EXERCISE, INHERITANCE – GETTING STARTED

Exercise 4, Inheritance – getting started

Objective

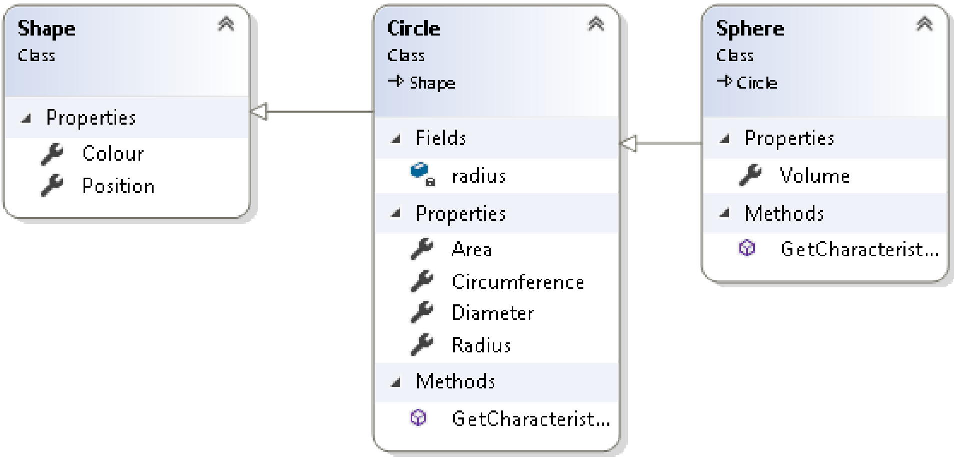
The primary objective for this lab is to enable you to derive new types and to add specialist functionality.

Overview

The lab introduces some of the basic concepts of the inheritance story. As mentioned in the associated session, in order to implement inheritance, you must first have a class that provides the fundamental definition or behavior you need. In this lab we will play about with circular shapes.
This practical will be built on two chapter's time.

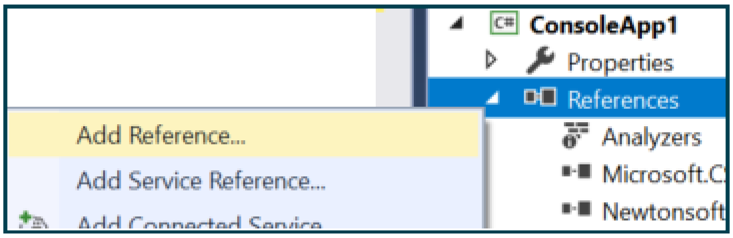
Step by step

1. Create a new Console application called Lab04.
2. Create the following class structure

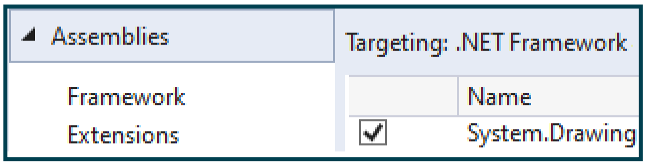


3. As you can see, Circle extends Shape and Sphere extends Circle.
   a. *Position* is of type Point which is a struct with built-in x and y.
   b. *Colour* is of type Color

   Both Point and Color are from the *System.Drawing* namespace but Console apps do not reference *System.Drawing.dll*. You therefore need to add reference to System.Drawing
   (see below).



And then select the reference.



Tip: Use Math.PI to get the value of PI.
You will need this to calculate the area and circumfrance of circle.

Volume of an sphere is calculated as 4/3 * PI * R^3
(R to power of 3).You can use the Math.pow(R,3) fuction or
R * R * R.

4. Create a constructor for Shape to set its colour and position
5. Create property methods for each of the fields (colour, radius…) as indicated in the class diagram above.
6. The GetCharacteristics () method returns a *string* containing all the attributes of the shape. It will be up to the caller how to display this information.

7. Create a few shape types in Main() such as Rectangle, Circle and Sphere.
8. Print the characteristics of the Rectangle, Circle and Sphere objects which you've created.

9. Create a List<Shape> called **shapes** in the main()

10. Add the shapes which you created earlier into the *shapes* list.

11. Create a foreach loop to scroll through each shape and print its colour and position (x,y).

    How does this work?! How can we store a shape like Rectangle in a list of Shapes? All will be revealed in the next chapter.

** End **