**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе №4

«Работа с классами в C#»

|  |  |  |
| --- | --- | --- |
| Выполнила: |  | Проверил: |
| студент группы ИУ5-33Б |  | преподаватель каф. ИУ5 |
| Хабленко Инна |  | Гапанюк Ю. Е. |
|  |  |  |

Москва, 2023 г.

# Задание

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

abstract class Geometric\_Shape

{

public abstract double GetArea(); // абстрактный метод для получения площади

public override string ToString()

{

return $"Площадь: {GetArea()}";

}

}

class Rectangle: Geometric\_Shape

{

private double width\_ { get; set; }

private double height\_ { get; set; }

public Rectangle(int width, int height)

{

width\_ = width;

height\_ = height;

}

public override double GetArea() => width\_ \* height\_; // переопрелеление получения площади

public void Print()

{

Console.WriteLine($"Треугольник: Ширина = {width\_}, Высота = {height\_}. {ToString()}");

}

}

class Squaree : Geometric\_Shape

{

private double side\_ { get; set; }

public Squaree(int side)

{

side\_ = side;

}

public override double GetArea() => Math.Pow(side\_, 2); // переопрелеление получения площади

public void Print()

{

Console.WriteLine($"Квадрат: Сторона = {side\_}. {ToString()}");

}

}

class Circle : Geometric\_Shape

{

private double radius\_ { get; set; }

public Circle(int radius)

{

radius\_ = radius;

}

public override double GetArea() => Math.PI \* Math.Pow(radius\_, 2); // переопрелеление получения площади

public void Print()

{

Console.WriteLine($"Круг: Радиус = {radius\_}. {ToString()}");

}

}

class Program

{

static void Main(string[] args)

{

Rectangle rectangle = new Rectangle(5, 10); // Создание объекта прямоугольник

rectangle.Print();

Squaree squaree = new Squaree(5); // Создание объекта квадрат

squaree.Print();

Circle circle = new Circle(10); // Создание объекта прямоугольника

circle.Print();

}

}

# Результат

