

Solution Review: Implement and Override the Method

This review provides a detailed analysis on how to solve the 'Implement and Override the Method' challenge.

WE'LL COVER THE FOLLOWING ^

- Solution
- Explanation

Solution

```
// Base Class
class Shape {

    public Shape() {} // Default Constructor

    // Getter Function
    public virtual double ClacArea() {
        return 0;
    }
}

// Derived Class
class Circle : Shape {

    private double _radius;

    public Circle(double radius) { // Constructor
        this._radius = radius;
    }

    // Overridden CalcArea() method which returns the area of Rectangle

    public override double ClacArea() {
        return (this._radius * this._radius) * 3.14;
    }
}

class Demo {

    public static void Main(string[] args) {
        Shape circle = new Circle(2);
    }
}
```

```
        Console.WriteLine(circle.CalcArea());  
    }  
  
}
```



Explanation

The solution is very simple:

- **Line 26 - 28:** The `CalcArea()` method is overridden in the `Circle` class to calculate and return the area of the circle.
- The area is calculated using the conventional formula:

$$pi * r^2$$