

ALGORITMA DAN STRUKTUR DATA

Jobsheet 2 Object - Tugas

Lavina 2341760062

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

Buku14.java

```
1      int halaman, stok, harga, totalHarga;
2
3      void hitungHargaTotal(int jml) {
4          totalHarga = harga * jml;
5      }
6
7      void tampilTransaksi() {
8          System.out.println("Total harga: " + totalHarga);
9      }
10
```

BukuMain14.java

```
1  Buku14 bukuLavina = new Buku14("Harry Potter", "JK Rowling", 250, 20, 99_000);
2  bukuLavina.hitungHargaTotal(2);
3  bukuLavina.tampilTransaksi();
```

Output

```
review' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp'
_14a4c95\bin' 'BukuMain14'
Total harga: 198000
PS D:\Coolyeah>
```

- 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

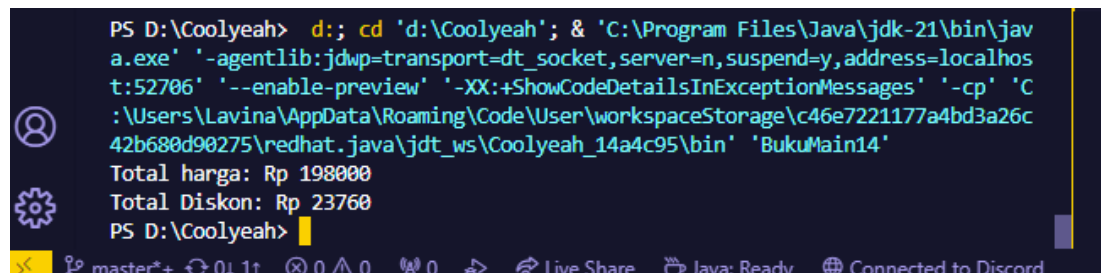
Buku14.java

```
1    double diskon;  
2  
3    void hitungDiskon() {  
4        if (totalHarga > 150_000) {  
5            diskon = totalHarga * 0.12;  
6        } else if (totalHarga > 75_000 && totalHarga < 150_000) {  
7            diskon = totalHarga * 0.05;  
8        } else {  
9            diskon = 0;  
10       }  
11   }  
12  
13   void tampilTransaksi() {  
14       System.out.println("Total harga: Rp " + totalHarga);  
15       System.out.println("Total Diskon: Rp " + (int) diskon);  
16   }  
17
```

BukuMain14.java

```
Buku14 bukuLavina = new Buku14("Harry Potter", "JK Rowling", 250, 20, 99_000);  
bukuLavina.hitungHargaTotal(2);  
bukuLavina.hitungDiskon();  
bukuLavina.tampilTransaksi();
```

Output :



```
PS D:\Coolyeah> d:; cd 'd:\Coolyeah'; & 'C:\Program Files\Java\jdk-21\bin\jav  
a.exe' '-agentlib:jwp=transport=dt_socket,server=n,suspend=y,address=localhos  
t:52706' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C  
:\Users\Lavina\AppData\Roaming\Code\User\workspaceStorage\c46e7221177a4bd3a26c  
42b680d90275\redhat.java\jdt_ws\Coolyeah_14a4c95\bin' 'BukuMain14'  
Total harga: Rp 198000  
Total Diskon: Rp 23760  
PS D:\Coolyeah>
```

- Method **hitungHargaBayar()** digunakan untuk menghitung harga total setelah dikurangi diskon.

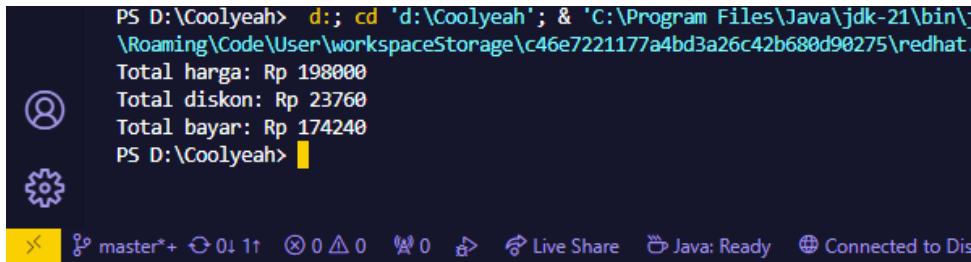
Buku14.java

```
1    int totalBayar;  
2  
3    void hitungHargaBayar() {  
4        totalBayar = totalHarga - (int) diskon;  
5    }  
6  
7    void tampilTransaksi() {  
8        System.out.println("Total harga: Rp " + totalHarga);  
9        System.out.println("Total diskon: Rp " + (int) diskon);  
10       System.out.println("Total bayar: Rp " + totalBayar);  
11   }  
12
```

BukuMain14.java

```
Buku14 bukuLavina = new Buku14("Harry Potter", "JK Rowling", 250, 20, 99_000);
bukuLavina.hitungHargaTotal(2);
bukuLavina.hitungDiskon();
bukuLavina.hitungHargaBayar();
bukuLavina.tampilTransaksi();
```

Output:



```
PS D:\Coolyeah> d:; cd 'd:\Coolyeah'; & 'C:\Program Files\Java\jdk-21\bin\
\Roaming\Code\User\workspaceStorage\c46e7221177a4bd3a26c42b680d90275\redhat
Total harga: Rp 198000
Total diskon: Rp 23760
Total bayar: Rp 174240
PS D:\Coolyeah>
```

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 > 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 > height maka panggil method **detectCollision()**
- Method **detectCollision()** akan mencetak pesan "Game Over" apabila dragon menyentuh ujung area permainan.

Dragon14.java

```
1  public class Dragon14 {
2      int x, y, width, height;
3
4      public Dragon14(int w, int h) {
5          width = w;
6          height = h;
7      }
8
9      void printPosition() {
10         System.out.println("Dragon Current Position");
11         System.out.printf("x: %d\ny: %d\n", x, y);
12         System.out.println();
13     }
14
15     void detectCollision(int x, int y) {
16         if (x < 0 || x > width || y < 0 || y > height) {
17             System.out.println("Game Over !");
18             System.exit(0);
19         }
20     }
21
22     void moveRight() {
23         x++;
24         detectCollision(x, y);
25     }
```

```

26
27     void moveLeft() {
28         x--;
29         detectCollision(x, y);
30     }
31
32     void moveUp() {
33         y++;
34         detectCollision(x, y);
35     }
36
37     void moveBottom() {
38         y--;
39         detectCollision(x, y);
40     }
41
42 }
43

```

DragonMain14.java

```

1  import java.util.Scanner;
2
3  public class DragonMain14 {
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6          Dragon14 dragon = new Dragon14(3, 5);
7
8          while (true) {
9              System.out.println("Move Dragon");
10             System.out.println("1. Move Right\n2. Move Left\n3. Move Up\n4. Move Bottom");
11             System.out.print("Enter: ");
12             int move = sc.nextInt();
13
14             System.out.println();
15
16             switch (move) {
17                 case 1:
18                     dragon.moveRight();
19                     break;
20                 case 2:
21                     dragon.moveLeft();
22                     break;
23                 case 3:
24                     dragon.moveUp();
25                     break;
26                 case 4:
27                     dragon.moveBottom();
28                     break;
29                 default:
30                     break;
31             }
32             dragon.printPosition();
33         }
34     }
35 }
36

```

Output:

```
Move Dragon
1. Move Right
2. Move Left
3. Move Up
4. Move Bottom
Enter: 3

Dragon Current Position
x: 2
y: 1

Move Dragon
1. Move Right
2. Move Left
3. Move Up
4. Move Bottom
Enter: 1

Dragon Current Position
x: 3
y: 1

Move Dragon
1. Move Right
2. Move Left
3. Move Up
4. Move Bottom
Enter: 1

Game Over !
PS D:\Coolyeah>
```

Link Github: [Coolyeah/SEMESTER2/Algoritma Struktur Data/praktikum/P2 Object/code at master · Lavina-23/Coolyeah \(github.com\)](https://github.com/Lavina-23/Coolyeah)