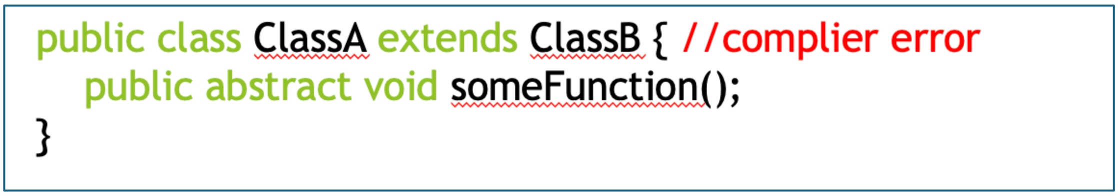
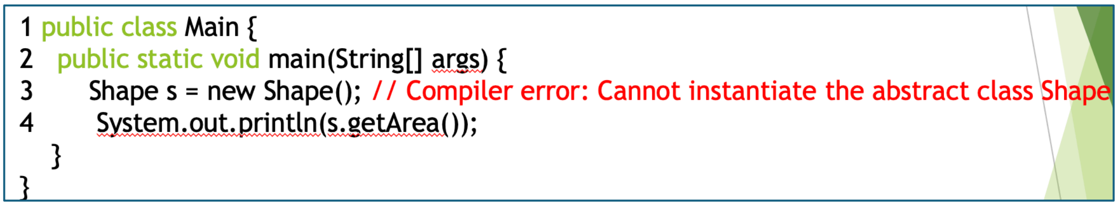
HW6: Chapter 6



```
public class ClassA extends ClassB { //complier error
   public abstract void someFunction();
}
```

Q1: According to the above code how solve this error:

A. The classA must be abstract if have one abstract method or more
B. The classA must be private
C. The method someFunction must be changed to public
D. The method someFunction must be changed to private



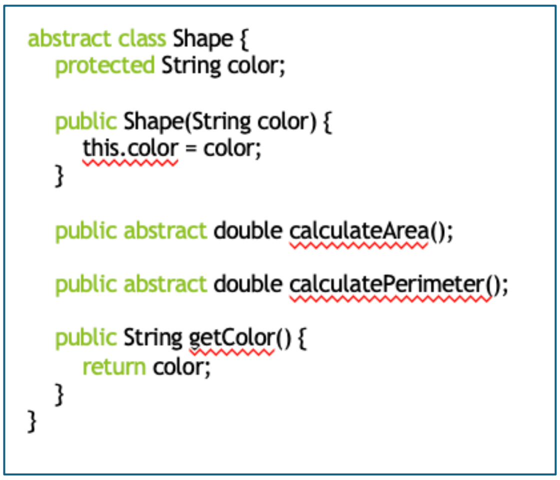
```
1 public class Main {
2   public static void main(String[] args) {
3       Shape s = new Shape(); // Compiler error: Cannot instantiate the abstract class Shape
4        System.out.println(s.getArea());
    }
}
```

Q2: According to the above code the error appears because:

A. Because the class shape private
B. The object can't be created from abstract class directly.

Q3: According to the above code the error in line 3 can be solved as follow:

A. Shape s = new Shape (5, 10)
B. Shape s = new Rectangle (5,10);



```
abstract class Shape {
    protected String color;

    public Shape(String color) {
        this.color = color;
    }

    public abstract double calculateArea();

    public abstract double calculatePerimeter();

    public String getColor() {
        return color;
    }
}
```

Q4: According to the above code class Shape is :

a. Abstract
b. Public
c. Private
d. Package default

Q5: Which of the following statements is true about the Shape class and its subclasses?

A) The `Shape` class cannot have any abstract methods.

B) Subclasses of `Shape` must provide an implementation for the `getColor()` method.

C) The `calculateArea()` method must have a return type of `int`.

D) Subclasses of `Shape` cannot have additional instance variables.