[Shaji Ram Prakash Balasubramaniyan](mailto:shaji.balasubramaniyan@relevantz.com)

| **Problem Solving :** *Define an application to find the area() and perimeter() of various shapes.* | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|
|
|
| 1.What? | | | |  | 2.How? | | | |
| **1.**What are the shapes we have to consider?  **Ans :** Circle, Rectangle, Square and Triangle.  **2.**What are the parameters we have to consider?  **Ans :** Length, breadth, Side and Radius.  **3.**What are the formulas we have to use to calculate area and  perimeter?  **Ans : Area - Circle :** PI\*R^2, **Rectangle :** Length \* Breadth,  **Square :** Side \* Side, **Triangle :** 1/2\*Breadth \* Height.  **Perimeter - Circle :** 2\*PI\*R, **Rectangle :** 2(Length+Breadth),  **Square :** 4 \* Side, **Triangle : (**Side1+ Side2 + Side3).  **4.**Any predefined values are required?  **Ans : PI :** 3.14 | | | |  | **1. Using Single class read all the required inputs and calculate area**  **and Perimeter. Display the result.**  **2. Using Single class and method overloading calculate area and**  **perimeter. Display the result.**  **3.Using different classes for various shapes and**  **calculate area and perimeter And displaythe result.**  **4.Using different classes for various shapes and inherit the common**  **properties from a class called "Shape" and calculate area and**  **perimeter and display the result.**  **5.Using different classes for various shapes and inherit the common**  **properties from a abstract class called "Shape" and calculate area**  **and perimeter and display the result.**  **6. Using different classes for various shapes and inherit the common**  **properties from a class called "Shape" and implements an interface**  **called "ShapePlan" and calculate area and perimeter and display**  **the result.** | | | |
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|  | | | |  | | | |
| 6. Using different classes for various shapes and inherit the common  properties from a class called "Shape" and implements an interface  called "ShapePlan" and calculate area and perimeter and display  the result.  **Reason :**  1.We can separate the common properties.  2. We can only declare in interface (Secure).  3. We can declare and also can define in abstract.  4. We can achieve 100% abstraction in interface.  5. Code reusability.  6. We can make a plan by using interface.  7. Code flexibility (Updation). | | | | 1. Comparing to remaining 2 that is more secure.  2. This 2 is come without interface.  3. We can't achieve 100% abstraction.  4. The 2nd one is partially completed.  5. When we compare to that code flexibility is less efficient in  this 2.  6. We can't plan in this 2 solutions because of absence of  interface. | | | |
|
|
|
|
|
|
|
|
|
| 3.Why? | | | |  | 4.Why Not? | | | |

**ALGORITHM :**

**Step1 :** Start.

**Step2 :** Get the inputs from the clients(Shape and Parameters).

**Step3 :** Create an Interface class named Shape\_Plan.

**Step4 :** Define a method perimeter and area and a variable PI = 3.14 in the interface.

**Step5 :** Make an Abstract class named Shapes and implement Shape\_Plan.

**Step6 :** Declare variables(Length, Breadth, Side, Radius).

**Step7 :** Inherit the abstract class and override the methods.

**Step8 :** Give the correct formulas inside the methods.

**Step9 :** Stop.

**CODING :**

**interface Shapes\_Plan {**

**void Perimeter();**

**void Area();**

**double *PI* = 3.14f;**

**}**

**abstract class Shapes implements Shapes\_Plan {**

**public abstract void Perimeter();**

**public abstract void Area();**

**}**

**class Circle extends Shapes {**

**float radius;**

**Circle(float radius){**

**this.radius = radius;**

**}**

**public void Perimeter() {**

**System.*out*.println("Perimeter of Circle is " + (2\**PI*\*radius));**

**}**

**public void Area() {**

**System.*out*.println("Area of the Circle is " + (*PI*\*radius\*radius));**

**}**

**}**

**class Rectangle extends Shapes {**

**int length,breadth;**

**Rectangle(int length, int breadth){**

**this.length = length;**

**this.breadth = breadth;**

**}**

**public void Perimeter() {**

**System.*out*.println("Perimeter of Rectangle is " + (2\*(length+breadth)));**

**}**

**public void Area() {**

**System.*out*.println("Area of the Rectangle is " + (length\*breadth));**

**}**

**}**

**class Square extends Shapes {**

**int side;**

**Square(int side){**

**this.side = side;**

**}**

**public void Perimeter() {**

**System.*out*.println("Perimeter of Square is " + (4\*side));**

**}**

**public void Area() {**

**System.*out*.println("Area of the Square is " + (side\*side));**

**}**

**}**

**class Triangle extends Shapes {**

**int length,base,height;**

**Triangle(int length, int base, int height){**

**this.length = length;**

**this.base = base;**

**this.height = height;**

**}**

**public void Perimeter() {**

**System.*out*.println("Perimeter of Triangle is " + ((base\*height)/2));**

**}**

**public void Area() {**

**System.*out*.println("Area of the Triangle is " + (length+base+height));**

**}**

**}**

**public class Problem\_Solving\_Shapes {**

**public static void main(String[] args) {**

**Circle circle = new Circle(24);**

**circle.Perimeter();**

**circle.Area();**

**System.*out*.println("");**

**Rectangle rectangle = new Rectangle(14,24);**

**rectangle.Perimeter();**

**rectangle.Area();**

**System.*out*.println("");**

**Square square = new Square(24);**

**square.Perimeter();**

**square.Area();**

**System.*out*.println("");**

**Triangle triangle = new Triangle(11,14,24);**

**triangle.Perimeter();**

**triangle.Area();**

**}**

**}**