**Московский государственный технический**

**университет им. Н. Э. Баумана**

Факультет «Информатика и управление»

Кафедра «Системы обработки информации и управления»

Курс «Базовые компоненты интернет-технологий»

Отчет по лабораторной работе №2

«Работа с классами»

Группа: ИУ5-32Б

Студент: Селедкина А. С.

Преподаватель: Гапанюк Ю. Е.

Москва, 2018 г.

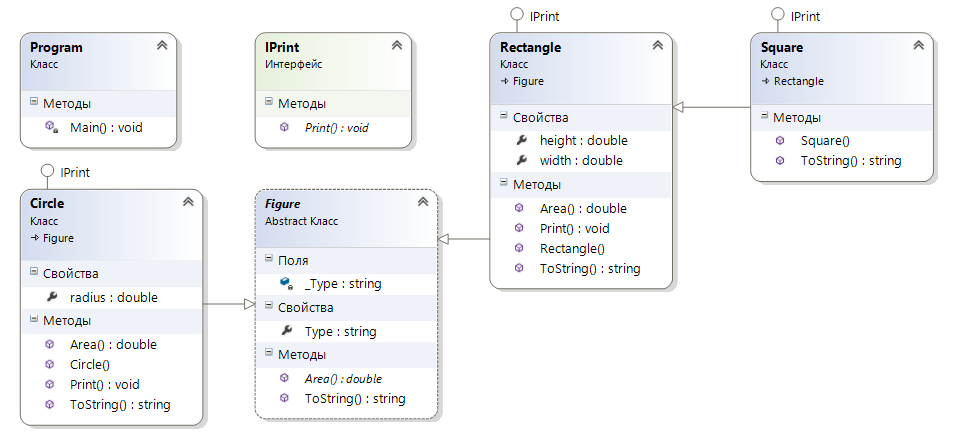
**Описание задания**

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.

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**

Файл **Program.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Lab2

{

class Program

{

static void Main(string[] args)

{

Rectangle rect = new Rectangle(3.5, 8);

Square square = new Square(6.3);

Circle circle = new Circle(2);

rect.Print();

square.Print();

circle.Print();

Console.ReadLine();

}

}

}

Файл **Figure.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Lab2

{

abstract class Figure

{

string \_Type;

public string Type

{

get

{

return this.\_Type;

}

protected set

{

this.\_Type = value;

}

}

public abstract double Area();

public override string ToString()

{

return base.ToString();

}

}

}

Файл **IPrint.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Lab2

{

interface IPrint

{

void Print();

}

}

Файл **Rectangle.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Lab2

{

class Rectangle : Figure, IPrint

{

public double height { get; protected set; }

public double width { get; protected set; }

public Rectangle(double h, double w)

{

this.height = h;

this.width = w;

this.Type = "Прямоугольник";

}

public override double Area()

{

return this.height \* this.width;

}

public override string ToString()

{

return this.Type + " со сторонами " + this.height.ToString() + " и "

+ this.width.ToString() + " и площадью " + this.Area().ToString();

}

public void Print()

{

Console.WriteLine(this.ToString());

}

}

}

Файл **Square.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Lab2

{

class Square : Rectangle, IPrint

{

public Square(double side)

: base(side, side)

{

this.Type = "Квадрат";

}

public override string ToString()

{

return this.Type + " со стороной " + this.height.ToString()

+ " и площадью " + this.Area().ToString();

}

}

}

Файл **Circle.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Lab2

{

class Circle : Figure, IPrint

{

public double radius { get; protected set; }

public Circle(double r)

{

this.radius = r;

this.Type = "Круг";

}

public override double Area()

{

return Math.PI \* this.radius \* this.radius;

}

public override string ToString()

{

return this.Type + " с радиусом " + this.radius.ToString()

+ " и площадью " + this.Area().ToString();

}

public void Print()

{

Console.WriteLine(this.ToString());

}

}

}

**Примеры выполнения программы**

