**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Базовые компоненты интернет - технологий»

Отчет по лабораторной работе № 2

Выполнила Проверил

Студент Воронцова Алина Гапанюк Ю.Е.

Группа ИУ5-32Б

Дата и подпись: Дата и подпись:

г.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**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Reflection.Metadata.Ecma335;

using System.Text;

using static System.Math;

namespace lab2.Vorontsova.Alina

{

interface IPrint

{

void Print();

}

class platform

{

static double Prov(string k)

{

double ans;

bool F = Double.TryParse(k, out ans);

while (!F)

{

Console.WriteLine("Введите корректное значение");

k = Console.ReadLine();

F = Double.TryParse(k, out ans);

}

return ans;

}

static void Main(string[] args)

{

int k;

do

{

Console.WriteLine("\_\_\_\_\_\_\_\_\_Выберите цифру\_\_\_\_\_\_\_\_");

Console.WriteLine("1.Прямоугольник");

Console.WriteLine("2.Квадрат");

Console.WriteLine("3.Круг");

Console.WriteLine("0.Выход");

k = (int)Prov(Console.ReadLine());

if (k == 1)

{

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

double h = System.Convert.ToDouble(Console.ReadLine());

double w = System.Convert.ToDouble(Console.ReadLine());

Rectangle rect = new Rectangle(h, w);

rect.Print();

}

else if (k == 2)

{

Console.WriteLine("Введите сторону квадрата");

double h = System.Convert.ToDouble(Console.ReadLine());

Square sqr = new Square(h);

sqr.Print();

}

else if (k == 3)

{

Console.WriteLine("Введите радиус круга");

double h = System.Convert.ToDouble(Console.ReadLine());

Circle cir = new Circle(h);

cir.Print();

}

} while (k != 0);

}

}

abstract class Figure

{

public abstract double Area();

}

class Rectangle : Figure, IPrint

{

public double Height { get; set; }

public double Width { get; set; }

public Rectangle(double height, double width)

{

this.Height = height;

this.Width = width;

}

public override double Area()

{

return this.Height \* this.Width;

}

public override string ToString()

{

return $"Площадь прямоугольника с высотой {Height} и шириной {Width} равна {Area()}";

}

public void Print()

{

Console.WriteLine(this.ToString());

}

}

class Circle : Figure, IPrint

{

public double Radius { get; set; }

public Circle(double radius)

{

this.Radius = radius;

}

public override double Area()

{

return this.Radius \* this.Radius \* PI;

}

public override string ToString()

{

return $"Площадь круга с радиусом {Radius} равна {Area()}";

}

public void Print()

{

Console.WriteLine(this.ToString());

}

}

class Square : Rectangle, IPrint

{

public double side;

public Square(double side) : base(side, side)

{

this.side = side;

}

public override double Area()

{

return this.side \* this.side;

}

public override string ToString()

{

return $"Площадь квадрата со стороной {side} равна {Area()}";

}

public void Print()

{

Console.WriteLine(this.ToString());

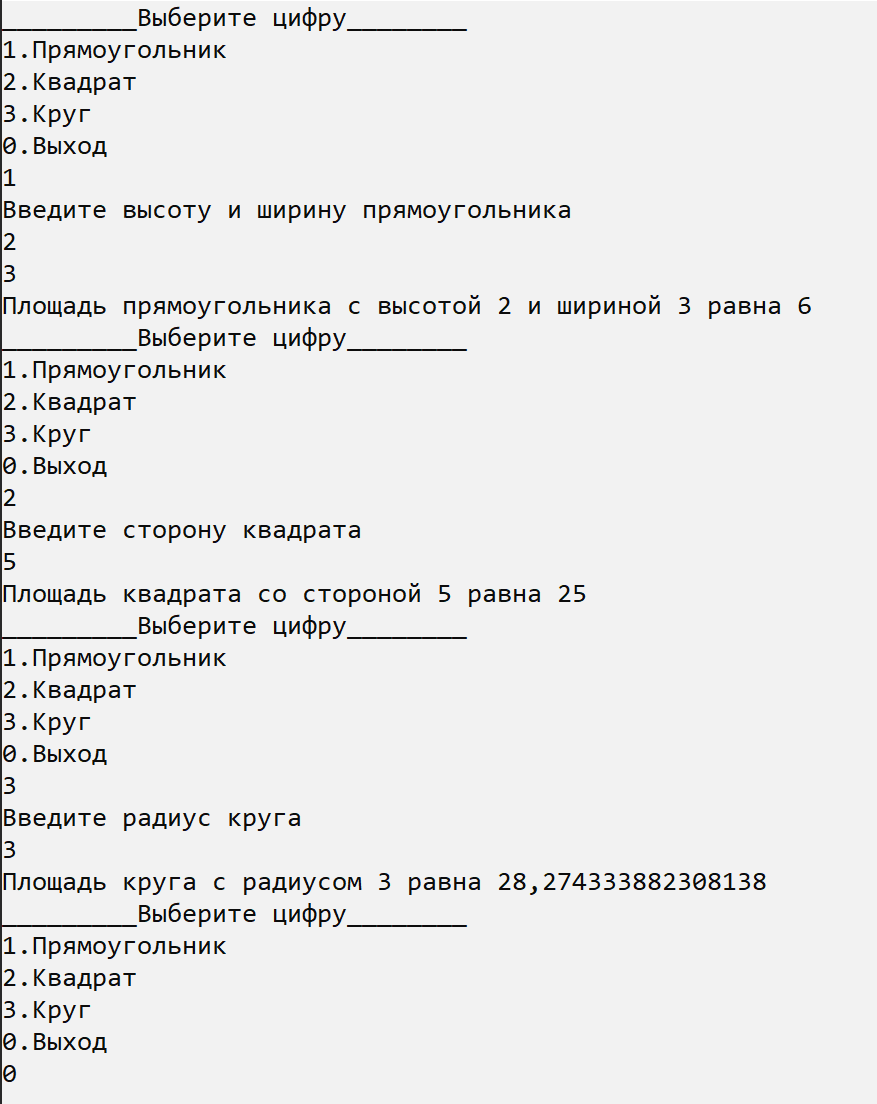
}

}

}

Результаты работы программы

1)В случае, когда вводимые значения верны и реальны



Для удобства выбора фигуры, чью площадь необходимо сосчитать, было реализовано «Меню». В данном примере приведены все посчитанные площади фигур, а в меню при выборе цифры введены корректные значения.

Для контроля ввода значений при выборе пункта Меню в программе была проведена работа с исключениями. Далее приведены скриншоты работы программы, в которой при выборе пункта меню были введены некорректные значения:

А) Некорректные численные значения – числа превышают допустимо-возможные

Б) Вместе цифр были введены буквы

2) Когда значения вводятся некорректно при выборе меню

