

Assignment in CompileTime Polymorphism

Create a Java program that demonstrates static polymorphism through method overloading. Define a class called `ShapeAreaCalculator` with the following methods:

- `calculateArea(int sideLength)` - calculates and prints the area of a square.
- `calculateArea(int length, int width)` - calculates and prints the area of a rectangle.
- `calculateArea(double radius)` - calculates and prints the area of a circle.

In the main method of a separate class, instantiate an object of the `ShapeAreaCalculator` class and call each of the `calculateArea` methods with different arguments to showcase the concept of method overloading and static polymorphism.

Example:

```
class ShapeAreaCalculator {
    // Implement calculateArea methods here
}

public class Main {
    public static void main(String[] args) {
        ShapeAreaCalculator calculator = new ShapeAreaCalculator();

        // Call calculateArea methods with different arguments to demonstrate static polymorphism
        calculator.calculateArea(5);
        calculator.calculateArea(3, 7);
        calculator.calculateArea(4.5);
    }
}
```

Your task is to complete the `ShapeAreaCalculator` class with the specified `calculateArea` methods and ensure that the program demonstrates static polymorphism by calling these methods with different argument types.