import math  
  
  
# using the math module supplied by python  
  
# class that defines Display\_Info and gives structure for the actual shapes  
class Shape:  
 def \_\_init\_\_(self, Name, Color):  
 self.Name = Name  
 self.Color = Color  
  
 def Display\_Info(self):  
 print(f"{self.Color} {self.Name}")  
  
 def Area(self):  
 pass  
  
 def Perimeter(self):  
 pass  
  
  
# a class that defines the color, name , area , and perimeter of the circle  
class Circle(Shape):  
 def \_\_init\_\_(self, Name, Color, Radius):  
 super().\_\_init\_\_(Name, Color)  
 self.Radius = Radius  
  
 def Area(self):  
 return math.pi \* self.Radius \*\* 2  
  
 def Perimeter(self):  
 return 2 \* math.pi \* self.Radius  
  
  
# a class that defines the color, name , area , length, width and perimeter of the rectangle  
class Rectangle(Shape):  
 def \_\_init\_\_(self, Name, Color, Length, Width):  
 super().\_\_init\_\_(Name, Color)  
 self.Length = Length  
 self.Width = Width  
  
 def Area(self):  
 return self.Length \* self.Width  
  
 def Perimeter(self):  
 return 2 \* (self.Length + self.Width)  
  
  
# a class that defines the color, name , area , three sides and perimeter of the triangle  
class Triangle(Shape):  
 def \_\_init\_\_(self, Name, Color, SideOne, SideTwo, SideThree):  
 super().\_\_init\_\_(Name, Color)  
 self.SideOne = SideOne  
 self.SideTwo = SideTwo  
 self.SideThree = SideThree  
  
 def Area(self):  
 SidesTotal = (self.SideOne + self.SideTwo + self.SideThree) / 2  
 return (math.sqrt(SidesTotal \* (SidesTotal - self.SideOne) \* (SidesTotal - self.SideTwo) \*  
 (SidesTotal - self.SideThree)))  
  
 def Perimeter(self):  
 return self.SideOne + self.SideTwo + self.SideThree  
  
  
# Example Usage  
Circle = Circle("Circle", "Red", 8)  
Rectangle = Rectangle("Rectangle", "Blue", 9, 12)  
Triangle = Triangle("Triangle", "Green", 13, 15, 19)  
  
Shapes = [Circle, Rectangle, Triangle]  
  
for Shape in Shapes:  
 Shape.Display\_Info()  
 print(f"Area: {Shape.Area():.2f}")  
 print(f"Perimeter: {Shape.Perimeter():.2f}")  
 print()

A screenshot of a computer

Description automatically generated