**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

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы ИУ5-33Б |  | преподаватель каф. ИУ5 |
| Кузнецов Григорий |  |  |
| Подпись и дата: |  | Подпись и дата: |

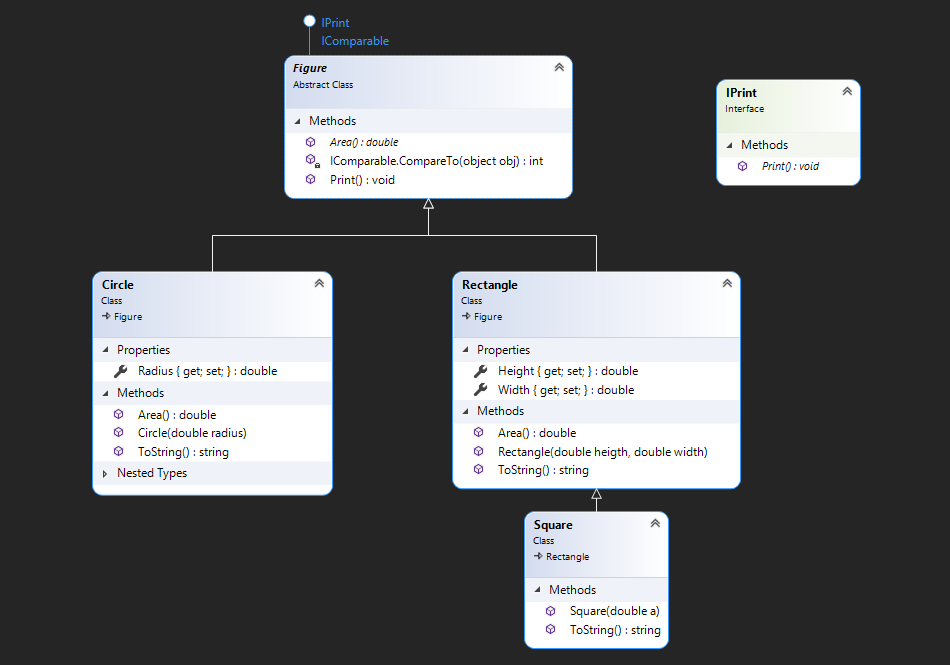
Москва, 2020 г.

**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;

namespace Lab2

{

public interface IPrint

{

void Print();

}

public abstract class Figure : IPrint, IComparable

{

public abstract double Area();

public void Print() =>

Console.WriteLine(ToString());

int IComparable.CompareTo(object obj)

{

var figure = obj as Figure;

return Math.Sign(Area() - figure.Area());

}

}

public class Rectangle : Figure

{

public double Height { get; set; }

public double Width { get; set; }

public Rectangle(double heigth, double width) =>

(Height, Width) = (heigth, width);

public override double Area() =>

Height \* Width;

public override string ToString() =>

$"Прямоугольник: Ширина: {Width}; Высота: {Height}; Площадь: {Area()}";

}

public class Square : Rectangle

{

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

public override string ToString() =>

$"Квадрат: Длина стороны: {Width}; Площадь: {Area()}";

}

public class Circle : Figure

{

public double Radius { get; set; }

public Circle(double radius) =>

Radius = radius;

public override double Area() =>

Math.PI \* Radius \* Radius;

public override string ToString() =>

$"Круг: Радиус: {Radius}; Площадь: {Area()}";

public static class Program

{

public static void Main()

{

var circle = new Circle(3d);

circle.Print();

var square = new Square(2d);

square.Print();

var rectangle = new Rectangle(1d, 3d);

rectangle.Print();

}

}

}

}

**Пример выполнения программы**

