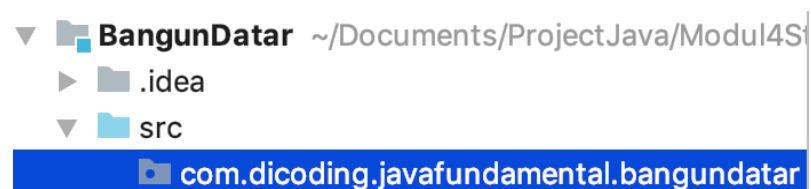


# Studi Kasus : Implementasi Pernyataan Switch, Operator Aritmatika, dan OOP

Program yang terakhir adalah menghitung luas dan keliling dari bangun datar. Kali ini kita akan mencoba menggunakan konsep atau paradigma OOP agar lebih mudah dalam proses pemrogramannya. Anda membuat interface bangun datar agar dapat mengimplementasikannya pada setiap kelas bangun datar yang akan dibuat.

1. Untuk studi kasus, kita bisa membuat project baru dan beri nama BangunDatar. Selanjutnya buatlah package di dalamnya dengan nama `com.dicoding.javafundamental.bangundatar`.



2. Studi kasus pertama ini cukup mudah. Kita buat kelas interface baru dan beri nama `BangunDatar`.

Kemudian tambahkanlah kode berikut:

```
1. package com.dicoding.javafundamental.bangundatar;
2.
3. public interface BangunDatar {
4.     public double luas(int s1, int s2);
5.     public double keliling(int s1, int s2);
6. }
```

3. Buatlah kelas baru dengan nama PersegiPanjang. Kemudian implementasikan interface `BangunDatar` dengan mendeklarasikan setiap fungsi yang ada pada interface tersebut. Karena ini adalah implements maka tambahkan juga statement isi fungsinya.

```
1. package com.dicoding.javafundamental.bangundatar;
2.
3. public class PersegiPanjang implements BangunDatar {
4.
5.     PersegiPanjang() {
6.     }
7.
8.     @Override
9.     public double luas(int s1, int s2) {
10.         return (s1 * s2);
11.     }
12.
13.     @Override
14.     public double keliling(int s1, int s2) {
15.         return (2 * (s1 + s2));
```

4. Buatlah kelas yang lain dengan nama **SegitigaSiku** . Implementasikan juga interface-nya

```
1. package com.dicoding.javafundamental.bangundatar;
2.
3. public class SegitigaSiku implements BangunDatar {
4.
5.     SegitigaSiku() {
6.     }
7.
8.     @Override
9.     public double luas(int s1, int s2) {
10.         return (0.5 * s1 * s2);
11.     }
12.
13.     @Override
14.     public double keliling(int s1, int s2) {
15.         return (s1 + s2 + Math.sqrt((s1 * s1) + (s2 * s2)));
```

5. Buatlah kelas **Lingkaran** , kemudian implementasikan interface kembali. Pada kelas **Lingkaran** ini, kita juga mencoba untuk menerapkan metode **overload** karena rumus menghitung luas dan keliling pada lingkaran cukup dengan satu parameter saja yaitu jari-jari.

```
1. package com.dicoding.javafundamental.bangundatar;
2.
3. public class Lingkaran implements BangunDatar {
4.
5.     private double pi = 3.1415;
6.
7.     Lingkaran() {
8.     }
9.
10.    @Override
11.    public double luas(int s1, int s2) {
12.        return 0;
13.    }
14.
15.    @Override
```

6. Terakhir buat kelas **Main** untuk menjalankan program. Pada kelas ini kita instansiasi kelas setiap bangun datar.

```
1. package com.dicoding.javafundamental.bangundatar;
2.
3. public class Main {
4.     public static void main(String[] args) {
5.         /*Kamus*/
6.         PersegiPanjang opersegi;
7.         SegitigaSiku osegitiga;
8.         Lingkaran olingkaran;
9.         int s1, s2, r;
10.
11.         opersegi = new PersegiPanjang();
12.         osegitiga = new SegitigaSiku();
13.         olingkaran = new Lingkaran();
14.     }
15. }
```

7. Kita buat metode `mainMenu`. Metode ini digunakan untuk menampilkan pilihan menu.

```
1. package com.dicoding.javafundamental.bangundatar;
2.
3. import java.util.Scanner;
4.
5. public class Main {
6.
7.     private static char pilihan;
8.     private static Scanner scanner = new Scanner(System.in);
9.
10.     public static void main(String[] args) {
11.         /*Kamus*/
12.         PersegiPanjang opersegi;
13.         SegitigaSiku osegitiga;
14.         Lingkaran olingkaran;
15.         int s1, s2, r;
```

8. Lengkapi kode di dalam main menjadi seperti ini:

```
1. package com.dicoding.javafundamental.bangundatar;
2.
3. import java.util.Scanner;
4.
5. public class Main {
6.
7.     private static char pilihan;
8.     private static Scanner scanner = new Scanner(System.in);
9.
10.    public static void main(String[] args) {
11.        /*Kamus*/
12.        PersegiPanjang opersegi;
13.        SegitigaSiku osegitiga;
14.        Lingkaran olingkaran;
15.        int s1, s2, r;
```

9. Untuk menjalankan program yang telah dibuat, klik tombol run. Setelah menalankan program tersebut, console akan menampilkan seperti berikut ini. Pertama akan muncul pilihan menu bangun datar yang akan kita hitung luas dan kelilingnya. Untuk mencoba, bisa pilih antara 1-3. Misalnya Anda bisa pilih menu 1, yaitu menghitung luas dan keliling persegi panjang. Maka Anda akan diminta untuk memasukkan panjang dan lebar. Setelah itu, akan keluar hasil perhitungan dari luas dan kelilingpersegi panjang. Outputnya akan jadi seperti ini:

```
=====
Hitung Luas dan Keliling :
1. Persegi Panjang
2. Segitiga Siku
3. Lingkaran
Masukan Menu : 1
Masukan panjang persegi: 10
Masukan lebar persegi: 20
Luas persegi panjang: 200.0
```

Untuk mencoba memilih menu lain, jalankan lagi programnya dan pilih menghitung luas dan keliling pada bangun datar yang lain. Selain itu Anda juga bisa membuat bangun datar yang lain, misalnya trapesium, layang-layang, belah ketupat, dll. Untuk tambahan menghitung luas dan kelilingnya. Selamat mencoba dan berkreasi.

Silakan unduh kode kasus di atas pada tautan berikut:

- [Source Code Bangun Datar](#).



Dicoding Space

Jl. Batik Kumeli No.50, Sukaluyu,  
Kec. Cibeunying Kaler, Kota Bandung  
Jawa Barat 40123

Penghargaan



Decode Ideas  
Discover Potential

> [Tentang Kami](#)

[Blog](#)

[Reward](#)

[Showcase](#)

[Hubungi Kami](#)

[FAQ](#)