**LAPORAN PRAKTIKUM**

**PEMROGRAMAN II**

**MODUL 5**

****

**Polimorfisme**

**Oleh:**

**Dhea Aprilinda Utami NIM. 2210817220019**

**PROGRAM STUDI TEKNOLOGI INFORMASI**

**FAKULTAS TEKNIK**

**UNIVERSITAS LAMBUNG MANGKURAT**

**DESEMBER 2023**

# LEMBAR PENGESAHAN

**LAPORAN PRAKTIKUM PEMROGRAMAN II**

**MODUL 4**

Laporan Praktikum Pemrograman II Modul 5: Polimorfisme ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman II. Laporan Praktikum ini dikerjakan oleh:

Nama Praktikan : Dhea Aprilinda Utami

NIM : 2210817220019

|  |  |
| --- | --- |
| Menyetujui,  Asisten Praktikum  Bachrul Uluum  NIM. 2010817210025 | Mengetahui,  Dosen Penanggung Jawab Praktikum  Andreyan Rizky Baskara, S.Kom., M.Kom NIP. 19930703 201903 1 011 |

# DAFTAR ISI

[LEMBAR PENGESAHAN 2](#_Toc149767889)

[DAFTAR ISI 3](#_Toc149767890)

[DAFTAR GAMBAR 4](#_Toc149767891)

[DAFTAR TABEL 5](#_Toc149767892)

[Soal 1 6](#_Toc149767893)

[A. Source Code 6](#_Toc149767894)

[B. Output Program 8](#_Toc149767895)

[C. Pembahasan 8](#_Toc149767896)

[D. Tautan GIT 9](#_Toc149767897)

[Soal 2 10](#_Toc149767898)

[A. Source Code 11](#_Toc149767899)

[B. Output Program 14](#_Toc149767900)

[C. Pembahasan 14](#_Toc149767901)

[D. Tautan GIT 17](#_Toc149767902)

[Soal 3 18](#_Toc149767903)

[A. Source Code 19](#_Toc149767904)

[B. Output Program 22](#_Toc149767905)

[C. Pembahasan 23](#_Toc149767906)

[D. Tautan GIT 26](#_Toc149767907)

# DAFTAR GAMBAR

[Gambar 1. Screenshot Output Soal 1 Java 8](#_Toc149767908)

[Gambar 2. Screenshot Output Soal 2 Java 14](#_Toc149767909)

[Gambar 3. Screenshot Output Soal 3 Java 23](#_Toc149767910)

# DAFTAR TABEL

[Tabel 1. Source Code Soal 1 Shape.Java 10](#_Toc152708974)

[Tabel 2. Source Code Soal 1 Shapire.Java 10](#_Toc152708975)

[Tabel 3. Source Code Soal 1 Rectangle.Java 11](#_Toc152708976)

[Tabel 4. Source Code Soal 1 Cylinder.Java 12](#_Toc152708977)

[Tabel 5. Source Code Soal 1 Paint.Java 12](#_Toc152708978)

[Tabel 6. Source Code Soal 1 PaintThings.Java 13](#_Toc152708979)

# Soal 1

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×𝜋×𝑟𝑎𝑑𝑖𝑢𝑠2

Rectangle: 𝑙𝑒𝑛𝑔𝑡ℎ ×𝑤𝑖𝑑𝑡ℎ

Cylinder: 𝜋×𝑟𝑎𝑑𝑖𝑢𝑠2×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.



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

𝑎𝑚𝑜𝑢𝑛𝑡 𝑜𝑓 𝑝𝑎𝑖𝑛𝑡= 𝑎𝑟𝑒𝑎 𝑜𝑓 𝑠ℎ𝑎𝑝𝑒/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:
2. objek bernama deck berbentuk persegi panjang dengan ukuran Panjang 20cm dan lebar 30cm.
3. objek bernama bigBall berbentuk bola dengan ukuran radius 15cm.
4. objek bernama tank berbentuk silinder dengan ukuran radius 10cm dan tinggi 30cm.
5. 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

## Source Code

Shape.Java

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | package soal1;  public abstract class Shape {  String Name;  public Shape(String shapename){  Name = shapename;  }  public abstract double area();  public String toString(){  return Name;  }  } |

Tabel 1. Source Code Soal 1 Shape.Java

Shapire.java

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | package soal1;  public class Sphere extends Shape {  private double radius;  public Sphere(double r){  super("Sphere");  radius = r;  }  public double area()  {  return 4\*Math.PI\*(radius\*radius);  }  public String toString()  {  return super.toString() + " of radius " + radius;  }  } |

Tabel 2. Source Code Soal 1 Shapire.Java

Rectangle.java

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24 | package soal1;  public class Rectangle extends Shape{  private double length;  private double width;  public Rectangle(double l, double w){  super("Rectangle");  length = l;  width = w ;  }  @Override  public double area() {  return length\*width;  }  @Override  public String toString() {  return "Rectangle" +  "length " + length +  "and width " + width;  }  } |

Tabel 3. Source Code Soal 1 Rectangle.Java

Cylinder.java

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23 | package soal1;  public class Cylinder extends Shape{  private double radius;  private double height;  public Cylinder(double r, double h){  super("Cylinder");  radius = r;  height = h;  }  @Override  public double area() {  return Math.PI\*(radius\*radius)\*height;  }  @Override  public String toString() {  return "Cylinder" +  "radius " + radius +  "and height " + height;  }  } |

Tabel 4. Source Code Soal 1 Cylinder.Java

Paint.java

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | package soal1;  public class Paint {  double coverage;  public Paint (double c){  coverage = c;  }  public double amount(Shape s)  {  System.out.println ("Computing amount for " + s);  return s.area()/coverage;  }  } |

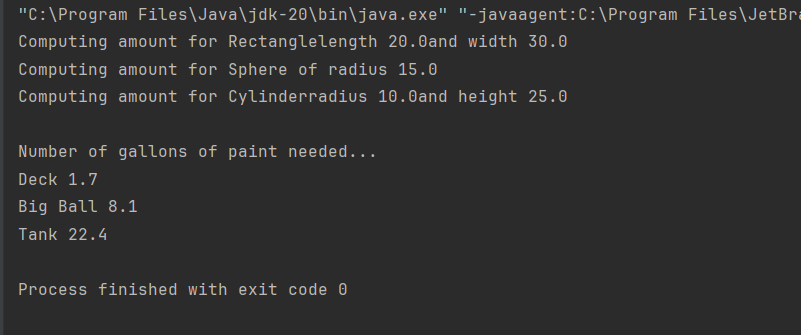
Tabel 5. Source Code Soal 1 Paint.Java

PaintThings.java

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21 | package soal1;  import java.text.DecimalFormat;  public class PaintThings {  public static void main (String[] args){  final double COVERAGE = 350;  Paint paint = new Paint(COVERAGE);  Rectangle deck = new Rectangle(20, 30);  Sphere bigBall = new Sphere(15);  Cylinder tank = new Cylinder(10, 25);  double deckAmt = paint.amount(deck);  double ballAmt = paint.amount(bigBall);  double tankAmt = paint.amount(tank);  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));  }  } |

Tabel 6. Source Code Soal 1 PaintThings.Java

## Output Program



Gambar 1. Screenshot Output Soal 1 Java

## Pembahasan

Shape.java:

1. Package soal1 merupakan penjelasan dimana file soal1main berada di package soal1.
2. public adalah sebuah keyword di Java yang menandakan bahwa objek, method, atau atribut dapat diakses dari class lain, abstract adalah kelas yang terletak di posisi tertinggi dalam hierarki class, class adalah sebuah keyword di Java yang digunakan untuk membuat sebuah class yaitu shape.
3. Mendefinisikan nama sebagai string.
4. Membuat constructor dari shape.
5. Mendefinisikan name = shapename.
6. }
7. public adalah sebuah keyword di Java yang menandakan bahwa objek, method, atau atribut dapat diakses dari class lain, mefiniskan area sebagai double.
8. Sintaks dasar dari method String toString() yang mengembalikan string sebagai hasil.
9. Mengembalikan ke dalam variable nama.
10. }
11. }

Shapire.java

1. Package soal1 merupakan penjelasan dimana file soal1main berada di package soal1.
3. public adalah sebuah keyword di Java yang menandakan bahwa objek, method, atau atribut dapat diakses dari class lain. Extends merupakan turunana dari parent class yaitu shape.
4. Mendefiniskan radius sebgai double pada private.
5. Membuat constructor dari shapire
6. Suoer spere.
7. Radius = r.
8. }
9. public adalah sebuah keyword di Java yang menandakan bahwa objek, method, atau atribut dapat diakses dari class lain, mefiniskan area sebagai double.
10. {
11. Mengembalikan nilai sebagai 4\*pi\*radius\*radius
12. }
13. Sintaks dasar dari method String toString().
14. }
15. Mengembalikan ke dalam variable radius.
16. }
17. }

Rectangle.java

1. Package soal1 merupakan penjelasan dimana file soal1main berada di package soal1.
3. public adalah sebuah keyword di Java yang menandakan bahwa objek, method, atau atribut dapat diakses dari class lain. Extends merupakan turunana dari parent class yaitu shape.
4. Mendefiniskan length sebgai double pada private.
5. Mendefiniskan width sebgai double pada private.
6. Membuat constructor dari Rectangle.
7. Suoer Rectangle.
8. Length = l.
9. Width = w.
10. }
12. @override adalah sebuah pembuatan ulang metode yang ada pada superclass di subclass atau anak kelas.
13. public adalah sebuah keyword di Java yang menandakan bahwa objek, method, atau atribut dapat diakses dari class lain, mefiniskan area sebagai double.
14. Mengembalikan nilai sebagai length\*width.
15. }
16. @override adalah sebuah pembuatan ulang metode yang ada pada superclass di subclass atau anak kelas.
17. Sintaks dasar dari method String toString().
18. Mengembalikan nilai.
19. Length.
20. Width.
21. }
22. }

Cylinder.java

1. Package soal1 merupakan penjelasan dimana file soal1main berada di package soal1.
3. public adalah sebuah keyword di Java yang menandakan bahwa objek, method, atau atribut dapat diakses dari class lain. Extends merupakan turunana dari parent class yaitu shape.
4. Mendefiniskan radius sebgai double pada private.
5. Mendefiniskan height sebgai double pada private.
6. Membuat constructor dari Cylinder.
7. Suoer Cylinder.
8. Radius = r.
9. Haight = h.
10. }
12. @override adalah sebuah pembuatan ulang metode yang ada pada superclass di subclass atau anak kelas.
13. public adalah sebuah keyword di Java yang menandakan bahwa objek, method, atau atribut dapat diakses dari class lain, mefiniskan area sebagai double.
14. Mengembalikan nilai sebagai pi\*radius\*radius\*height.
15. }
16. @override adalah sebuah pembuatan ulang metode yang ada pada superclass di subclass atau anak kelas.
17. Sintaks dasar dari method String toString().
18. Mengembalikan nilai.
19. radius.
20. height.
21. }
22. }

Paint.java

1. Package soal1 merupakan penjelasan dimana file soal1main berada di package soal1.
3. public adalah sebuah keyword di Java yang menandakan bahwa objek, method, atau atribut dapat diakses dari class lain. Extends merupakan turunana dari parent class yaitu Paint.
4. Mendefiniskan radius sebgai double.
5. Membuat constructor dari paint.
6. Coverage = c.
7. }
8. public adalah sebuah keyword di Java yang menandakan bahwa objek, method, atau atribut dapat diakses dari class lain, mefiniskan area sebagai double.
9. {
10. Perintah untuk mencetak hasil output.
11. Mengembalikan nilai sebagai s.area/coverage.
12. }
13. }

PaintThings.java

1. Package soal1 merupakan penjelasan dimana file soal1main berada di package soal1.
2. Digunakan untuk mengimport teks dengan format decimal.
4. public adalah sebuah keyword di Java yang menandakan bahwa objek, method, atau atribut dapat diakses dari class lain. Extends merupakan turunana dari parent class yaitu PaintThings.
5. public adalah sebuah keyword di Java yang menandakan bahwa objek, method, atau atribut dapat diakses dari class lain, public adalah sebuah keyword di Java yang menandakan bahwa objek, method, atau atribut dapat diakses dari class lain, public adalah sebuah keyword di Java yang menandakan bahwa objek, method, atau atribut dapat diakses dari class lain, main adalah nama fungsi yang digunakan oleh Java sebagai titik masuk ke program. String[] args adalah parameter yang diperlukan untuk fungsi main. Parameter ini adalah array dari argumen baris perintah yang dapat diteruskan ke program Java.
7. Memasukkan dinali dari coverage sebanyak 350.
8. Membuat objek baru Bernama paint.
9. Membuat objek baru Bernama Rectangle.
10. Membuat objek baru Bernama Sphere.
11. Membuat objek baru Bernama Cylinder.
12. Digunakan untuk menghitung Rectangle.
13. Digunakan untuk menghitung Sphere.
14. Digunakan untuk menghitung Cylinder.
15. Digunkan untuk merubah menjadi format decimal.
16. Perintah untuk mencetak hasil output.
17. Perintah untuk mencetak hasil output.
18. Perintah untuk mencetak hasil output.
19. Perintah untuk mencetak hasil output.
20. }
21. }

## Tautan GIT

https://github.com/dheaaprilinda01/Praktikum-Pemprograman-2/tree/main/PRAKTIKUM%204/src/soal1