

# LEMBAR KERJA PRAKTIKUM

## PEMROGRAMAN II

Dosen Pengampu:

1. Andreyan Rizky Baskara, S.Kom., M.Kom

### Praktikum 5. Polimorfisme

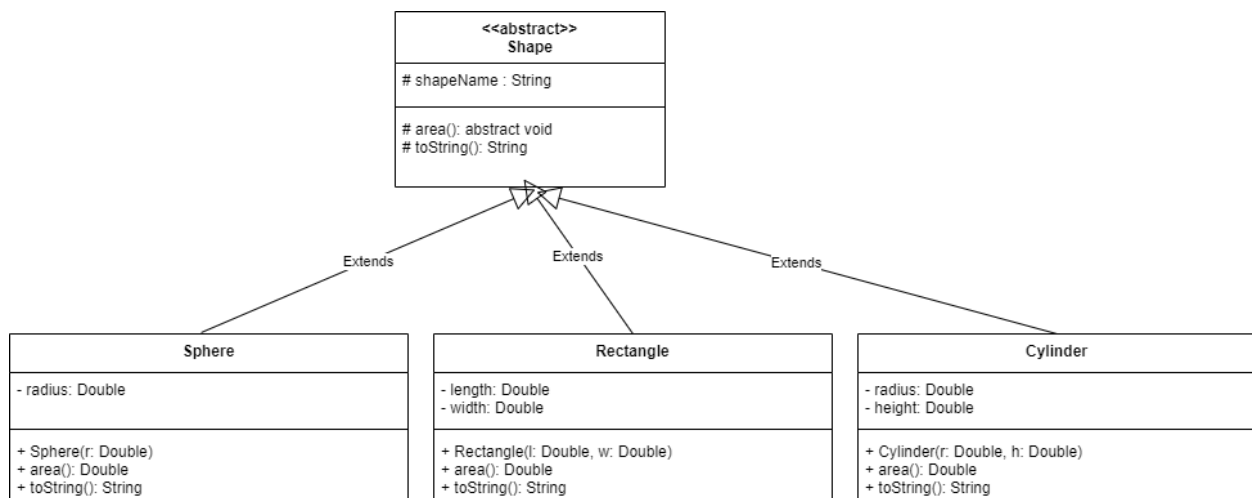
#### Tujuan Praktikum:

1. Dapat mengimplementasikan polimorfisme yang baik dan benar menggunakan Bahasa pemrograman Java

Soal Praktikum:

Pada praktikum kali ini anda akan diminta untuk membuat sebuah program yang dapat menghitung banyaknya liter cat yang digunakan untuk mewarnai bentuk ruang yang beragam.

Buatlah sebuah hierarki kelas abstrak Shape dimana memiliki 3 kelas anak yaitu Sphere, Rectangle, dan Cylinder seperti ditunjukkan oleh diagram kelas berikut.



Method `area()` digunakan untuk menghitung luas masing-masing objek. Berikut adalah formula yang digunakan untuk menghitung luas masing-masing bangun yang harus diimplementasikan.

Sphere:  $4 \times \pi \times radius^2$

Rectangle:  $length \times width$

Cylinder:  $\pi \times radius^2 \times height$

Method toString() digunakan untuk mengembalikan nilai String dari nama bangun

Berikut adalah ilustrasi dari kelas Sphere.java. Implementasikan kelas lainnya untuk Shape, Rectangle dan Cyclinder

#### Contoh Ilustrasi Sphere.java

```
public class Sphere extends Shape
{
private double radius; //radius in feet
//-----
// Constructor: Sets up the sphere.
//-----
public Sphere(double r)
{
super("Sphere");
radius = r;
}
//-----
// Returns the surface area of the sphere.
//-----
public double area()
{
return 4*Math.PI*(radius*radius);
}
//-----
// Returns the sphere as a String.
//-----
public String toString()
{
return super.toString() + " of radius " + radius;
}
}
```

Selanjutnya, Buatlah kelas Paint.java seperti ditunjukkan diagram kelas berikut.

Paint
- coverage: Double
+ Paint(c: Double) + amount(s: Shape) : Double

Method amount digunakan untuk menghitung banyaknya liter cat yang digunakan dengan persamaan berikut:

$$\text{amount of paint} = \frac{\text{area of shape}}{\text{coverage}}$$

Lengkapi kode dibawah supaya menghasilkan keluaran yang diinginkan

#### Paint.java

```
public class Paint
{
    private double coverage; //number of square feet per gallon
    //-----
    // Constructor: Sets up the paint object.
    //-----
    public Paint(double c)
    {
        coverage = c;
    }
    //-----
    // Returns the amount of paint (number of gallons)
    // needed to paint the shape given as the parameter.
    //-----
    public double amount(Shape s)
    {
        System.out.println ("Computing amount for " + s);
        return 0;
    }
}
```

Terakhir, Buatlah kelas main bernama PaintThings.java. Tambahkan beberapa hal berikut agar program berjalan sesuai yang diinginkan.

1. Instansiasi 3 bentuk objek:
  - a. objek bernama deck berbentuk persegi panjang dengan ukuran Panjang 20cm dan lebar 30cm.
  - b. objek bernama bigBall berbentuk bola dengan ukuran radius 15cm.
  - c. objek bernama tank berbentuk silinder dengan ukuran radius 10cm dan tinggi 30cm.
2. Panggil fungsi yang tepat agar dapat menghitung jumlah cat yang diperlukan.

Petunjuk untuk kelas main PaintThings.java

```
import java.text.DecimalFormat;
public class PaintThings
{
//-----
// Creates some shapes and a Paint object
// and prints the amount of paint needed
// to paint each shape.
//-----
public static void main (String[] args)
{
final double COVERAGE = 350;
Paint paint = new Paint(COVERAGE);
Rectangle deck;
Sphere bigBall;
Cylinder tank;
double deckAmt, ballAmt, tankAmt;

// Instantiate the three shapes to paint

// Compute the amount of paint needed for each shape

// Print the amount of paint for each.
DecimalFormat fmt = new DecimalFormat("0.##");
System.out.println ("\nNumber of gallons of paint needed...");
System.out.println ("Deck " + fmt.format(deckAmt));
```

```
System.out.println ("Big Ball " + fmt.format(ballAmt));  
System.out.println ("Tank " + fmt.format(tankAmt));  
}  
}
```

1. Jalankan program dan perhatikan hasil untuk ketiga bentuk yang berbeda, screenshot hasil yang didapatkan dan lampirkan di dalam source code.
2. Simpan coding anda dengan nama package: **soal1**
3. Pastikan terdapat screenshoot pada repositori github