

Laporan Praktikum Algoritma dan Struktur data

Jobsheet 1 , Object -1

Dosen Pengampuh : Bu.Triana Fatmawati, S.T.,M.T.



Muhammad Afiq Firdaus

2341760189

SIB-1E

JURUSAN TEKNOLOGI INFORMASI

PROGRAM STUDI D-IV SISTEM INFORMASI BISNIS

POLITEKNIK NEGERI MALANG

2024

Percobaan 1, Deklarasi Class, Atribut, dan Method

Berikut merupakan code berdasarkan pada percobaan 1

```
J Buku21.java > Buku21
1 public class Buku21 {
2     String judul, pengarang;
3     int halaman, stok, harga;
4
5     void tampilInformasi() {
6         System.out.println("Judul : " + judul);
7         System.out.println("Pengarang : " + pengarang);
8         System.out.println("Jumlah halaman : " + halaman);
9         System.out.println("Sisa stok : " + stok);
10        System.out.println("Harga : Rp " + harga);
11    }
12    void terjual(int jml) {
13        stok -= jml;
14    }
15    void restock(int jml) {
16        stok += jml;
17    }
18    void gantiHarga(int hrg) {
19        harga = hrg;
20    }
21 }
```

Cannot find a class with the main method.
Source: Debugger for Java

Pertanyaan

1. Sebutkan dua karakteristik class atau object!

Jawab : Dua karakteristik yang ada pada kode pemrograman tersebut adalah Atribut dan Method.

2. Perhatikan class **Buku** pada Praktikum 1 tersebut, ada berapa atribut yang dimiliki oleh class Buku? Sebutkan apa saja atributnya!

Jawab : Ada 5 yaitu :

- Judul
- Pengarang
- Halaman
- Stok
- Harga.

3. Ada berapa method yang dimiliki oleh class tersebut? Sebutkan apa saja methodnya!**Jawab** : Ada 4 yaitu :

- tampilInformasi()
- terjual()
- restock()
- gantiHarga()

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)!

Jawab : Berikut adalah hasil modifikasi method **terjual()**.

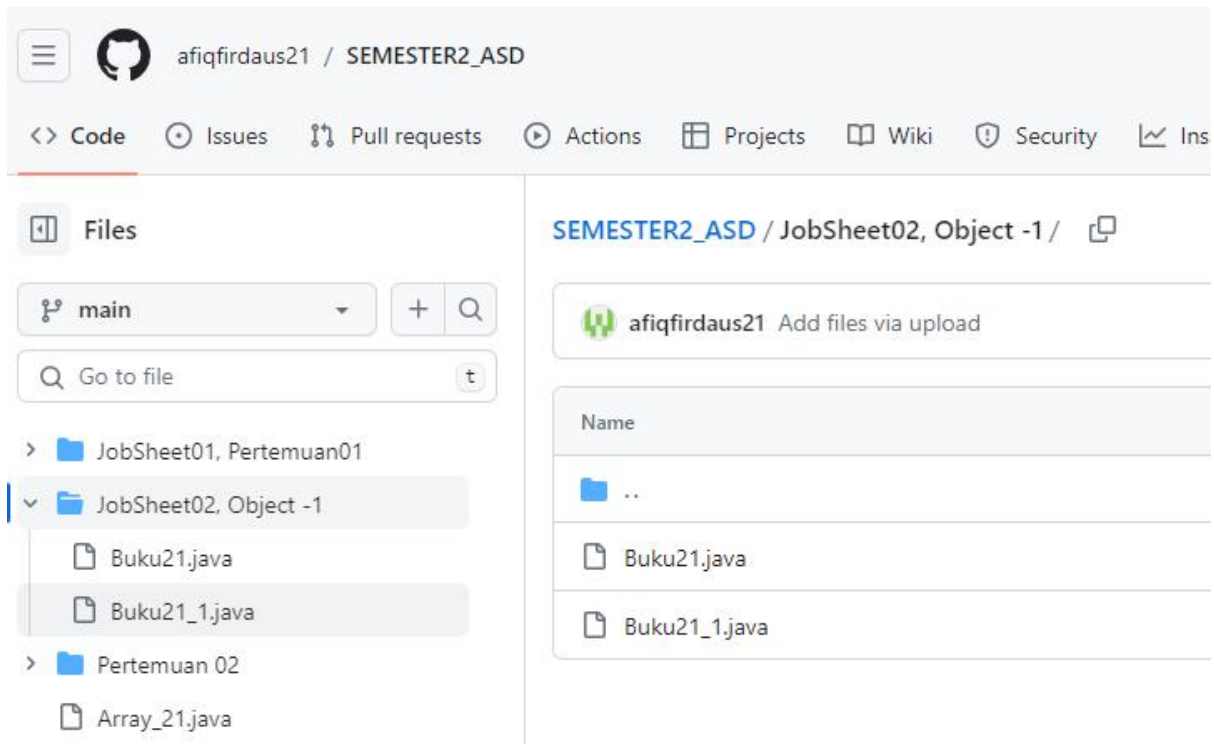
```
1  public class Buku21_1 {
2      String judul, pengarang;
3      int halaman, stok, harga;
4
5      void tampilInformasi() {
6          System.out.println("Judul : " + judul);
7          System.out.println("Pengarang : " + pengarang);
8          System.out.println("Jumlah halaman : " + halaman);
9          System.out.println("Sisa stok : " + stok);
10         System.out.println("Harga : Rp " + harga);
11     }
12     void terjual(int jml) {
13         if (stok > 0) {
14             stok -= jml;
15         } else {
16             System.out.println("Stok habis!");
17         }
18     }
19     void restock(int jml) {
20         stok += jml;
21     }
22     void gantiHarga(int hrg) {
23         harga = hrg;
24     }
25 }
```

5. Menurut Anda, mengapa method **restock()** mempunyai satu parameter berupa bilangan int?

Jawab : Method **restock()** memiliki satu parameter berupa bilangan integer (**int**) karena fungsinya adalah untuk menambahkan jumlah stok buku yang tersedia. Parameter **int** yang diterima oleh method ini menunjukkan jumlah buku yang akan ditambahkan ke stok. Dengan adanya parameter tersebut, pengguna memiliki fleksibilitas untuk menentukan jumlah buku yang akan ditambahkan ke dalam stok setiap kali melakukan proses restock. Ini memungkinkan pengguna untuk mengatur jumlah restock sesuai dengan kebutuhan atau kondisi spesifik dari sistem atau aplikasi yang menggunakan class **Buku16.java**.

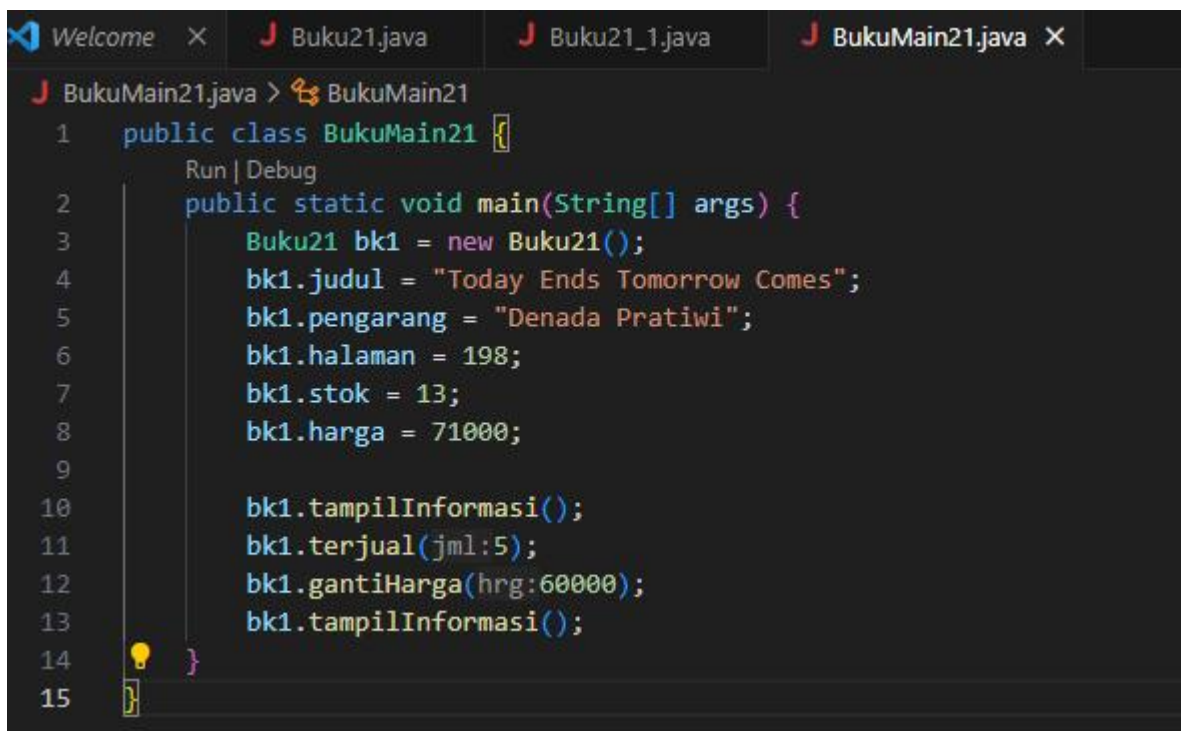
6. Commit dan push kode program ke Github

Jawab :



Percobaan 2 : Instansi object, serta Mengakses Atribut dan Method.

Berikut adalah Hasil code berdasarkan percobaan 2



Berikut adalah hasil running dari code sebelumnya

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Muhammad Afiq Firdaus\Algoritma dan Struktur Data\Praktikum\JobSheet02, Object -1> &
ssages' '-cp' 'C:\Users\HP\AppData\Roaming\Code\User\workspaceStorage\be5ef34b61bb41dbf8e6f
Judul : Today Ends Tomorrow Comes
Pengarang : Denada Pratiwi
Jumlah halaman : 198
Sisa stok : 13
Harga : Rp 71000
Judul : Today Ends Tomorrow Comes
Pengarang : Denada Pratiwi
Jumlah halaman : 198
Sisa stok : 8
Harga : Rp 60000
PS C:\Muhammad Afiq Firdaus\Algoritma dan Struktur Data\Praktikum\JobSheet02, Object -1>
```

Pertanyaan

1. Pada class **BukuMain**, tunjukkan baris kode program yang digunakan untuk proses instansiasi! Apa nama object yang dihasilkan?

Jawab : Nama object yang dihasilkan dari proses instansiasi adalah **bk1**.

```
Buku21 bk1 = new Buku21();
```

2. Bagaimana cara mengakses atribut dan method dari suatu objek?

Jawab : Untuk mengakses atribut dan method dari suatu objek, kita dapat menggunakan notasi titik (.). Notasi titik digunakan untuk mengakses atribut dan method dari objek yang telah dibuat.

Contoh akses atribut : `bk1.judul = "Today Ends Tomorrow Comes";`

Contoh akses method : `bk1.tampilInformasi();`

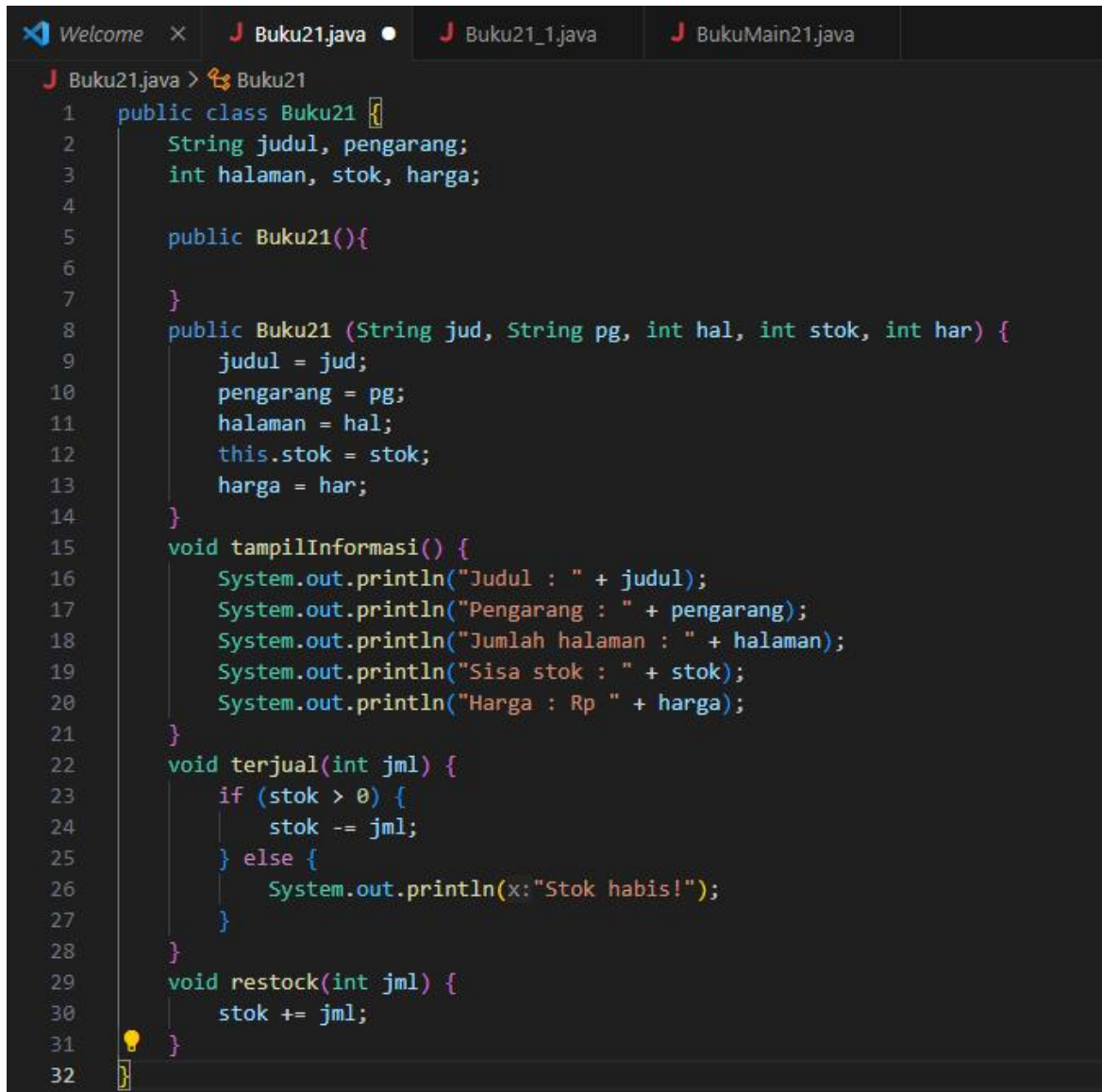
3. Mengapa hasil output pemanggilan method **tampilInformasi()** pertama dan kedua berbeda?

Jawab : Hasil output pemanggilan method **tampilInformasi()** pertama dan kedua berbeda dikarenakan melakukan operasi pengurangan stok dan perubahan harga pada pemanggilan method tersebut.

Pada method **tampilInformasi()** pertama menampilkan informasi buku sebelum melakukan operasi pengurangan stok dan perubahan harga. Sedangkan pada method **tampilInformasi()** kedua, telah melakukan operasi pengurangan stok **5 buku** dan perubahan harga menjadi **60000** sehingga hal itu menjadikan perubahan pada tampilan method yang kedua.

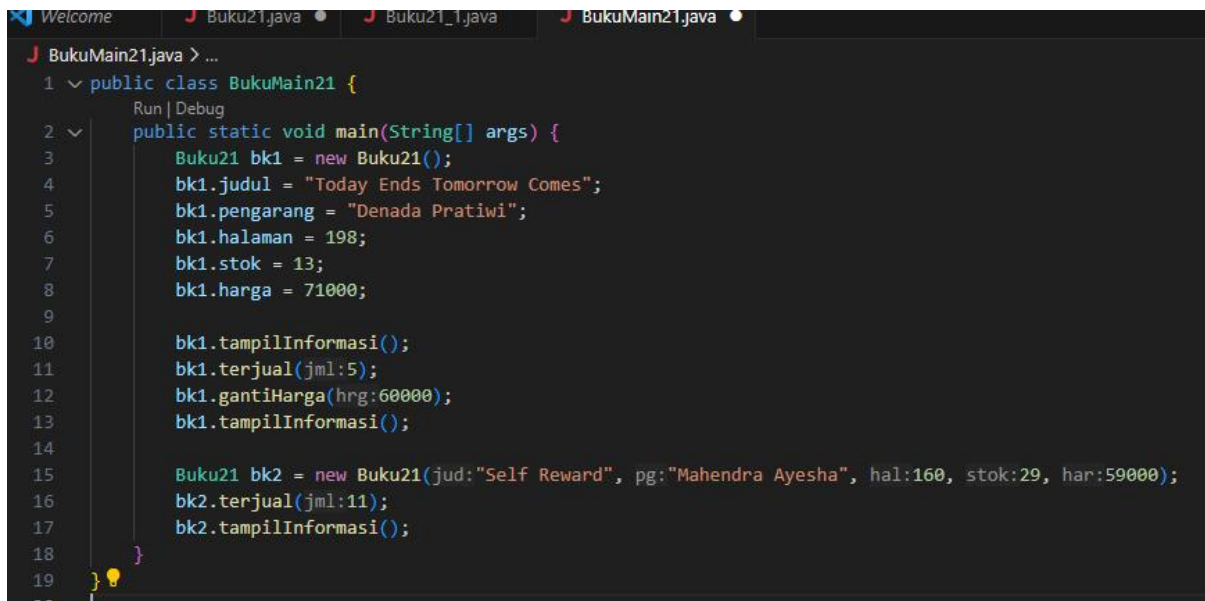
Percobaan 3 : Membuat Kontruktor

Berikut adalah hasil penambahan code pada **Buku21.java**



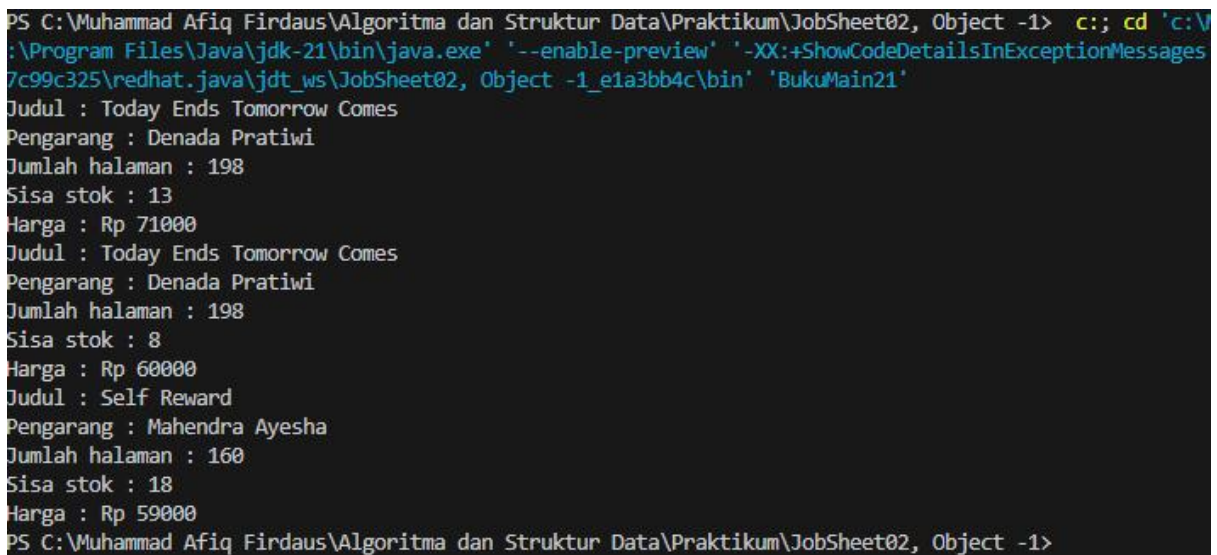
```
1 public class Buku21 {
2     String judul, pengarang;
3     int halaman, stok, harga;
4
5     public Buku21(){
6
7     }
8     public Buku21 (String jud, String pg, int hal, int stok, int har) {
9         judul = jud;
10        pengarang = pg;
11        halaman = hal;
12        this.stok = stok;
13        harga = har;
14    }
15    void tampilInformasi() {
16        System.out.println("Judul : " + judul);
17        System.out.println("Pengarang : " + pengarang);
18        System.out.println("Jumlah halaman : " + halaman);
19        System.out.println("Sisa stok : " + stok);
20        System.out.println("Harga : Rp " + harga);
21    }
22    void terjual(int jml) {
23        if (stok > 0) {
24            stok -= jml;
25        } else {
26            System.out.println(x:"Stok habis!");
27        }
28    }
29    void restock(int jml) {
30        stok += jml;
31    }
32 }
```

Berikut Adalah Hasil Penambahan Code pada **BukuMain21.java**



```
1 public class BukuMain21 {
2     public static void main(String[] args) {
3         Buku21 bk1 = new Buku21();
4         bk1.judul = "Today Ends Tomorrow Comes";
5         bk1.pengarang = "Denada Pratiwi";
6         bk1.halaman = 198;
7         bk1.stok = 13;
8         bk1.harga = 71000;
9
10        bk1.tampilInformasi();
11        bk1.terjual(jml:5);
12        bk1.gantiHarga(hrg:60000);
13        bk1.tampilInformasi();
14
15        Buku21 bk2 = new Buku21(jud:"Self Reward", pg:"Mahendra Ayesha", hal:160, stok:29, har:59000);
16        bk2.terjual(jml:11);
17        bk2.tampilInformasi();
18    }
19 }
20
```

Berikut adalah hasil running dari code diatas



```
PS C:\Muhammad Afiq Firdaus\Algoritma dan Struktur Data\Praktikum\JobSheet02, Object -1> c:; cd 'c:\Program Files\Java\jdk-21\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '7c99c325\redhat.java\jdt_ws\JobSheet02, Object -1_e1a3bb4c\bin' 'BukuMain21'
Judul : Today Ends Tomorrow Comes
Pengarang : Denada Pratiwi
Jumlah halaman : 198
Sisa stok : 13
Harga : Rp 71000
Judul : Today Ends Tomorrow Comes
Pengarang : Denada Pratiwi
Jumlah halaman : 198
Sisa stok : 8
Harga : Rp 60000
Judul : Self Reward
Pengarang : Mahendra Ayesha
Jumlah halaman : 160
Sisa stok : 18
Harga : Rp 59000
PS C:\Muhammad Afiq Firdaus\Algoritma dan Struktur Data\Praktikum\JobSheet02, Object -1>
```

Pertanyaan

1. Pada class **Buku** di Percobaan 3, tunjukkan baris kode program yang digunakan untuk mendeklarasikan konstruktor berparameter!

Jawab : Berikut adalah code program yang diminta

```
public Buku21 (String jud, String pg, int hal, int stok, int har) {  
    judul = jud;  
    pengarang = pg;  
    halaman = hal;  
    this.stok = stok;  
    harga = har;  
}
```

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

Jawab : Pada baris program tersebut, terdapat objek baru bernama **bk2** yang dibuat dari class **Buku21**. Objek ini memiliki beberapa atribut, yaitu judul buku ("Self Reward"), pengarang ("Maheera Ayesha"), jumlah halaman (160), stok (29), dan harga (59000).

3. Hapus konstruktor default pada class **Buku**, kemudian compile dan run program. Bagaimana hasilnya? Jelaskan mengapa hasilnya demikian!

Jawab : Jika menghapus konstruktor default pada class **Buku**, hal yang akan terjadi pada saat run program adalah error. Hal ini dikarenakan pemrograman Java membutuhkan konstruktor untuk membuat objek, dan jika konstruktor default dihapus, maka program tidak dapat dijalankan / error.

4. Setelah melakukan instansiasi object, apakah method di dalam class **Buku** harus diakses secara berurutan? Jelaskan alasannya!

Jawab : Setelah melakukan instansiasi object, method tidak harus diakses secara berurutan. Setiap method dapat dipanggil secara terpisah dan bebas dari urutan pemanggilnya. Hal ini dikarenakan method tersebut adalah method yang independen dan tidak saling terkait, sehingga dapat dipanggil kapan saja sesuai kebutuhan.

5. Buat object baru dengan nama **buku<NamaMahasiswa>** menggunakan konstruktor berparameter dari class **Buku**!

Jawab : Berikut adalah hasil penambahan object baru dengan nama **bukuAfiq** dengan menggunakan konstruktor berparameter.

```
Buku21 bukuAfiq = new Buku21(jud:"Laskar Pelangi", pg:"Andrea Hirata", hal:529, stok:100, har:79200);  
bukuAfiq.terjual(jml:18);  
bukuAfiq.tampilInformasi();
```


Berikut adalah hasil output nya.

```
Judul : Today Ends Tomorrow Comes
Pengarang : Denada Pratiwi
Jumlah halaman : 198
Sisa stok : 13
Harga : Rp 71000
Judul : Today Ends Tomorrow Comes
Pengarang : Denada Pratiwi
Jumlah halaman : 198
Sisa stok : 8
Harga : Rp 60000
Judul : Self Reward
Pengarang : Mahendra Ayesha
Jumlah halaman : 160
Sisa stok : 18
Harga : Rp 59000
Judul : Laskar Pelangi
Pengarang : Andrea Hirata
Jumlah halaman : 529
Sisa stok : 82
Harga : Rp 79200
PS C:\Muhammad Afiq Firdaus\Algoritma dan Struktur Data\Praktikum\JobSheet02, Object -1>
```

6. Commit dan push kode program ke Github

Jawab : Berikut adalah tampilan setelah push di github.

The screenshot displays the GitHub interface for the repository 'afiqfirdaus21 / SEMESTER2_ASD'. The 'Files' tab is active, showing a file tree on the left and a list of files on the right. The file tree includes 'JobSheet01, Pertemuan01', 'JobSheet02, Object -1' (selected), and 'Pertemuan 02'. The file list shows files like 'Buku21.java', 'Buku21_1.java', and 'BukuMain21.java' with their last commit messages.

Name	Last commit message
..	
Buku21.java	Add files via upload
Buku21_1.java	Add files via upload
BukuMain21.java	Add files via upload

Latihan Praktikum

1. Pada class Buku yang telah dibuat, tambahkan tiga method yaitu **hitungHargaTotal()**, **hitungDiskon()**, dan **hitungHargaBayar()** dengan penjelasan sebagai berikut:

- Method **hitungHargaTotal()** digunakan untuk menghitung harga total yang merupakan perkalian antara harga dengan jumlah buku yang terjual
- Method **hitungDiskon()** digunakan untuk menghitung diskon dengan aturan berikut:
 - Jika harga total lebih dari 150000, maka harga didiskon sebesar 12%
 - Jika harga total antara 75000 sampai 150000, maka harga didiskon sebesar 5%
 - Jika harga total kurang dari 75000, maka harga tidak didiskon
- Method **hitungHargaBayar()** digunakan untuk menghitung harga total setelah dikurangi diskon

Class diagram **Buku** setelah penambahan ketiga method tersebut adalah sebagai berikut.

Buku
judul: String pengarang: String halaman: int stok: int harga: int
tampilInformasi(): void terjual(jml: int): void restock(n: int): void gantiHarga(hrg: int): int hitungHargaTotal(): int hitungDiskon(): int hitungHargaBayar(): int

2. Buat program berdasarkan class diagram berikut ini!

Dragon
x: int y: int width: int height: int
moveLeft(): void moveRight(): void moveUp(): void moveDown(): void printPosition(): void detectCollision(x: int, y: int): void

Penjelasan dari atribut dan method pada class Dragon tersebut adalah sebagai berikut:

- Atribut **x** digunakan untuk menyimpan posisi koordinat x (mendatar) dari dragon, sedangkan atribut **y** untuk posisi koordinat y (vertikal)
- Atribut **width** digunakan untuk menyimpan lebar dari area permainan, sedangkan **height** untuk menyimpan panjang area
- Method **moveLeft()** digunakan untuk mengubah posisi dragon ke kiri (koordinat x akan berkurang 1), sedangkan **moveRight()** untuk bergerak ke kanan (koordinat x akan bertambah 1). Perlu diperhatikan bahwa koordinat x tidak boleh lebih kecil dari 0 atau lebih

besar dari nilai width. Jika koordinat $x < 0$ atau $x > \text{width}$ maka panggil method

detectCollision()

- Method **moveUp()** digunakan untuk mengubah posisi dragon ke atas (koordinat y akan berkurang 1), sedangkan **moveDown()** untuk bergerak ke bawah (koordinat y akan bertambah

1). Perlu diperhatikan bahwa koordinat y tidak boleh lebih kecil dari 0 atau lebih besar dari nilai height. Jika koordinat $y < 0$ atau $y > \text{height}$ maka panggil method

detectCollision()

- Method **detectCollision()** akan mencetak pesan “Game Over” apabila dragon menyentuh ujung area permainan.

Jawaban :

1. Berikut adalah hasil penambahan code method pada **Buku21.java**

```
int hitungHargaTotal(int jumlahTerjual) {  
    return harga * jumlahTerjual;  
}  
  
int hitungDiskon(int hargaTotal) {  
    if (hargaTotal > 150000) {  
        return hargaTotal * 12 / 100;  
    } else if (hargaTotal >= 75000 && hargaTotal <= 150000) {  
        return hargaTotal * 5 / 100;  
    } else {  
        return 0;  
    }  
}  
  
int hitungHargaBayar(int hargaTotal, int diskon) {  
    return hargaTotal - diskon;  
}
```

Berikut adalah hasil penambahan code pada **BukuMain21.java**.

```
int jumlahTerjual = 5;
int hargaTotal = bukuAfiq.hitungHargaTotal(jumlahTerjual);
int diskon = bukuAfiq.hitungDiskon(hargaTotal);
int hargaBayar = bukuAfiq.hitungHargaBayar(hargaTotal, diskon);

System.out.println("Jumlah terjual : " + jumlahTerjual);
System.out.println("Harga total: Rp " + hargaTotal);
System.out.println("Diskon: Rp " + diskon);
System.out.println("Harga bayar: Rp " + hargaBayar);|
```

Berikut adalah contoh dari hasil output kode tersebut.

```
Judul : Laskar Pelangi
Pengarang : Andrea Hirata
Jumlah halaman : 529
Sisa stok : 82
Harga : Rp 79200
Jumlah terjual : 5
Harga total: Rp 396000
Diskon: Rp 47520
Harga bayar: Rp 348480
```

2. Berikut adalah tampilan code pemrograman java **Dragon21.java**.

```
Welcome x Buku21.java Buku21_1.java BukuMain21.java Dragon21.java
Dragon21.java > Dragon21 > moveLeft()
1 public class Dragon21 {
2     int x, y, width, height;
3
4     void moveLeft(){
5         x -= 1;
6         System.out.println("x: "+x+" y: "+y);
7         detectCollision(x,y);
8     }
9     void moveRight(){
10        x += 1;
11        System.out.println("x: "+x+" y: "+y);
12        detectCollision(x,y);
13    }
14    void moveUp(){
15        y -= 1;
16        System.out.println("x: "+x+" y: "+y);
17        detectCollision(x,y);
18    }
19    void moveDown(){
20        y += 1;
21        System.out.println("x: "+x+" y: "+y);
22        detectCollision(x,y);
23    }
24    void detectCollision(int x, int y){
25        if(x <= 0 || x > width || y <= 0 || y > height){
26            System.out.println(x:"Game Over");
27        }
28    }
29 }
```

Berikut adalah tampilan code pemrograman java **DragonMain21.java**.

```

Welcome
J Buku21.java
J Buku21_1.java
J BukuMain2
J DragonMain21.java > DragonMain21
1  import java.util.Scanner;
2  public class DragonMain21 {
    Run | Debug
3      public static void main(String[] args) {
4          Dragon21 drg21 = new Dragon21();
5          drg21.x = 3;
6          drg21.y = 2;
7          drg21.width = 5;
8          drg21.height = 5;
9          drg21.moveDown();
10         drg21.moveRight();
11         drg21.moveLeft();
12         drg21.moveUp();
13         drg21.moveUp();
14         drg21.moveUp();
15     }
16 }

```

Berikut adalah hasil running dari code diatas

```
PS C:\Muhammad Afiq Firdaus\Algoritma dan Struktur Data\Praktikum\JobSheet02, Object -1> & 'C:\Users\HP\AppData\Roaming\Code\User\workspaceStorage\be5ef341n\DragonMain21'
x: 3 y: 3
x: 4 y: 3
x: 3 y: 3
x: 3 y: 2
x: 3 y: 1
x: 3 y: 0
Game Over
PS C:\Muhammad Afiq Firdaus\Algoritma dan Struktur Data\Praktikum\JobSheet02, Object -1>
```