**Министерство науки и высшего образования Российской Федерации** **Федеральное государственное бюджетное образовательное учреждение высшего образования** **«Московский государственный технический университет** **имени Н. Э. Баумана** **(национальный исследовательский университет)»**

**(МГТУ им. Н.Э. Баумана)**

**Факультет «Радиотехнический»**

**Кафедра «Системы обработки информации и управления»**

Отчет по лабораторной работе №2

по курсу БКИТ

Выполнила:

Студентка Группы РТ5-31Б   
Веденеева Екатерина Ильинична

подпись, дата

Проверил:

подпись, дата

2020г.

1. **Задание**

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().
8. **Код**

using System;

namespace Lab2

{

class Program

{

/// <summary>

/// Абстрактный клас "Геометрическая фгура"

/// </summary>

abstract class Figure

{

//public int area;

public virtual double Calculate\_area()

{

return 0;

}

}

/// <summary>

/// Класс "Прямоугольник", наследник Г.ф.

/// </summary>

class Rectangle : Figure, IPrint

{

public Rectangle(int h, int w)

{

Height = h;

Width = w;

}

private int \_height;

public int Height

{

get { return \_height; }

set { \_height = value; }

}

private int \_width;

public int Width

{

get { return \_width; }

set { \_width = value; }

}

public override double Calculate\_area()

{

return Height \* Width;

}

public override string ToString()

{

return "Прямоугольнк со сторонами " + Height + " и "

+ Width + " площадью " + Calculate\_area();

}

public void Print()

{

Console.WriteLine(ToString());

}

}

/// <summary>

/// Класс "Квадрат", наследник "Прямоугольника"

/// </summary>

class Square : Rectangle, IPrint

{

public Square(int l) : base(l, 0)

{

Height = l;

Width = l;

}

public override string ToString()

{

return "Квадрат со стороной " + Height + " площадью " + Calculate\_area();

}

new public void Print() {

Console.WriteLine(ToString());

}

}

/// <summary>

/// Класс "Круг", наследник Г.Ф.

/// </summary>

class Circle : Figure, IPrint

{

public Circle(int r)

{

Radius = r;

}

private int \_radius;

public int Radius

{

get { return \_radius; }

set { \_radius = value; }

}

public override double Calculate\_area()

{

return 3.14 \* Radius \* Radius;

}

public override string ToString()

{

return "Круг с радусом " + Radius + " площадью " + Calculate\_area();

}

public void Print()

{

Console.WriteLine(ToString());

}

}

class APrint {

public APrint(Rectangle r) {

Console.WriteLine(r.ToString());

}

public APrint(Square s)

{

Console.WriteLine(s.ToString());

}

public APrint(Circle c)

{

Console.WriteLine(c.ToString());

}

}

interface IPrint

{

void Print();

}

static void Main(string[] args)

{

Console.WriteLine("Веденеева Екатерина РТ5-31Б Лабораторная 2");

Rectangle r = new Rectangle(2,4);

Square s = new Square(4);

Circle c = new Circle(1);

r.Print();

s.Print();

c.Print();

int i = Console.Read();

}

}

}

**3. Результат выполнения программы**

