

Московский Государственный Технический Университет имени Н. Э. Баумана

Базовые компоненты интернет-технологий

Лабораторная работа №2

ИСПОЛНИТЕЛЬ:

ФИО

Болотов Н.А.

Группа РТ5-31

ПРЕПОДАВАТЕЛЬ:

ФИО

Гапанюк Ю. Е.

Москва 2017

Программа лабораторной работы:

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()`.

Код программы:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace _laba
{
    /// <summary>
    /// Figure.
    /// </summary>
    abstract class Figure
    {
        public string Type
        { get; set; }
        public abstract double Area();

        public override string ToString()
        {
            return this.Type + "area " + this.Area().ToString();
        }
    }
    /// <summary>
    /// Print.
    /// </summary>
    interface IPrint
    {
        void Print();
    }
    /// <summary>
    /// Rectangle.
    /// </summary>
    class Rectangle : Figure, IPrint
    {
        double height;
        double width;
        public Rectangle(double ph, double pw)
        {
            this.height = ph;
            this.width = pw;
            this.Type = " rectangle ";
        }
        public override double Area()
        {
            double Result = this.width * this.height;
            return Result;
        }
        public void Print()
        {
            Console.WriteLine(this.ToString());
        }
    }
    /// <summary>
    /// Square.
    /// </summary>
    class Square : Rectangle, IPrint
    {
```

```

        public Square(double size)
            : base(size, size)
        {
            this.Type = " square ";
        }
    }
    /// <summary>
    /// Circle.
    /// </summary>
    class Circle : Figure, IPrint
    {
        double rad;
        public Circle(double pr)
        {
            this.rad = pr;
            this.Type = " Circle ";
        }
        public override double Area()
        {
            double Result = Math.PI * this.rad * this.rad;
            return Result;
        }
        public void Print()
        {
            Console.WriteLine(this.ToString());
        }
    }
    /// <summary>
    /// Main class.
    /// </summary>
    class MainClass
    {
        static double proverka()
        {
            while (true)
            {
                int result;
                result = Convert.ToInt32(Console.ReadLine());
                if (result > 0)
                {
                    return result;
                }
                else
                {
                    Console.WriteLine("Invalid input, Try once more");
                }
            }
        }
        public static void Main(string[] args)
        {
            int a = 0;
            while (a != 1488)
            {
                double B, D;

                int swi = 0;
                Console.WriteLine("1 for Rectangle, 2 for Square, 3 for Circle");
                swi = Convert.ToInt32(Console.ReadLine());
                switch (swi)
                {

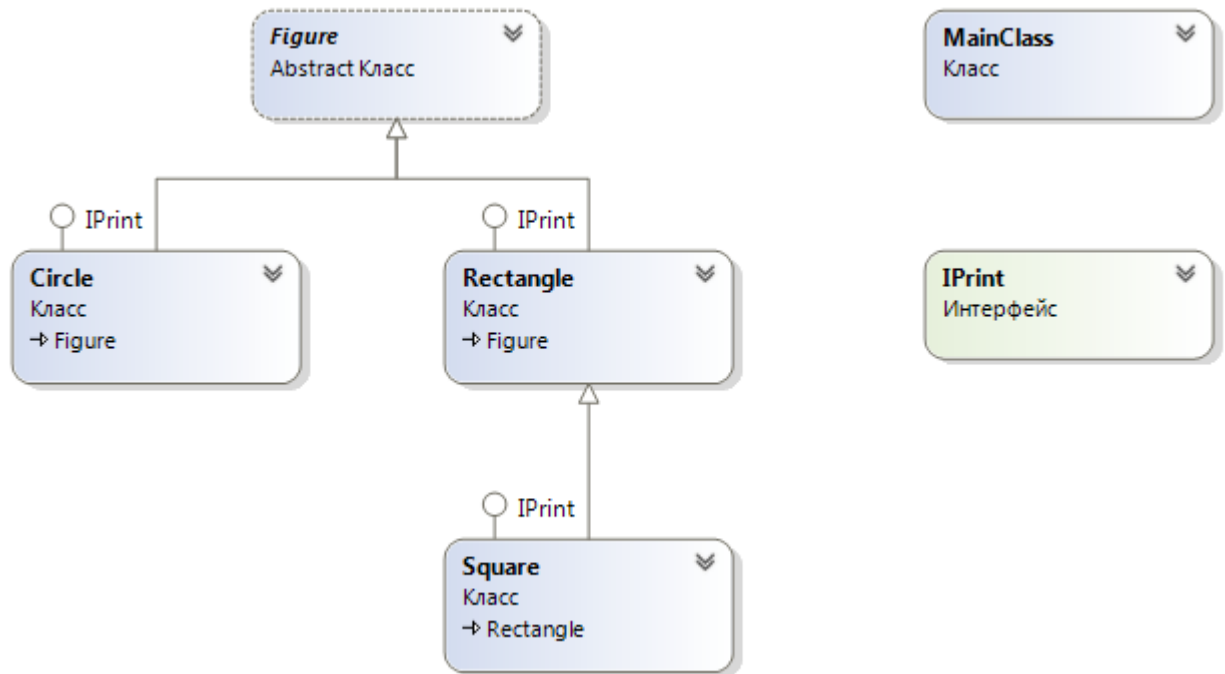
```

```

        case 1:
            Console.WriteLine("Insert wide");
            B = proverka();
            Console.WriteLine("Insert high");
            D = proverka();
            Rectangle rect = new Rectangle(B, D);
            rect.Print();
            break;
        case 2:
            Console.WriteLine("Insert side");
            B = proverka();
            Square sqr = new Square(B);
            sqr.Print();
            break;
        case 3:
            Console.WriteLine("Insert radius");
            D = proverka();
            Circle cir = new Circle(D);
            cir.Print();
            break;
        default:
            Console.WriteLine("Invalid input, Try once more");
            break;
    }
    Console.WriteLine("\n\nEnter any key for continue...");
    Console.ReadKey();
}
}
}
}
}

```

Диаграмма классов:



Пример консольного вывода:

```
Insert wide
20
Insert hight
30
rectangle area 600

Enter any key for continue...
a1 for Rectangle, 2 for Square, 3 for Circle
2
Insert side
10
square area 100

Enter any key for continue...
1 for Rectangle, 2 for Square, 3 for Circle
3
Insert radius
14
Circle area 615,752160103599

Enter any key for continue...
1 for Rectangle, 2 for Square, 3 for Circle
4
Invalid input, Try once more

Enter any key for continue...
11 for Rectangle, 2 for Square, 3 for Circle
1
Insert wide
-30
Invalid input, Try once more
```