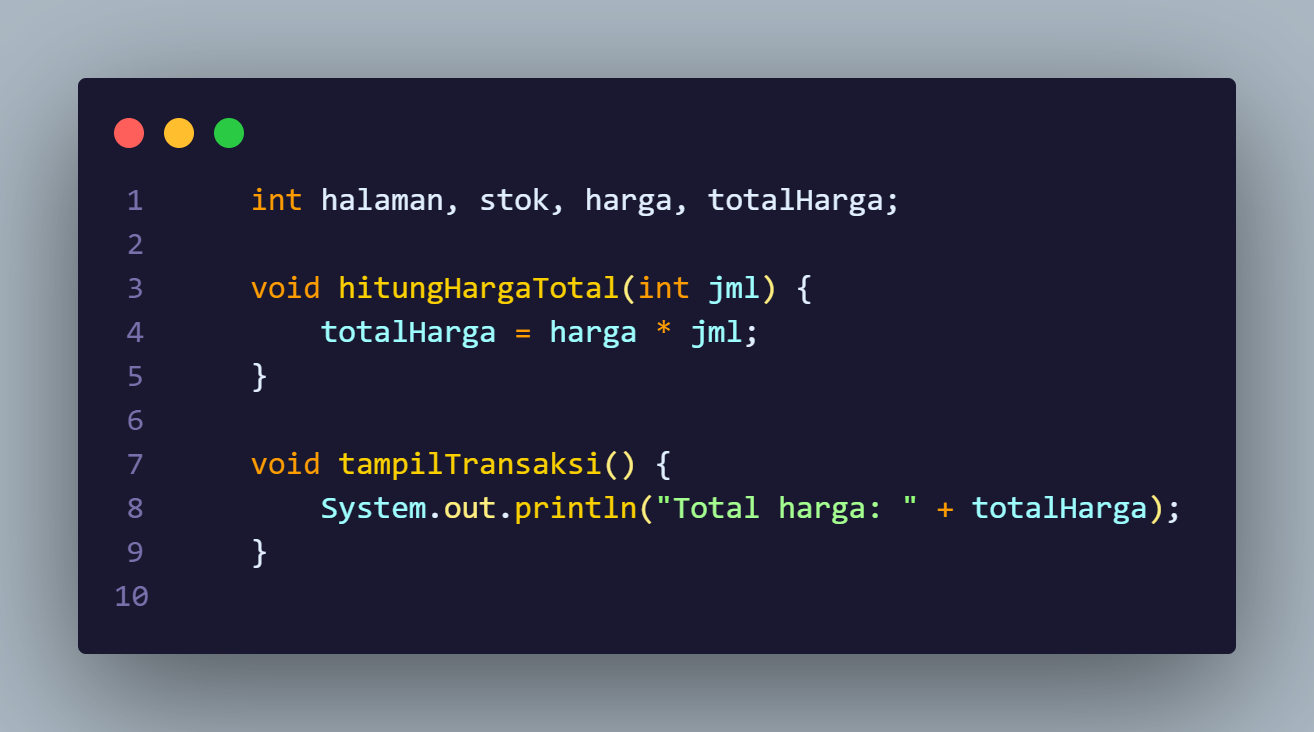
**ALGORITMA DAN STRUKTUR DATA**

**Jobsheet 2 Object - Tugas**

**Lavina 2341760062**

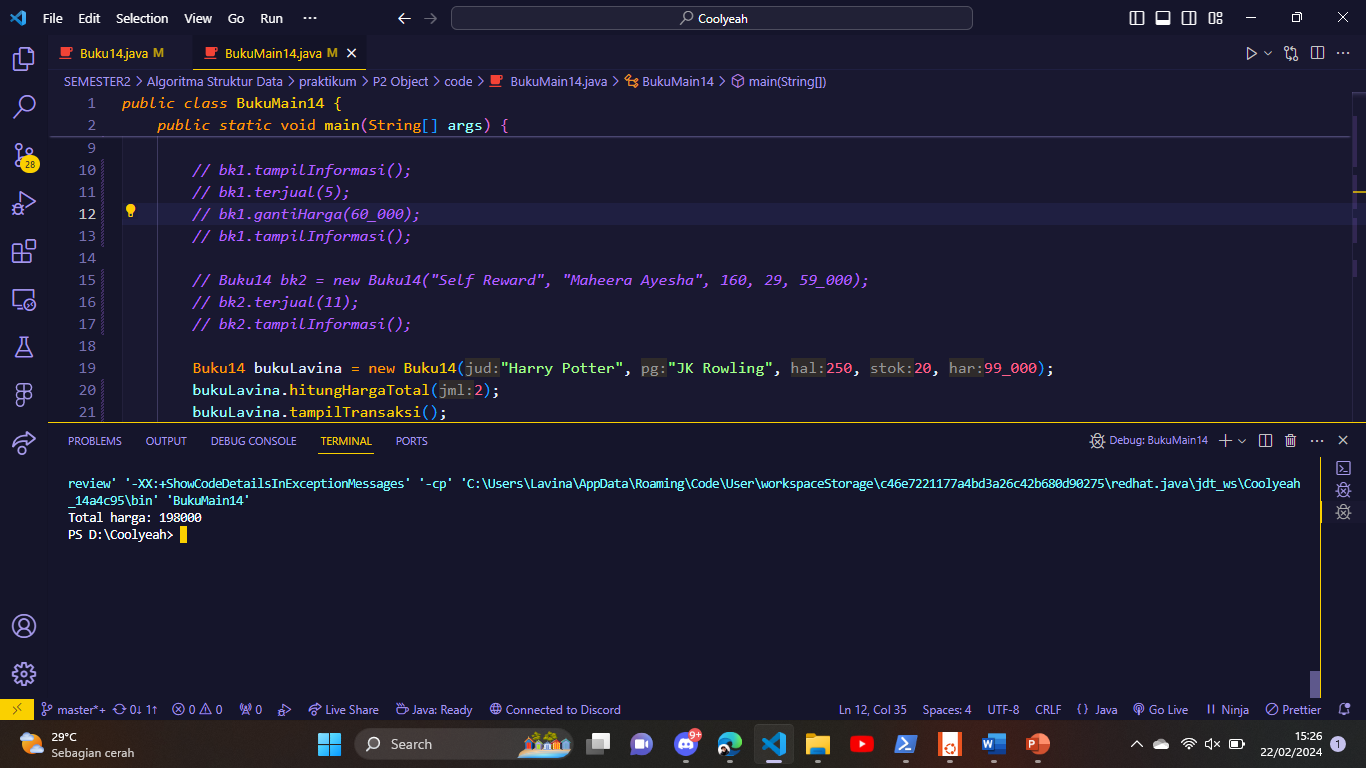
1. Pada class Buku yang telah dibuat, tambahkan tiga method yaitu **hitungHargaTotal()**, **hitungDiskon()**, dan **hitungHargaBayar()** dengan penjelasan sebagai berikut:
   * Method **hitungHargaTotal()** digunakan untuk menghitung harga total yang merupakan perkalian antara harga dengan jumlah buku yang terjual

Buku14.java



BukuMain14.java

Output



* + 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

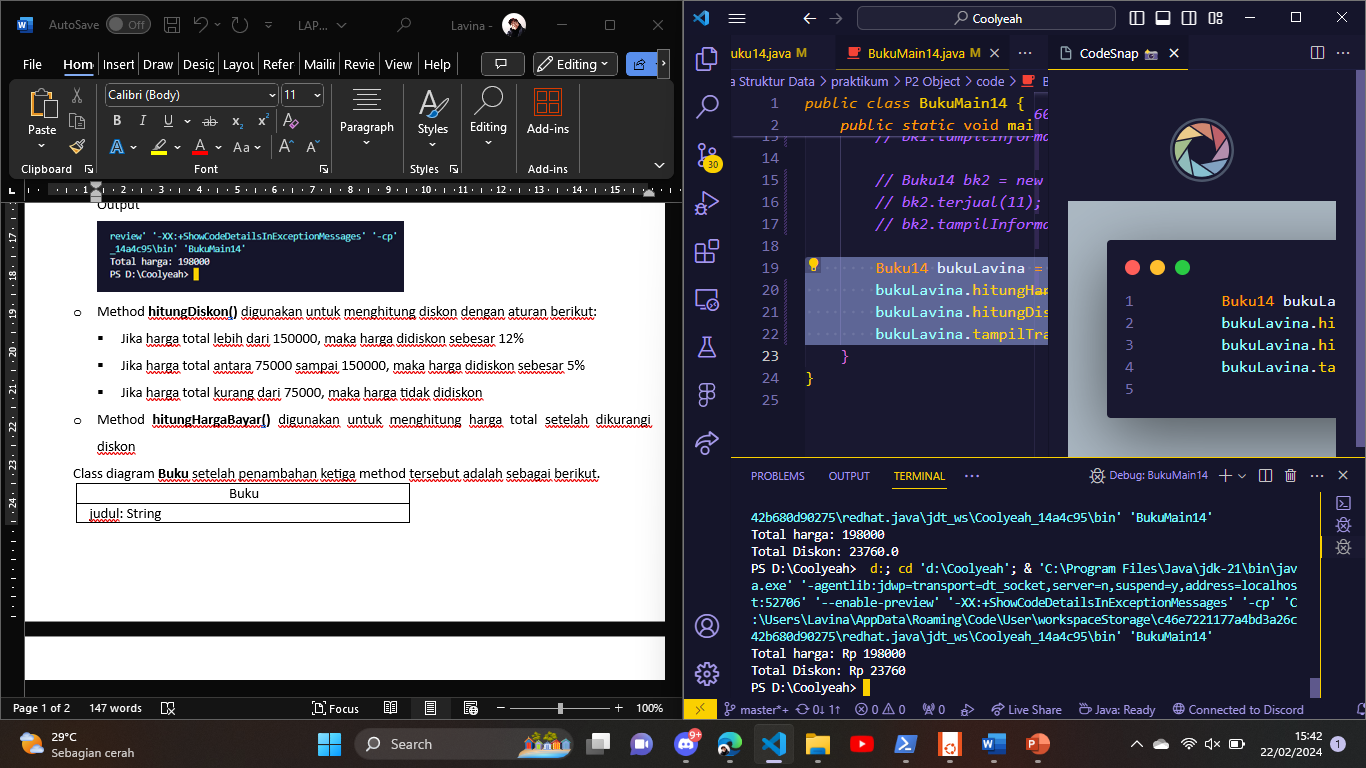
Buku14.java



BukuMain14.java

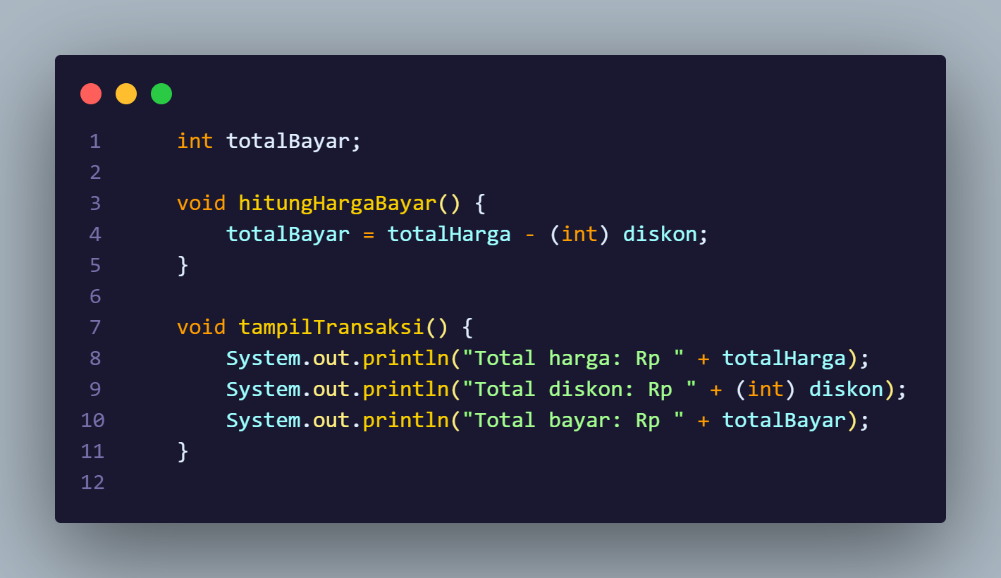


Output :



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

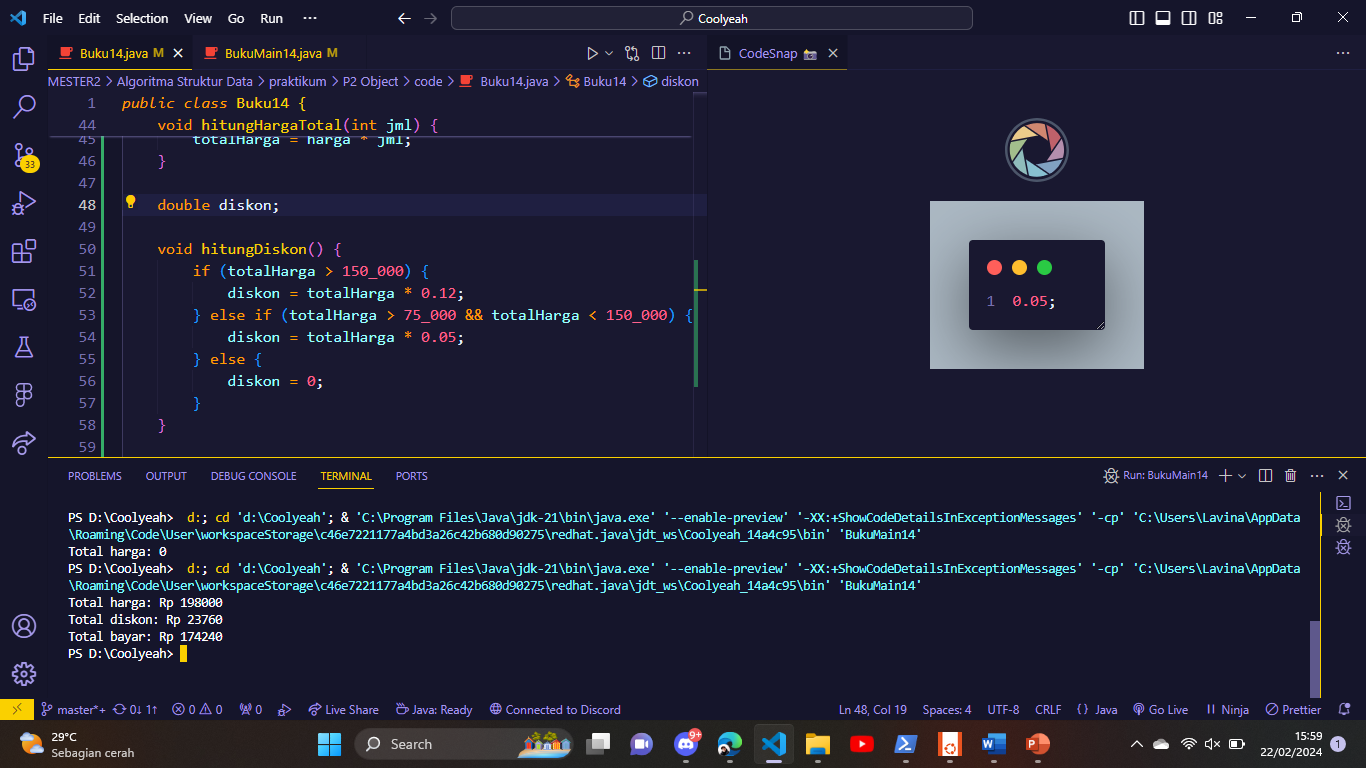
Buku14.java



BukuMain14.java



Output:



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** |

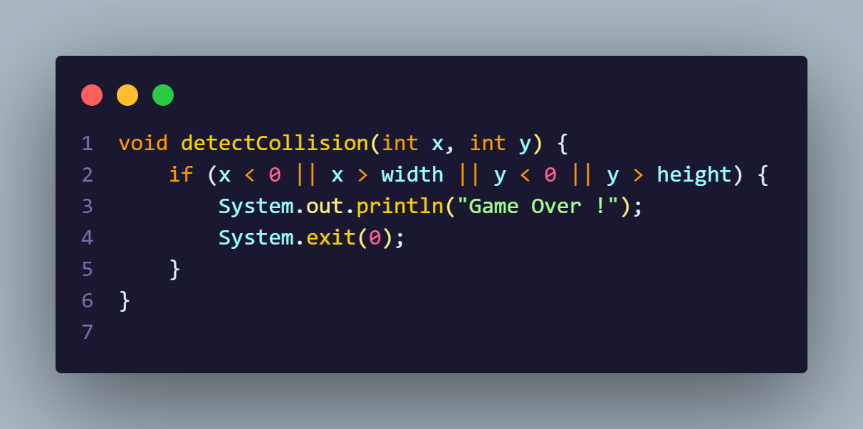
1. Buat program berdasarkan class diagram berikut ini!

|  |
| --- |
| 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

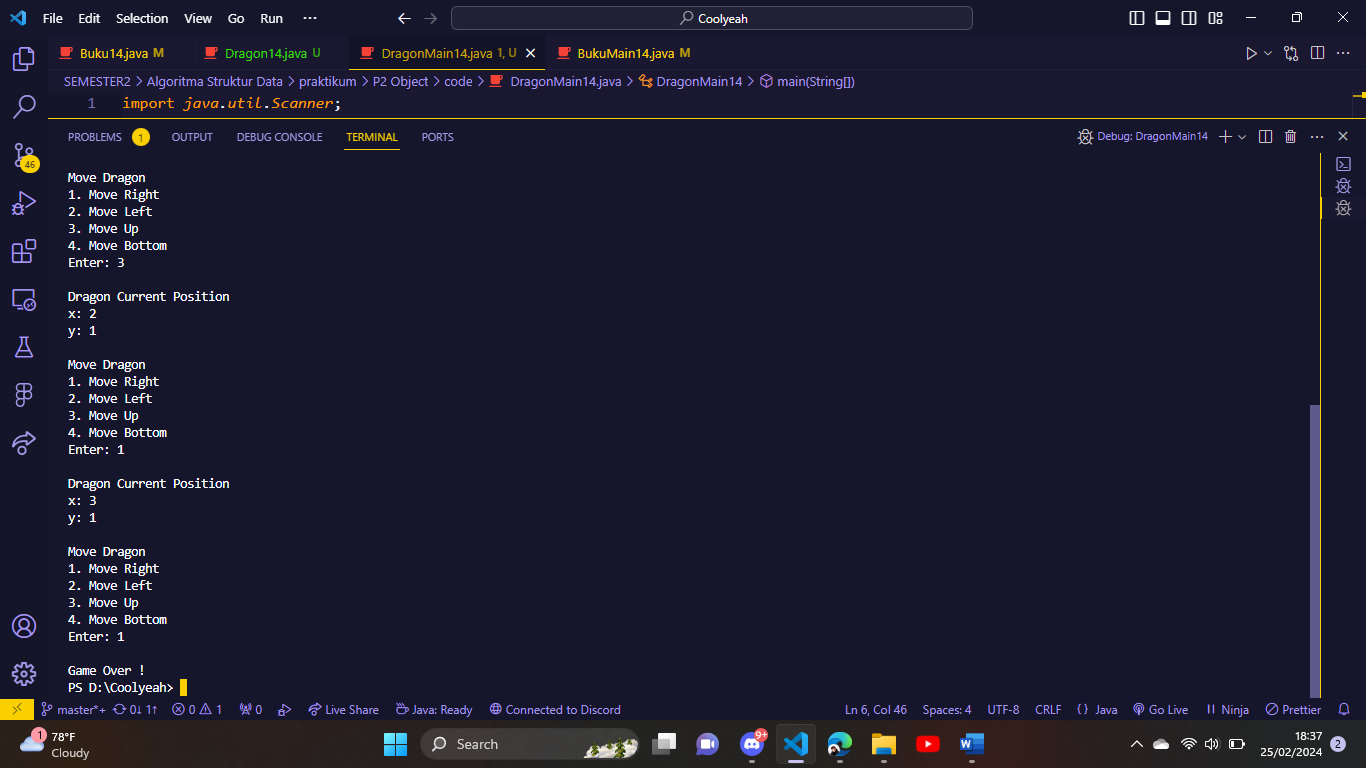
****

****

DragonMain14.java



Output:



**Link Github:** [Coolyeah/SEMESTER2/Algoritma Struktur Data/praktikum/P2 Object/code at master · Lavina-23/Coolyeah (github.com)](https://github.com/Lavina-23/Coolyeah/tree/master/SEMESTER2/Algoritma%20Struktur%20Data/praktikum/P2%20Object/code)