```
using System;

namespace OO_Tasks_Inheritence {
    public abstract class Shape {
        private string _name;
        private string _colour;

        public Shape (string name, string colour)
        {
            _name = name;
            _colour = colour;
        }
    }

    public abstract class Quadrilateral : Shape{
        public const int _numsides = 4;

        public Quadrilateral(string name, string colour) : base(name, colour)
        {
        }
    }

    public class Square : Quadrilateral{
        private double _sidelength;

        public double GetArea()
        {
            double area = Math.Round(_sidelength * _sidelength, 2);
            return area;
        }

        public double GetPerimeter()
        {
            double perimeter = Math.Round(_sidelength * _numsides, 2);
            return perimeter;
        }

        public Square(string name, string colour, double sidelength) : base(name, colour)
        {
            _sidelength = sidelength;
        }
    }

    public class Rectangle : Quadrilateral {
        private double _length;
        private double _width;

        public double GetArea()
        {
            double area = Math.Round(_length * _width, 2);
            return area;
        }

        public double GetPerimeter()
        {
            double perimeter = Math.Round((_length * 2) + (_width * 2), 2);
            return perimeter;
        }

        public Rectangle(string name, string colour, double length, double width) : base(name, colour)
        {
            _length = length;
            _width = width;
        }
    }

    public class Circle : Shape{
        public const double PI = 3.142;
        private double _radius;

        public double GetArea()
        {
            double area = Math.Round(PI * (_radius * _radius), 2);
            return area;
        }

        public double GetCircumference()
        {
            double circumference = Math.Round(2 * PI * _radius, 2);
            return circumference;
        }

        public double GetPerimeter()
        {
            double perimeter = Math.Round(2 * PI * _radius, 2);
            return perimeter;
        }

        public Circle(string name, string colour, double radius) : base(name, colour)
        {
            _radius = radius;
        }
    }


}
```