|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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 «Информатика и системы управления»**

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

Отчет по Лабораторной работе № 2

по дисциплине «Базовые компоненты интернет-технологий»

Выполнил:

студент(ка) группы ИУ5-34Б Бузин Е.М

подпись, дата

Проверил:

к.т.н., Ю.Е. Гапанюк

подпись, дата

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().

**Текст программы**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using static System.Math;

namespace LAB2

{

class Figures

{

abstract class GeometricFigure

{

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 + " площадью " +

this.Area().ToString();

}

}

interface IPrint

{

void Print();

}

class Rectangle : GeometricFigure, IPrint

{

double height;

double width;

public double Height

{

set

{

if (this.height >= 0)

{

this.height = value;

}

else

{

Console.WriteLine("Высота не может быть отрицательной");

}

}

get

{

return this.height;

}

}

public double Width

{

set

{

if (this.width >= 0)

{

this.width = value;

}

else

{

Console.WriteLine("Ширина не может быть отрицательной");

}

}

get

{

return this.width;

}

}

public Rectangle(double h, double w)

{

this.height = h;

this.Height = h;

this.width = w;

this.Width = w;

this.Type = "Прямоугольник";

}

public override double Area()

{

double Result = this.width \* this.height;

if (this.width < 0 || this.height < 0)

{

Console.WriteLine("Площадь не может быть отрицательной");

Result = 0;

}

return Result;

}

public void Print()

{

Console.WriteLine(this.ToString());

}

}

class Square : Rectangle, IPrint

{

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

{

this.Type = "Квадрат";

}

}

class Circle : GeometricFigure, IPrint

{

double radius;

public double Rad

{

set

{

if (this.radius >= 0)

{

this.radius = value;

}

else

{

Console.WriteLine("Радиус не может быть отрицательным");

}

}

get

{

return this.radius;

}

}

public Circle(double pr)

{

this.radius = pr;

this.Rad = pr;

this.Type = "Круг";

}

public override double Area()

{

double Result = Math.PI \* this.radius \* this.radius;

if (this.radius < 0)

{

Console.WriteLine("Площадь не может быть отрицательной");

Result = 0;

}

return Result;

}

public void Print()

{

Console.WriteLine(this.ToString());

}

}

static void Main()

{

bool f = true;

do

{

Console.WriteLine("1 - Ввести данные для прямоугольника:");

Console.WriteLine("2 - Ввести данные для квадрата:");

Console.WriteLine("3 - Ввести данные для круга:");

Console.WriteLine("4 - Очистить консоль");

Console.WriteLine("5 - Выход");

int num = Convert.ToInt32(Console.ReadLine());

if (num <= 0 || num > 5)

{

Console.WriteLine("Повторите ввод пункта от 1 до 5:");

}

switch (num)

{

case 1:

double a, b;

Console.WriteLine("Введите высоту прямоугольника:");

a = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Введите ширину прямоугольника:");

b = Convert.ToDouble(Console.ReadLine());

Rectangle rect = new Rectangle(a, b);

rect.Print();

break;

case 2:

double с;

Console.WriteLine("Введите сторону квадрата:");

с = Convert.ToDouble(Console.ReadLine());

Square square = new Square(с);

square.Print();

break;

case 3:

double radius;

Console.WriteLine("Введите радиус круга:");

radius = Convert.ToDouble(Console.ReadLine());

Circle circle = new Circle(radius);

circle.Print();

break;

case 4:

Console.Clear();

break;

case 5:

f = false;

break;

}

} while (f);

}

}

}

**Экранные формы с примерами выполнения программы**

 

 