

```
// William Kelley
// PollyMoTest.cs
// ITE365 – Lab 10(Polymorphism Main)

using System;
using System.Collections.Generic;

namespace PollyMoTestApp
{
    class MainClass
    {
        public static void Main(string[] args)
        {
            // Polymorphism at work #1: a Rectangle, Triangle and Circle
            // can all be used wherever a Shape is expected. No cast is
            // required because an implicit conversion exists from a derived
            // class to its base class.
            var shapes = new List<Shape>
            {
                new Rectangle(),
                new Triangle(),
                new Circle()
            };
            // Polymorphism at work #2: the virtual method Draw is
            // invoked on each of the derived classes, not the base class.
            foreach (Shape s in shapes)
            {
                s.Draw();
            }
            // Keep the console open in debug mode.
            Console.WriteLine("Press any key to exit.");
            Console.ReadKey();
        }
    }
}
```

```
// William Kelley
// Shape.cs
// ITE365 – Lab 10(Polymorphism Shape File)

using System;

namespace PollyMoTestApp
{
    public class Shape
    {
        public int X { get; private set; }
        public int Y { get; private set; }
        public int Height { get; set; }
        public int Width { get; set; }

        public virtual void Draw()
        {
            Console.WriteLine("Performing base class drawing tasks");
        }
    }

    class Circle : Shape
```

```

    {
        public override void Draw()
        {
            // Code to draw a circle...
            Console.WriteLine("Drawing a circle");
            base.Draw();
        }
    }
    class Rectangle : Shape
    {
        public override void Draw()
        {
            // Code to draw a rectangle...
            Console.WriteLine("Drawing a rectangle");
            base.Draw();
        }
    }
    class Triangle : Shape
    {
        public override void Draw()
        {
            // Code to draw a triangle...
            Console.WriteLine("Drawing a triangle");
            base.Draw();
        }
    }
}

```

Drawing a rectangle
 Performing base class drawing tasks
 Drawing a triangle
 Performing base class drawing tasks
 Drawing a circle
 Performing base class drawing tasks
 Press any key to exit.

