Solution Review: Implement Rectangle Class Using the Encapsulatio

This review provides a detailed analysis to solve the 'Implement the Rectangle Class using the Concepts of Encapsulation' challenge.

We'll cover the following

Solution

```
class Rectangle:
    def __init__(self, length, width):
        self.__length = length
        self.__width = width

    def area(self):
        return (self.__length * self.__width)

    def perimeter(self):
        return (2 * (self.__length + self.__width))


obj1 = Rectangle(4, 5)
print("Area is", obj1.area())
print("Perimeter is", obj1.perimeter())

```

Explanation

1. In **lines 3-4**, we defined the initializer for the class and declared private properties, `__length` and `__width` in it.

2. In **line** 7, we defined the method `area()` and returned the product of the

two properties, `__length` and `__width`, in it.

3. In **line 10**, we defined the method `perimeter()` and returned twice the sum of the two properties, `__length` and `__width`, in it.

4. In the main function at **line 13**, we have defined a `Rectangle` class object, `obj1` with properties 4 and 5.

5. In **lines 14-15**, we call the methods `area()` and `perimeter()` and printed their values.