



МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ ИМЕНИ Н.Э. БАУМАНА

Факультет Информатика и системы управления

Кафедра Системы обработки информации и управления (ИУ5)

Базовые компоненты интернет-технологий

Отчет по лабораторной работе №2

Выполнила: Бессонова Ксения Сергеевна

Группа: ИУ5-31Б

Преподаватель: Гапанюк Юрий Евгеньевич

Дата: 18.12.20

Подпись:

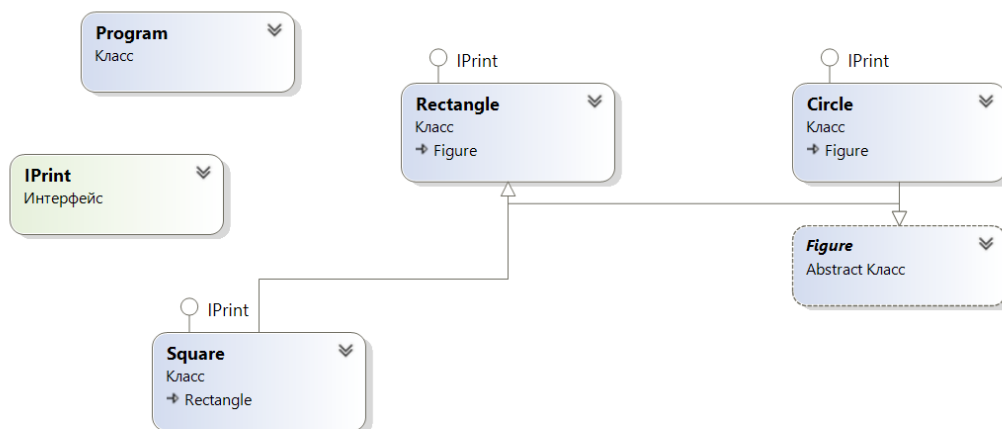
Москва, 2020 г.

## Описание задания:

Разработать программу, реализующую работу с классами.

1. Программа должна быть разработана в виде консольного приложения на языке C#.
2. Абстрактный класс «Геометрическая фигура» содержит виртуальный метод для вычисления площади фигуры.
3. Класс «Прямоугольник» наследуется от «Геометрическая фигура». Ширина и высота объявляются как свойства (property). Класс должен содержать конструктор по параметрам «ширина» и «высота».
4. Класс «Квадрат» наследуется от «Прямоугольник». Класс должен содержать конструктор по длине стороны.
5. Класс «Круг» наследуется от «Геометрическая фигура». Радиус объявляется как свойство (property). Класс должен содержать конструктор по параметру «радиус».
6. Для классов «Прямоугольник», «Квадрат», «Круг» переопределить виртуальный метод `Object.ToString()`, который возвращает в виде строки основные параметры фигуры и ее площадь.
7. Разработать интерфейс `IPrint`. Интерфейс содержит метод `Print()`, который не принимает параметров и возвращает `void`. Для классов «Прямоугольник», «Квадрат», «Круг» реализовать наследование от интерфейса `IPrint`. Переопределяемый метод `Print()` выводит на консоль информацию, возвращаемую переопределенным методом `ToString()`.

## Диаграмма классов:



## Текст программы:

Circle.cs

```

using System;

namespace ConsoleApp3
{
    class Circle : Figure, IPrint
    {
        public Circle(double radius = 0)
        {
            Radius = radius;
        }
        public double Radius { get; set; }

        public override string FigureName => "Круг";

        public override double Area() => Math.PI * Radius * Radius;

        public void Print() => Console.WriteLine(this.ToString());

        public override string ToString()
        {
            return $"{this.FigureName} с площадью {this.Area()} и радиусом {Radius}";
        }
    }
}

```

## Figure.cs

```

using System;
using System.Collections.Generic;
using System.Text;

namespace ConsoleApp3
{
    abstract class Figure
    {
        public abstract string FigureName { get; }
        public abstract double Area();
    }
}

```

## IPrint.cs

```

using System;
using System.Collections.Generic;
using System.Text;

namespace ConsoleApp3
{
    interface IPrint
    {
        void Print(); // выводит в консоль то, что переопределяется методом ToString()
    }
}

```

## Rectangle.cs

```

using System;
using System.Collections.Generic;
using System.Text;

namespace ConsoleApp3
{

```

```

class Rectangle : Figure, IPrint
{
    public Rectangle(double height = 0, double width = 0)
    {
        H = height;
        W = width;
    }
    public double H { get; set; }
    public double W { get; set; }

    public override string FigureName => "Прямоугольник"; // виртуальный метод

    public override double Area()
    {
        return W * H;
    }

    public void Print() => Console.WriteLine(this.ToString());

    public override string ToString()
    {
        return $"{this.FigureName} с площадью {this.Area()} и высотой {this.H}, а
шириной {this.W}";
    }
}

```

## Square.cs

```

using System;
using System.Collections.Generic;
using System.Text;

namespace ConsoleApp3
{
    class Square : Rectangle, IPrint
    {
        public Square(double height = 0) : base(height, height) { }
        public override string FigureName => "Квадрат";
        public new void Print() => Console.WriteLine(this.ToString());

        public override string ToString()
        {
            return $"{this.FigureName} с площадью {this.Area()} и стороной {H}";
        }
    }
}

```

## Program.cs

```

using System;

namespace ConsoleApp3
{
    class Program
    {
        static void Main(string[] args)
        {
            Circle a = new Circle(100);
            Rectangle b = new Rectangle(7, 8);
            Square c = new Square(6);
            a.Print();
            b.Print();
            c.Print();
        }
    }
}

```

```
        Console.ReadKey();  
    }  
}
```

## Результаты программы:

