

Jobsheet 2

2.2 Deklarasi Class, Atribut dan Method

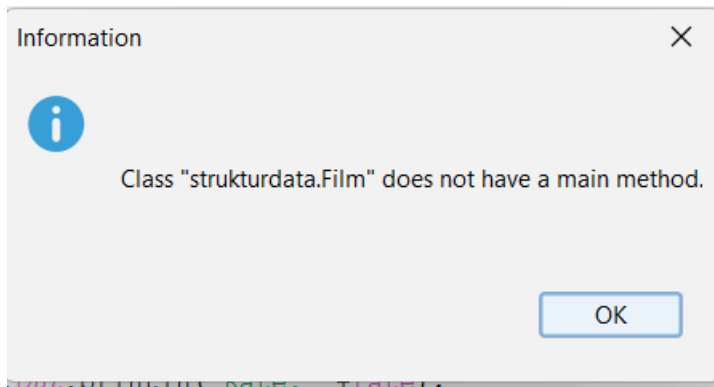
Perhatikan Diagram Class berikut ini:

Film
judul: String genre: String rate: String jumlahTiket: int hargaTiket: int
tampilFilm(): void tambahTiket(n: int): void kurangiTiket(n: int): void totalRevenue(jumlah: int): int

```
public class Film {  
  
    String judul, genre, rate;  
    int jumlahTiket, hargaTiket;  
  
    void tampilFilm() {  
        System.out.println("Judul: "+judul);  
        System.out.println("Genre: "+genre);  
        System.out.println("Rate: "+rate);  
        System.out.println("Jumlah Tiket: "+jumlahTiket);  
        System.out.println("Harga Yiket: "+hargaTiket);  
    }  
    void tambahTiket(int n) {  
        jumlahTiket+=n;  
    }  
    void kurangiTiket(int n) {  
        jumlahTiket-=n;  
    }  
    int totalRevenue(int jumlah) {  
        return jumlah*hargaTiket;  
    }  
}
```

Coba jalankan (Run) class Barang tersebut. Apakah bisa?

- Tidak bisa dijalankan karena tidak ada main



Pertanyaan

1. Sebutkan 2 karakteristik class/objek!

- Memiliki sesuatu dan melakukan sesuatu

2. Kata kunci apakah yang digunakan untuk mendeklarasikan class?

- Public class

3. Perhatikan class Film yang ada di Praktikum di atas, ada berapa atribut yang dimiliki oleh class tersebut? Sebutkan! Dan pada baris berapa saja deklarasi atribut dilakukan?

- Terdapat 5 atribut
- Atribut judul, genre, rate pada baris ke 3
- Atribut jumlah tiket, harga tiket pada baris ke 4

4. Ada berapa method yang dimiliki oleh class tersebut? Sebutkan!

- Terdapat 4 method
- Method tampil film, tambah tiket, kurangi tiket, total revenue

5. Perhatikan method kurangiTiket() yang ada di class Film, modifikasi isi method tersebut sehingga proses pengurangan hanya dilakukan jika stok masih ada (masih lebih besar dari 0)

```
void kurangiTiket(int n) {  
    if(jumlahTiket>0) {  
        jumlahTiket-=n;  
    }  
}
```

6. Menurut Anda, mengapa method tambahTiket() dibuat dengan memiliki 1 parameter berupa bilangan int?

- Dikarenakan pada method tambahTiket melakukan penjumlahan jumlahTiket+=n dan nilai n diambil dari Filmmain

7. Menurut Anda, mengapa method `totalRevenue()` memiliki tipe data `int`?

- Karena method `totalRevenue` melakukan return value

8. Menurut Anda, mengapa method `tambahTiket()` memiliki tipe data `void`?

- Dikarenakan method `tambahTiket` tidak melakukan return value

2.3 Instansiasi Objek dan Mengakses Atribut & Method

```
public class FilmMain {
    public static void main(String[] args) {
        Film film1 = new Film();

        film1.judul="Quantumania Mancing";
        film1.genre="Action Comedy";
        film1.rate="Remaja";
        film1.jumlahTiket=3000;
        film1.hargaTiket=40000;

        film1.tambahTiket(1);
        film1.kurangiTiket(3);
        film1.tampilFilm();

        int income =film1.totalRevenue(4);

        System.out.println("Total Jual 4 tiket= "+income);
    }
}
```

Judul: Quantumania Mancing
Genre: Action Comedy
Rate: Remaja
Jumlah Tiket:2998
Harga Yiket: 40000
Total Jual 4 tiket= 160000

Pertanyaan

1. Pada class `FilmMain`, pada kode apa yang digunakan untuk proses instansiasi? Apa nama objek yang dihasilkan?

- `Film film1=newFilm();`
- Nama objek `film1`

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

- Dengan cara `nama objek.method` seperti `film1.tampilFilm();`

2.4 Membuat Konstruktor

```
Film() {  
}  
  
Film(String jd,String gr,String rt,int jt, int ht){  
    judul=jd;  
    genre=gr;  
    rate=rt;  
    jumlahTiket=jt;  
    hargaTiket=ht;  
}
```

Di Filmmain

```
Film film2= new Film("Maniaquantum", "Horor", "dewasa", 2000, 40000);  
film2.tampilFilm();
```

```
Judul: Quantumania Mancing  
Genre: Action Comedy  
Rate: Remaja  
Jumlah Tiket:2998  
Harga Yiket: 40000  
Total Jual 4 tiket= 160000  
=====
```

```
Judul: Maniaquantum  
Genre: Horror  
Rate: dewasa  
Jumlah Tiket:2000  
Harga Yiket: 40000
```

Pertanyaan

1. Perhatikan class Film yang ada di Praktikum 2.4.1, pada baris berapakah deklarasi konstruktor berparameter dilakukan?

- Pada baris ke 8
 Film(String jd, String gr, String rt, int jt, int ht)

2. Perhatikan class FilmMain di Praktikum 2.4.1, apa sebenarnya yang dilakukan pada baris program dibawah ini?

```
Film film2 = new Film("Maniaquantum", "Horor", "Dewasa", 2000, 40000);
```

- Dikernakan menggunakan konstruktor berparameter maka untuk mengakses atribut perlu dilakukan seperti diatas.

2.5 Latihan Praktikum

1. Buat program berdasarkan diagram class berikut ini!

Buku
nama: String pengarang: String penerbit: String hargaSatuan: int jumlah: int
hitungHargaTotal(): int hitungDiskon(): int hitungHargaBayar(): int

o Method `hitungHargaTotal()` digunakan untuk menghitung harga total yang merupakan perkalian antara `hargaSatuan` dengan `jumlah` barang yang dibeli

o Method `hitungDiskon()` digunakan untuk menghitung diskon dengan aturan sbb:

- Jika harga total > 100000, akan mendapat diskon 10%
- Jika harga total mulai dari 50000 sampai 100000 akan mendapat diskon sebesar 5%
- Jika dibawah 50000 tidak mendapat diskon

o Method `hitungHargaBayar()` digunakan untuk menghitung harga total setelah dikurangi diskon

```
public class Buku {  
    String nama, pengarang, penerbit;  
    int hargaSatuan, jumlah;  
  
    int hitungHargaTotal() {  
        return hargaSatuan * jumlah;  
    }  
    int diskon(int harga) {  
        int diskon = 0;  
        if (harga > 100000) {  
            diskon = harga * 10 / 100;  
        }  
        else if (harga > 50000 || harga < 100000) {  
            diskon = harga * 5 / 100;  
        }  
        return diskon;  
    }  
    int hargaBayar(int harga, int diskon) {  
        int total = harga - diskon;  
        return total;  
    }  
}
```

```

public class Bukumain {
    public static void main(String[] args) {
        Buku bukul = new Buku();
        bukul.nama="PBO";
        bukul.hargaSatuan=20000;
        bukul.jumlah=10;
        bukul.penerbit="polinema";
        bukul.pengarang="naafi";

        int harga=bukul.hitungHargaTotal();
        System.out.println("Harga awal: "+harga);
        int diskon=bukul.diskon(harga);
        System.out.println("Diskon buku: "+diskon);
        int totalHarga=bukul.hargaBayar(harga, diskon);
        System.out.println("Total Harga: "+totalHarga);
    }
}

```

Output

```

Harga awal: 200000
Diskon buku: 20000
Total Harga: 180000

```

2. Buat program berdasarkan diagram class berikut ini!

- Atribut x digunakan untuk menyimpan posisi koordinat x (mendatar) dari snake, 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 snake 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 snake 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 snake menyentuh ujung area permainan

```

public class snake {
    int x,y,width,height;

    void moveLeft() {
        if(x>0&&x<width) {
            x-=1;
            printPosition();
        }
        else{
            detectCollision(x,y);
        }
    }

    void moveRight() {
        if(x>0&&x<width) {
            x+=1;
            printPosition();
        }
        else{
            detectCollision(x,y);
        }
    }

    void moveUp() {
        if(y>0&&y<height) {
            y+=1;
            printPosition();
        }
        else{
            detectCollision(x,y);
        }
    }

    void moveDown() {
        if(y>0&&y<height) {
            y-=1;
            printPosition();
        }
        else{
            detectCollision(x,y);
        }
    }

    void detectCollision(int x, int y) {
        System.out.println("Game over");
    }

    void printPosition() {
        System.out.println(x+", "+y);
    }
}

```

```
public class snakeMain {  
    public static void main(String[] args) {  
        snake snake1=new snake();  
        snake1.x=2;  
        snake1.y=5;  
        snake1.width=10;  
        snake1.height=7;  
  
        snake1.moveLeft();  
        snake1.moveLeft();  
        snake1.moveLeft();  
    }  
}
```

Output

```
1,5  
0,5  
Game over
```