

Object Oriented Programming Inheritance PPT



From:

AL AZHAR RIZQI RIFA'I FIRDAUS

Class:

2 I

Absence:

01

Student Number Identity:

2241720263

Department:

Information Technology

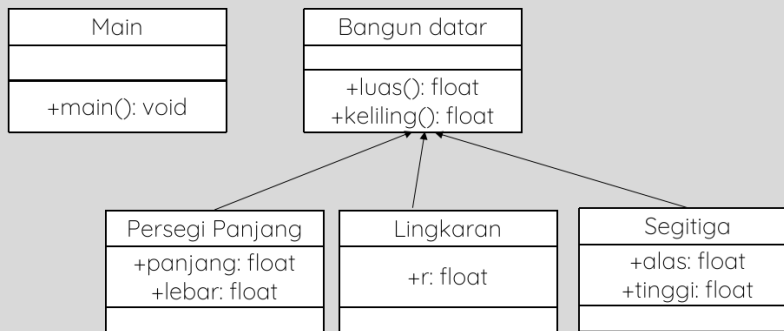
Study Program:

Informatics Engineering

Assignment PowerPoint

Soal

1. Tuliskan program yang berfungsi untuk menghitung luas dan keliling bangun datar.



Code:

```
Main.java × FlatShape.java Rectangle.java Circle.java Triangle.java
src > main > java > com > azhar > ppt > Main.java > Main > main(String[])
1 package com.azhar.ppt;
2
3 public class Main {
    Run | Debug | Codeium: Refactor | Explain | Generate Javadoc
4     public static void main(String[] args) {
5         System.out.println("Rectangle");
6         Rectangle rectangle = new Rectangle();
7         rectangle.setLength(length:10);
8         rectangle.setWidth(width:20);
9         System.out.println("Length: " + rectangle.getLength());
10        System.out.println("Width: " + rectangle.getWidth());
11        System.out.println("Area: " + rectangle.area());
12        System.out.println("Circumference: " + rectangle.circumference()+ "\n");
13
14        System.out.println("Circle");
15        Circle circle = new Circle();
16        circle.setR(r:10);
17        System.out.println("Radius: " + circle.getR());
18        System.out.println("Area: " + circle.area());
19        System.out.println("Circumference: " + circle.circumference()+ "\n");
20
21        System.out.println("Triangle");
22        Triangle triangle = new Triangle();
23        triangle.setHeight(height:10);
24        triangle.setBase(base:20);
25        System.out.println("Height: " + triangle.getHeight());
26        System.out.println("Base: " + triangle.getBase());
27        System.out.println("Area: " + triangle.area());
28        System.out.println("Circumference: " + triangle.circumference()+ "\n");
29    }
30 }
31
```

```
Main.java FlatShape.java × Rectangle.java
src > main > java > com > azhar > ppt > FlatShape.java
1 package com.azhar.ppt;
2
3 public class FlatShape {
    Codeium: Refactor | Explain | Generate Javadoc
4     public float area() {
5         return 0;
6     }
7
8     Codeium: Refactor | Explain | Generate Javadoc
9     public float circumference() {
10         return 0;
11     }
12
```

Main.java FlatShape.java Rectangle.java X Circle.java

src > main > java > com > azhar > ppt > Rectangle.java > Rectangle >

```
1  package com.azhar.ppt;
2
3  public class Rectangle extends FlatShape{
4      private float length;
5      private float width;
6
7      public Rectangle() {}
8
9      }
10
11     public Rectangle(float length, float width) {
12         this.length = length;
13         this.width = width;
14     }
15
16     Codeium: Refactor | Explain | Generate Javadoc
17     @Override
18     public float area() {
19         return length * width;
20     }
21
22     Codeium: Refactor | Explain | Generate Javadoc
23     @Override
24     public float circumference() {
25         return 2 * (length + width);
26     }
27
28     Codeium: Refactor | Explain | Generate Javadoc
29     public void setLength(float length) {
30         this.length = length;
31     }
32
33     Codeium: Refactor | Explain | Generate Javadoc
34     public float getLength() {
35         return length;
36     }
37
38     Codeium: Refactor | Explain | Generate Javadoc
39     public void setWidth(float width) {
40         this.width = width;
41     }
42 }
```

Codeium: Refactor | Explain | Generate Javadoc

```
38     public float getWidth() {  
39         return width;  
40     }  
41 }
```

Main.java FlatShape.java Rectangle.java Circle.java X

src > main > java > com > azhar > ppt > Circle.java > Circle > Circle()

```
1 package com.azhar.ppt;
2
3 public class Circle extends FlatShape{
4     private float r;
5
6     public Circle() {}
7
8
9     public Circle(float r) {
10         this.r = r;
11     }
12
13     @Override
14     public float area() {
15         return (float) (Math.PI * r * r);
16     }
17
18     @Override
19     public float circumference() {
20         return 2 * (float) (Math.PI * r);
21     }
22
23     public void setR(float r) {
24         this.r = r;
25     }
26
27     public float getR() {
28         return r;
29     }
30
31 }
32
```

Triangle.java ×

src > main > java > com > azhar > ppt > Triangle.java > Triangle > getHe

```
1  package com.azhar.ppt;
2
3  public class Triangle extends FlatShape {
4      private float height;
5      private float base;
6
7      public Triangle() {
8
9      }
10
11     public Triangle(float height, float base) {
12         this.height = height;
13         this.base = base;
14     }
15
```

Codeium: Refactor | Explain | Generate Javadoc

```
16  @Override
17  public float area() {
18      return (0.5f) * base * height;
19  }
20
```

Codeium: Refactor | Explain | Generate Javadoc

```
21  @Override
22  public float circumference() {
23      return base + (height * 2f);
24  }
25
```

Codeium: Refactor | Explain | Generate Javadoc

```
26  public float getHeight() {
27      return height;
28  }
```

Codeium: Refactor | Explain | Generate Javadoc

```
30  public void setHeight(float height) {
31      this.height = height;
32  }
33
```

Codeium: Refactor | Explain | Generate Javadoc

```
34  public float getBase() {
35      return base;
36  }
37
```

```
Codeium: Refactor | Explain | Generate Javadoc
38   public void setBase(float base) {
39       this.base = base;
40   }
41 }
42
```

Result:

```
→ zharsuke@box ~/Documents/College/Semester_3/oop/meet-6/coding git:(master) x
/home/zharsuke/Documents/College/Semester_3/oop/meet-6/coding/target/classes com
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Rectangle
Length: 10.0
Width: 20.0
Area: 200.0
Circumference: 60.0

Circle
Radius: 10.0
Area: 314.15927
Circumference: 62.831852

Triangle
Height: 10.0
Base: 20.0
Area: 100.0
Circumference: 40.0

→ zharsuke@box ~/Documents/College/Semester_3/oop/meet-6/coding git:(master) x
```