You are tasked with designing a program to model different shapes. Each shape has a unique ID and the ability to calculate its area. Implement the following requirements using classes, inheritance, and interfaces in C#:

```
using System;

// Shape abstract base class
public abstract class Shape
{
    public int ID { get; set; }
    public string Name { get; set; }

    public abstract double CalculateArea();
}

// IDrawable interface
public interface IDrawable
{
    void Draw();
}

// Circle derived class
public class Circle : Shape, IDrawable
{
    public double Radius { get; set; }

    public override double CalculateArea()
    {
        return Math.PI * Radius * Radius;
    }

    public void Draw()
    {
        Console.WriteLine("Drawing a circle");
    }
}

// Rectangle derived class
public class Rectangle : Shape, IDrawable
{
    public double Width { get; set; }
    public double Height { get; set; }

    public override double CalculateArea()
    {
        return Width * Height;
    }

    public void Draw()
    {
        Console.WriteLine("Drawing a rectangle");
    }
}

// Triangle derived class
public class Triangle : Shape
{
    public double Base { get; set; }
    public double Height { get; set; }
```

```
    public override double CalculateArea()
    {
        return 0.5 * Base * Height;
    }
}
public class Program
{
    public static void Main(string[] args)
    {
        // Polymorphism
        Shape[] shapes = new Shape[3];
        shapes[0] = new Circle() { ID = 1, Name = "Circle", Radius = 5 };
        shapes[1] = new Rectangle() { ID = 2, Name = "Rectangle", Width = 4, Height =
6 };
        shapes[2] = new Triangle() { ID = 3, Name = "Triangle", Base = 3, Height =
4 };

        foreach (var shape in shapes)
        {
            double area = shape.CalculateArea();
            Console.WriteLine($"Shape: {shape.Name} (ID: {shape.ID}), Area: {area}");
        }

        Console.WriteLine();

        // Interface
        Circle circle = new Circle() { ID = 4, Name = "Circle", Radius = 7 };
        Rectangle rectangle = new Rectangle() { ID = 5, Name = "Rectangle", Width = 5,
Height = 3 };

        IDrawable drawableCircle = circle;
        IDrawable drawableRectangle = rectangle;

        drawableCircle.Draw();
        drawableRectangle.Draw();
    }
}
```

OUTPUT: