**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Радиотехнический»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Базовые компоненты интернет технологий»

Отчет по лабораторной работе №2

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы РТ5-31Б |  | преподаватель каф. ИУ5 |
| Алиев Тимур |  | Гапанюк Ю.Е. |
| Подпись и дата: |  | Подпись и дата: |

# 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;

namespace lab2

{

abstract class GeometricFigure

{

// Название фигуры

public string name;

// Вычисление площади фигуры

virtual public double Area(){

return(0);

}

}

public interface IPrint

{

public void Print();

}

class Square: GeometricFigure, IPrint

{

private double length;

public Square(double l)

{

name = "Square";

length = l;

}

public override double Area()

{

return (length\*length);

}

public override string ToString() {

return ($"name - {name}, length = {length}, area = {this.Area()}");

}

public void Print()

{

Console.WriteLine(this.ToString());

}

}

class Circle : GeometricFigure, IPrint

{

private double radius;

public Circle (double r)

{

name = "Circle";

radius = r;

}

public override double Area()

{

return (3.14 \* radius \* radius);

}

public override string ToString()

{

return ($"name - {name}, radius = {radius}, area = {this.Area()}");

}

void IPrint.Print()

{

Console.WriteLine(this.Area());

}

public void Print()

{

Console.WriteLine(this.ToString());

}

}

class Rectangle : GeometricFigure, IPrint

{

private double length;

private double height;

public Rectangle(double l, double h)

{

name = "Rectangle";

length = l;

height = h;

}

public override double Area()

{

return (height \* length);

}

public override string ToString()

{

return ($"name - {name}, length = {length}, height = {height}, area = {this.Area()}");

}

void IPrint.Print()

{

Console.WriteLine(this.Area());

}

public void Print()

{

Console.WriteLine(this.ToString());

}

}

class Program

{

static void Main(string[] args)

{

Rectangle rectangle = new Rectangle(5,4);

Circle circle = new Circle(5);

Square square = new Square(6);

rectangle.Print();

circle.Print();

square.Print();

}

}

}

# Результат работы программы

