| **Problem Statement : Define a application to find the area() and volume() of various Shapes** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
| **What ? 1** | | | |  | **How ? 2** | | | |
| 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, Radius  3) What are the formulas we have to use to calculate area and volume?  **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? | | | |  | 1) Using Single class read all the required inputs and calculate area and perimeter and display the result  2) Using Single class and method overloading calculate area and perimeter and display the result.  3) Using different classes for various shapes and calculate area and perimeter and display the 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 abstract class called "Shape" and implements an interface called "ShapePlan" and calculate area and perimeter and display the result.** | | | |
|  |  |  |  |  |  |  |  |  |
| **Why ? 3** | | | |  | **Why Not ? 4** | | | |
| Using different classes for various shapes and inherit the common properties from a abstract class called "Shape" and implements an interface called "ShapePlan" and calculate area and perimeter and display the result.  **Reason:**  1) It is fully Abstract Class  2) We can achieve multiple inheritance using interface  3) Code reusability using inheritance  4) We can declare and define in abstract class  5) We can only declare in interface | | | |  | 1) 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.  2) 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.  **Reason:**  1) We cannot instantiate object for an abstract class  2) Abstract class doesn't support multiple inheritance. | | | |

**ALGORITHM: (For Shapes)**

**Step 1:** Start

**Step 2:** Create an interface called “ShapePlan”

**Step 3:** Declare the area and perimeter methods inside the interface

**Step 4:** Implement the interface in the abstract class named “Shape”

**Step 5:** Declare the variables (Length, Breadth, Side, Height)

**Step 6:** Define the area and perimeter methods in abstract class

**Step 7:** Create the class for various shapes (Rectangle, Square, Triangle, Circle) and extends by the “Shape” class and define the methods

**Step 8:** Apply the correct formula for the shapes

**Step 9:** Inside the formula declare the Pi value as (constant)-3.14

**Step 10:** Declare the object inside the main class and call the subclasses by their names (Shapes, Area, Perimeter)

**Step 11:** Display the results

**Step 12:** Stop.

**PROGRAM:**

**package** com.shiva.day23;

**interface** ShapePlan

{

**void** area();

**void** perimeter();

}

**abstract** **class** Shape **implements** ShapePlan

{

**public** **abstract** **void** area();

**public** **abstract** **void** perimeter();

}

**class** Rectangle **extends** Shape

{

**float** breadth,length;

Rectangle(**float** length, **float** breadth)

{

**this**.length=length;

**this**.breadth=breadth;

}

**public** **void** area()

{

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

}

**public** **void** perimeter()

{

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

}

}

**class** Square **extends** Shape

{

**int** side;

Square(**int** side)

{

**this**.side=side;

}

**public** **void** area()

{

System.***out***.println("The area of the Square : "+(side\*side));

}

**public** **void** perimeter()

{

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

}

}

**class** Circle **extends** Shape

{

**float** radius;

Circle(**float** radius)

{

**this**.radius=radius;

}

**public** **void** area()

{

System.***out***.println("The area of the Circle : "+(3.14\*radius\*radius));

}

**public** **void** perimeter()

{

System.***out***.println("Perimeter of the Square : "+(2\*3.14\*radius));

}

}

**class** Triangle **extends** Shape

{

**float** breadth,height,side1,side2,side3;

Triangle(**float** breadth, **float** height, **float** side1, **float** side2, **float** side3)

{

**this**.breadth=breadth;

**this**.height=height;

**this**.side1=side1;

**this**.side2=side2;

**this**.side3=side3;

}

**public** **void** area()

{

System.***out***.println("Area of the Triangle : "+(1/2\*breadth\*height));

}

**public** **void** perimeter()

{

System.***out***.println("Perimeter of the Triangle : "+(side1+side2+side3));

}

}

**public** **class** ProblemSolving {

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

Rectangle obj= **new** Rectangle(2.0f,2.0f);

obj.area();

obj.perimeter();

Square obj1= **new** Square(2);

obj1.area();

obj1.perimeter();

Circle obj2= **new** Circle(2);

obj2.area();

obj2.perimeter();

Triangle obj3= **new** Triangle(5.0f,5.0f,2.0f,2.0f,5.0f);

obj3.area();

obj3.perimeter();

}

}