

# Praktikum Algoritma dan Struktur Data

## Jobsheet 2



Politeknik Negeri Malang  
Semester Ganjil  
2023

**NIM**  
2341720105

**Nama**  
Achmad Anfasa Rabbany

**Kelas**  
1E

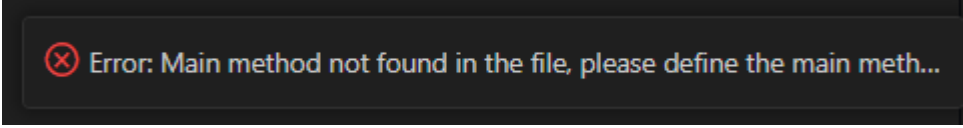
**Jurusan**  
Teknologi Informasi

**Program Studi**  
D-IV Teknik Informatika

# 1. Praktikum

## 2.1 Percobaan 1: Deklarasi Class, Atribut dan Method

### 1. Verifikasi



✖ Error: Main method not found in the file, please define the main meth...

## Pertanyaan

1.

Q: Sebutkan dua karakteristik class atau object!

A: Memiliki Atribut dan method

2.

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

A: Ada 5 atribut, judul, pengarang, halaman, stok, dan harga.

3.

Q: Ada berapa method yang dimiliki oleh class tersebut? Sebutkan apa saja methodnya!

A: Ada 4 method, tampilInformasi(), terjual(), restock(), dan gantiHarga().

4.

Q: 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)

A:

```
void terjual(int jml){  
    if (stok > 0){  
        stok -= jml;  
    }else {  
        System.out.println(x: "Stok habis!");  
    }  
}
```

5.

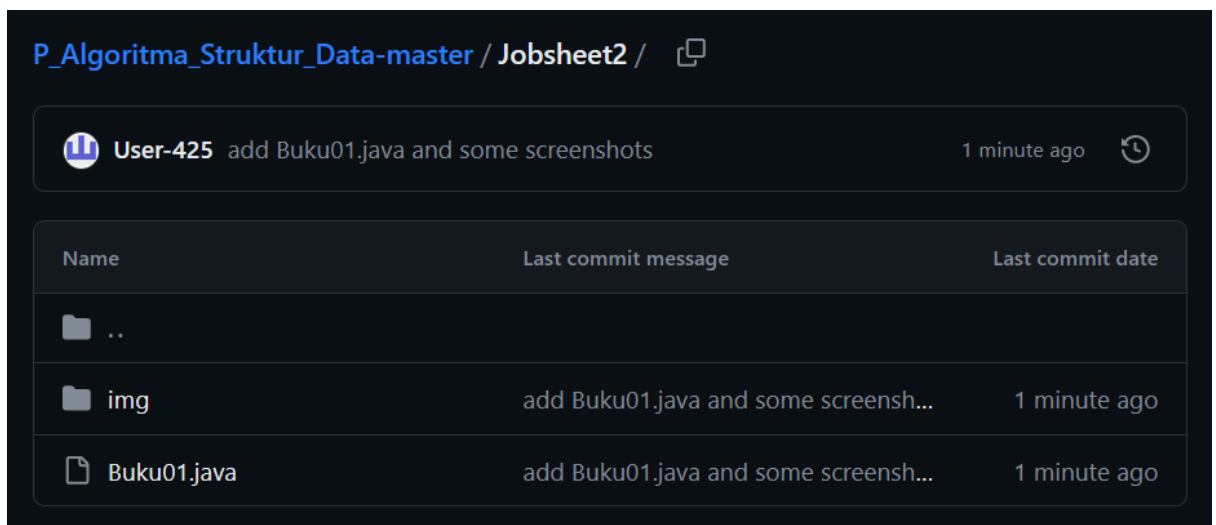
Q: Menurut Anda, mengapa method restock() mempunyai satu parameter berupa bilangan int?

A: Karena restock memerlukan jumlah barang yang direstock agar dapat menambah stok sesuai dengan jumlah barang yang direstock

6.

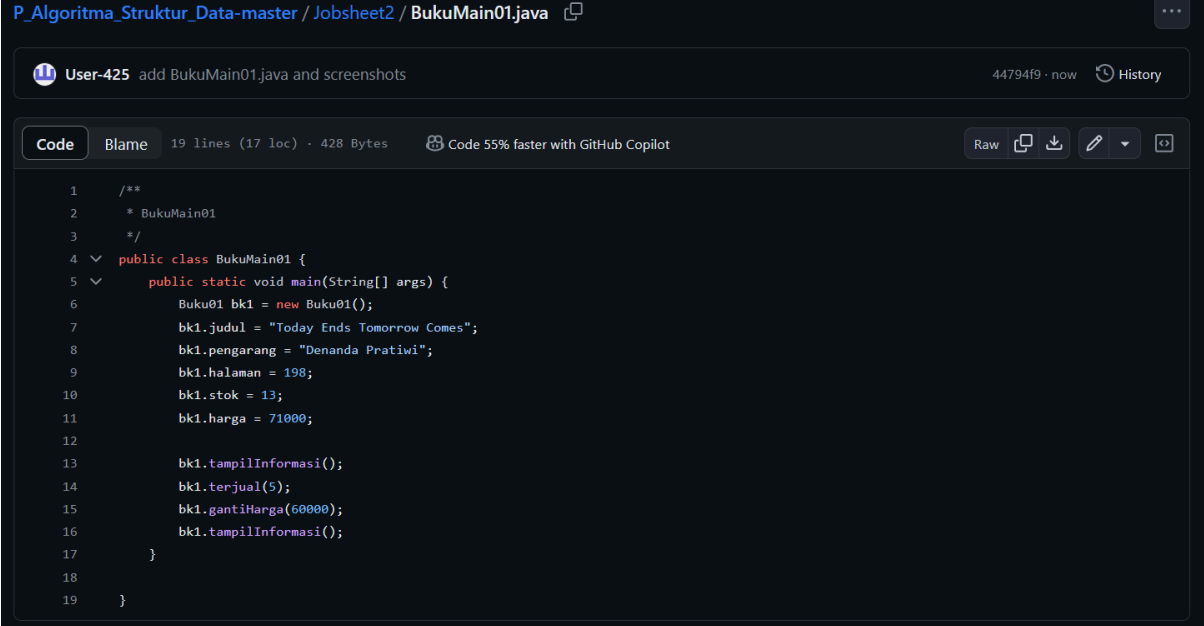
Q: Commit dan push kode program ke Github

A:



## 2.2 Percobaan 2 : Instansiasi Object, serta Mengakses Atribut dan Method

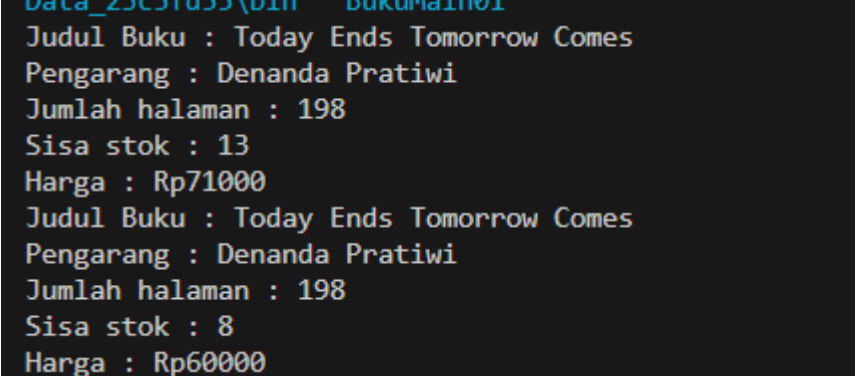
### 1. Commit dan Push ke Github



The screenshot shows a GitHub repository named 'P\_Algoritma\_Struktur\_Data-master' with a file 'Jobsheet2 / BukuMain01.java'. The commit message is 'User-425 add BukuMain01.java and screenshots'. The code is as follows:

```
1  /**
2   * BukuMain01
3   */
4  public class BukuMain01 {
5      public static void main(String[] args) {
6          Buku01 bk1 = new Buku01();
7          bk1.judul = "Today Ends Tomorrow Comes";
8          bk1.pengarang = "Denanda Pratiwi";
9          bk1.halaman = 198;
10         bk1.stok = 13;
11         bk1.harga = 71000;
12
13         bk1.tampilInformasi();
14         bk1.terjual(5);
15         bk1.gantiHarga(60000);
16         bk1.tampilInformasi();
17     }
18 }
19 }
```

### 2. Verifikasi



The screenshot shows the output of the program in a terminal window. The output is as follows:

```
data_23031033\bin - BukuMain01
Judul Buku : Today Ends Tomorrow Comes
Pengarang : Denanda Pratiwi
Jumlah halaman : 198
Sisa stok : 13
Harga : Rp71000
Judul Buku : Today Ends Tomorrow Comes
Pengarang : Denanda Pratiwi
Jumlah halaman : 198
Sisa stok : 8
Harga : Rp60000
```

## Pertanyaan:

1.

Q: Pada class BukuMain, tunjukkan baris kode program yang digunakan untuk proses instansiasi! Apa nama object yang dihasilkan?

A: nama dari object yang dihasilkan adalah bk1

```
Buku01 bk1 = new Buku01();
```

2.

Q: Bagaimana cara mengakses atribut dan method dari suatu objek?

A: Kita dapat mengakses atribut ataupun method menggunakan format namaObject.namaAtribut atau namaObject.namaMethod(parameter)

Contoh : bk1.pengarang

3.

Q: Mengapa hasil output pemanggilan method tampilInformasi() pertama dan kedua berbeda?

A: Karena setelah tampilInformasi() yang pertama, dilakukan perubahan atribut dengan menggunakan method terjual() dan gantiHarga() sehingga ketika tampilIndormasi dipanggil lagi, terjadi perbedaan nilai

## 2.3 Percobaan 3 : Membuat Konstruktor

1. Membuat dua buah konstruktor, default dan berparameter. Dan juga membuat sebuah object dengan nama bk2 menggunakan konstruktor berparameter

```
public Buku01(){  
  
}  
  
public Buku01(String jud, String pg, int hal, int stok, int har){  
    judul = jud;  
    pengarang = pg;  
    halaman = hal;  
    this.stok = stok;  
    harga = har;  
}
```

```
Buku01 bk2 = new Buku01(jud:"Self Reward", pg:"Maheera Ayesha", hal:160, stok:29, har:59000);  
bk2.terjual(jml:11);  
bk2.tampilInformasi();
```

2. Verifikasi

```
Judul Buku : Today Ends Tomorrow Comes  
Pengarang : Denanda Pratiwi  
Jumlah halaman : 198  
Sisa stok : 13  
Harga : Rp71000  
Judul Buku : Today Ends Tomorrow Comes  
Pengarang : Denanda Pratiwi  
Jumlah halaman : 198  
Sisa stok : 8  
Harga : Rp60000  
Judul Buku : Self Reward  
Pengarang : Maheera Ayesha  
Jumlah halaman : 160  
Sisa stok : 18  
Harga : Rp59000
```

## Pertanyaan:

1.

Q: Pada class Buku di Percobaan 3, tunjukkan baris kode program yang digunakan untuk mendeklarasikan konstruktor berparameter!

A:

```
public Buku01(String jud, String pg, int hal, int stok, int har){  
    judul = jud;  
    pengarang = pg;  
    halaman = hal;  
    this.stok = stok;  
    harga = har;  
}
```

2.

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

```
Buku bk2 = new Buku(jud:"Self Reward", pg:"Maheera Ayesha", hal:160, stok:29, har:59000);
```

A: Membuat object baru bernama bk2 dengan class Buku menggunakan konstruktor berparameter. Sehingga bk2 telah memiliki nilai judul, pengarang, halaman, stok, dan harga.

3.

Q: Hapus konstruktor default pada class Buku, kemudian compile dan run program.

Bagaimana hasilnya? Jelaskan mengapa hasilnya demikian!

A: Terjadi error karena pada ada baris program yang menggunakan konstruktor default

```
Buku01 bk1 = new Buku01();
```

4.

Q: Setelah melakukan instansiasi object, apakah method di dalam class Buku harus diakses secara berurutan? Jelaskan alasannya!

A: Tidak, karena method itu seperti fungsi, bisa dipanggil kapan saja, dan juga bahkan bisa tidak dipanggil sama sekali.

5.

Q: Buat object baru dengan nama buku<NamaMahasiswa> menggunakan konstruktor berparameter dari class Buku!

A:

```
Buku01 bukuAchmadAnfasaRabbany = new Buku01(jud:"Another Day, Another Journey", pg:"Achmad Anfasa Rabbany", hal:143, stok:21, har:62500);
```

6.

Q: Commit dan push kode program ke Github

A:

P\_Algoritma\_Struktur\_Data-master / Jobsheet2 /



User-425 finish

1 minute ago



Name	Last commit message	Last commit date
..		
img	finish	1 minute ago
pdf	finish	1 minute ago
Buku01.java	finish	1 minute ago
BukuMain01.java	finish	1 minute ago



## 2.4 Latihan Praktikum

1. Membuat `hitungHargaTotal()`, `hitungDiskon()`, dan `hitungHargaBayar()`

```
Codeium: Refactor | Explain | Generate Javadoc | ✕  
int hitungHargaTotal(int jml){  
    return harga * jml;  
}  
  
Codeium: Refactor | Explain | Generate Javadoc | ✕  
int hitungDiskon(int jml){  
    int diskon;  
    if (hitungHargaTotal(jml) > 150000) {  
        diskon = 12;  
    }else if(hitungHargaTotal(jml) <= 150000 || hitungHargaTotal(jml) > 75000){  
        diskon = 5;  
    }else{  
        diskon = 0;  
    }  
    return diskon;  
}  
  
Codeium: Refactor | Explain | Generate Javadoc | ✕  
int hitungHargaBayar(int jml){  
    return (hitungHargaTotal(jml) - (hitungHargaTotal(jml)*hitungDiskon(jml)/100));  
}
```

2. Membuat Dragon game dengan function:
  - a. `moveLeft()`
  - b. `moveRight()`
  - c. `moveUp()`
  - d. `moveDown()`

```
Codeium: Refactor | Explain | Generate Javadoc | ✕
void moveLeft() {
    x--;
}
```

```
Codeium: Refactor | Explain | Generate Javadoc | ✕
void moveRight() {
    x++;
}
```

```
Codeium: Refactor | Explain | Generate Javadoc | ✕
void moveDown() {
    y--;
}
```

```
Codeium: Refactor | Explain | Generate Javadoc | ✕
void moveUp() {
    y++;
}
```

e. printPosition()

```
public static void printPosition() {
    System.out.println("-".repeat((width + 2) * 2));
    for (int i = 0; i < height; i++) {
        String leftSide = " ".repeat(x * 2);
        String rightSide = " ".repeat((width - x) * 2);
        if (height - 1 - y == i) {
            System.out.println("|" + leftSide + "++" + rightSide + "|");
        } else {
            System.out.println("|" + leftSide + "  " + rightSide + "|");
        }
    }
    System.out.println("-".repeat((width + 2) * 2));
    System.out.println("X = " + x + ", Y = " + y);
}
```

f. detectCollision()

```
Codeium: Refactor | Explain | Generate Javadoc | ✕
void detectCollision(int x, int y) {
    if (x < 0 || y < 0 || x > width || y > height) {
        System.out.println(x: "Game over!");
        running = false;
    }
}
```

g. main()

```
public static void main(String[] args) {  
    Scanner input = new Scanner(System.in);  
    running = true;  
    while (running) {  
        printPosition();  
        String command = input.nextLine();  
        if (command.equalsIgnoreCase("quit")) {  
            break;  
        } else if (command.equalsIgnoreCase("a")) {  
            Dragon dragon = new Dragon();  
            dragon.moveLeft();  
            dragon.detectCollision(dragon.x, dragon.y);  
        } else if (command.equalsIgnoreCase("d")) {  
            Dragon dragon = new Dragon();  
            dragon.moveRight();  
            dragon.detectCollision(dragon.x, dragon.y);  
        } else if (command.equalsIgnoreCase("s")) {  
            Dragon dragon = new Dragon();  
            dragon.moveDown();  
            dragon.detectCollision(dragon.x, dragon.y);  
        } else if (command.equalsIgnoreCase("w")) {  
            Dragon dragon = new Dragon();  
            dragon.moveUp();  
            dragon.detectCollision(dragon.x, dragon.y);  
        }  
    }  
}
```