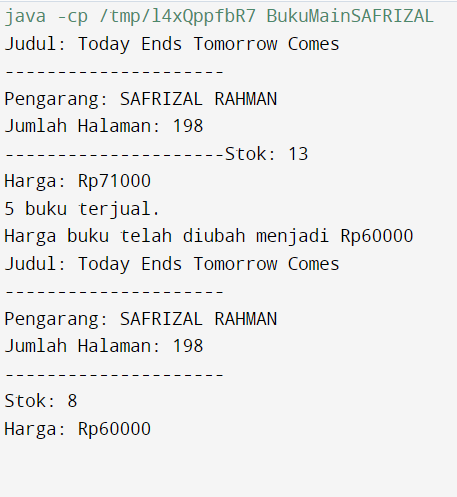
    void gantiHarga(int hargaBaru) {

        harga = hargaBaru;

        System.out.println("Harga buku telah diubah menjadi Rp" + harga);

    }

}



1. Baris kode program yang digunakan untuk proses instansiasi adalah `Buku bk1 = new Buku();`. Objek yang dihasilkan memiliki nama `bk1`.

2. Anda dapat mengakses atribut dan metode dari suatu objek dengan menggunakan operator titik (`.`). Misalnya, untuk mengakses atribut `judul` dari objek `bk1`, Anda dapat menulis `bk1.judul`. Untuk memanggil metode `tampilInformasi()` dari objek `bk1`, Anda dapat menulis `bk1.tampilInformasi();`.

3. Hasil output pemanggilan metode `tampilInformasi()` pertama dan kedua berbeda karena ada beberapa operasi yang dilakukan di antara dua pemanggilan tersebut. Setelah pemanggilan pertama, metode `terjual(5)` dipanggil, yang mengurangi stok buku sebanyak 5. Kemudian, metode `gantiHarga(60000)` dipanggil, yang mengubah harga buku menjadi 60000. Oleh karena itu, ketika `tampilInformasi()` dipanggil untuk kedua kalinya, nilai stok dan harga telah berubah.

public class BukuMainSafrizal {

    public static void main(String[] args) {

        BukuSafrizal bk1 = new BukuSafrizal();

        bk1.jdl = "Today Ends Tomorrow Comes";

        bk1.pengarang = "SAFRIZAL RAHMAN";

        bk1.halaman = 198;

        bk1.stok = 13;

        bk1.harga = 71000;

        bk1.tampilinformasi();

        bk1.terjual(5);

        bk1.gantiHarga(60000);

        bk1.tampilinformasi();

        System.out.println("+++++++++++++++++++++++++++++++++++++++++++");

        BukuSafrizal bk2 = new BukuSafrizal("20 th Century Girl", "Kim Boesook", 160, 29,  45000);

        bk2.terjual(11);

        bk2.tampilinformasi();

    }

}

*// class Buku {*

*//     String judul;*

*//     String pengarang;*

*//     int halaman;*

*//     int stok;*

*//     int harga;*

*//     // Konstruktor default*

*//     Buku() {}*

*// void tampilInformasi() {*

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

*//     System.out.println("---------------------");*

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

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

*//     System.out.println("---------------------");*

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

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

*// }*

*// void terjual(int jumlahTerjual) {*

*//     if (jumlahTerjual <= stok) {*

*//         stok -= jumlahTerjual;*

*//         System.out.println(jumlahTerjual + " buku terjual.");*

*//     } else {*

*//         System.out.println("Stok tidak mencukupi.");*

*//     }*

*// }*

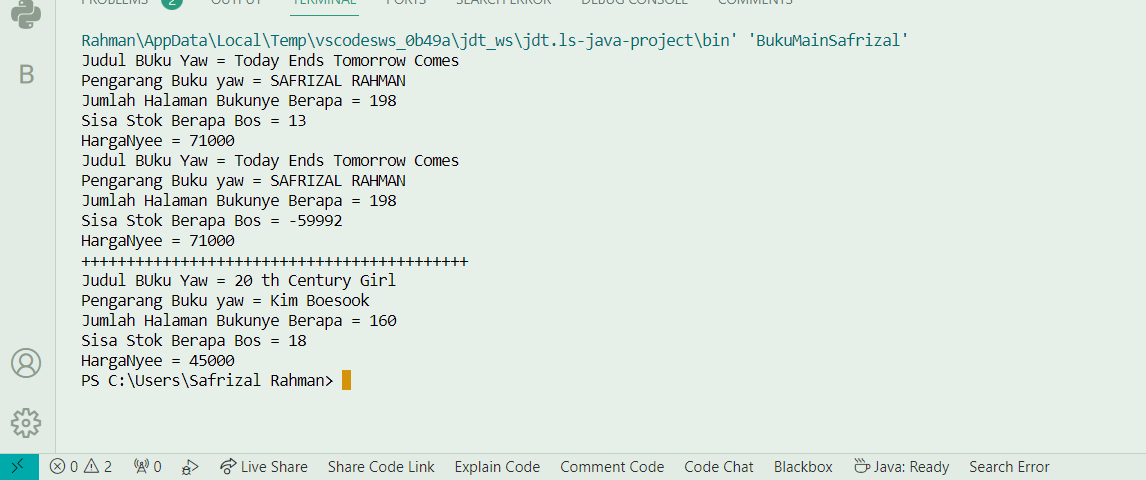
*// // Metode untuk mengubah harga buku*

*// void gantiHarga(int hargaBaru) {*

*//     harga = hargaBaru;*

*//     System.out.println("Harga buku telah diubah menjadi Rp" + harga);*

*// }*



1. baris kode program yang digunakan untuk mendeklarasikan konstruktor berparameter

public BUKUSAFRIZAL(String jdl, String pgg, int hlm, int stok, int hg){

        jdl = jdl;

        pengarang = pgg;

        halaman = hlm;

        this.stok = stok;

        harga = hg;

    }

2. Pada baris program tersebut:

 BukuSafrizal bk2 = new BukuSafrizal("20 th Century Girl", "Kim Boesook", 160, 29,  45000);

Dilakukan pembuatan objek `bk2` dari kelas `BukuSafrizal` menggunakan konstruktor berparameter. Objek `bk2` dibuat dengan inisialisasi atribut-atributnya berdasarkan nilai yang diberikan dalam argumen konstruktor.

Secara lebih rinci, objek `bk2` akan memiliki judul "20 th Century Girl", pengarang "Kim Boesook", jumlah halaman 160, stok 29, dan harga Rp 45.000.

3. Setelah menghapus konstruktor default pada class Buku, saat melakukan kompilasi, akan muncul error. Ini karena tidak ada konstruktor default yang didefinisikan secara eksplisit, dan saat mencoba membuat objek tanpa menggunakan konstruktor, tidak ada konstruktor default yang dapat digunakan.

4. Setelah melakukan instansiasi objek, metode di dalam class BukuSafrizal tidak harus diakses secara berurutan. Urutan pemanggilan metode biasanya ditentukan oleh logika program Anda. Namun, Anda harus memastikan bahwa sebelum memanggil metode pada objek, atribut yang diperlukan oleh metode tersebut sudah diatur

5.

