**Отчет по лабораторной работе № 2**

**по курсу «Базовые компоненты интернет технологий»**

Выполнил:

студент группы ИУ5-32Б

Нищук Роман

**Описание задания 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».

1. Класс «Квадрат» наследуется от «Прямоугольник». Класс должен содержать конструктор по длине стороны.
2. Класс «Круг» наследуется от «Геометрическая фигура». Радиус объявляется как свойство (property).Класс должен содержать конструктор по параметру «радиус».
3. Для классов «Прямоугольник», «Квадрат», «Круг» переопределить виртуальный метод Object.ToString(), который возвращает в виде строки основные параметры фигуры и ее площадь.
4. 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().

1. Программно, лабораторная работа должна быть реализована в виде отдельного проекта в рамках единого решения, созданного при выполнении лабораторной работы №1. Все классы и интерфейсы должны быть реализованы как отдельные элементы проекта

**Текст программы на языке C#.**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace Lab\_2

{

class Program

{

static void Main(string[] args)

{

IPrint wer;

Console.Title = "Нищук Роман ИУ5-32Б";

double a, b, c, d;

Console.WriteLine("Введите длину и высоту прямоугольника");

a = double.Parse(Console.ReadLine());

b = double.Parse(Console.ReadLine());

wer = new Rectangle(a, b);

wer.Print();

Console.ReadKey();

Console.WriteLine("Введите сторону квадрата");

c = double.Parse(Console.ReadLine());

wer = new Square(c);

wer.Print();

Console.ReadKey();

Console.WriteLine("Введите радиус круга");

d = double.Parse(Console.ReadLine());

wer = new Circle(d);

wer.Print();

Console.ReadKey();

System.Environment.Exit(0);

Console.ReadLine();

}

}

// Абстрактный класс «Геометрическая фигура»

abstract class Figure

{

public string Type

{

get

{

return this.\_Type;

}

protected set

{

this.\_Type = value;

}

}

string \_Type;

public abstract double Area();

public override string ToString()

{

return this.\_Type + this.Area().ToString();

}

}

// Интерфуйс IPrint

interface IPrint

{

void Print();

}

// Класс круг

class Circle : Figure, IPrint

{

public Circle(double radius)

{

\_radius = radius;

}

private double \_radius = 0;

public double radius

{

get { return \_radius; }

set { \_radius = value; }

}

public override double Area()

{

return Math.PI \* \_radius \* \_radius;

}

public override string ToString()

{

return "Площадь круга = " + Area().ToString();

}

public void Print()

{

Console.WriteLine(this);

}

}

// Класс прямоугольник

class Rectangle : Figure, IPrint

{

public Rectangle(double height1, double width1)

{

\_height = height1;

\_width = width1;

}

private double \_height = 0;

public double height

{

get { return \_height; }

set { \_height = value; }

}

private double \_width = 0;

public double width

{

get { return \_width; }

set { \_width = value; }

}

public override double Area()

{

return \_width \* \_height;

}

public override string ToString()

{

return "Площадь прямоуголька = " + Area().ToString();

}

public void Print()

{

Console.WriteLine(this);

}

}

// Класс квадрат

class Square : Rectangle

{

public Square(double height1) : base(height1, height1) { }

public override double Area()

{

return height \* height;

}

public override string ToString()

{

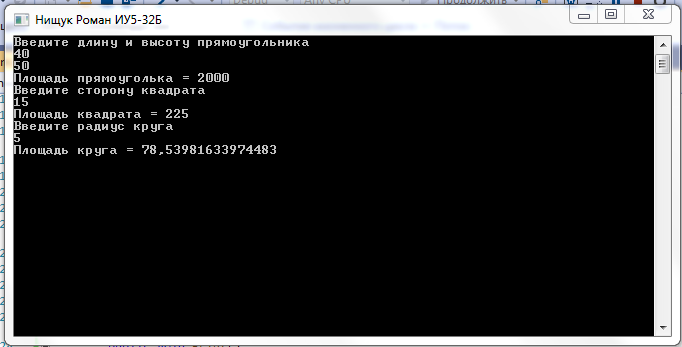
return "Площадь квадрата = " + Area().ToString();

}

}

}

**Результаты выполнения программы:**



**Диаграмма классов:**

Изображение выглядит как снимок экрана

Автоматически созданное описание