Міністерство освіти і науки України

Львівський національний університет імені Івана Франка

Факультет електроніки та комп’ютерних технологій

Кафедра системного проектування

**Звіт**

про виконання лабораторної роботи № 4

«Успадковування»

Виконав:

Студент групи ФеП-11

Лебідь Роман

Перевірив:

Щербак. С. С.

Львів 2020

**Хід роботи:**

**Завдання 1.** Реалізувати базовий клас Figure з абстрактним методом Draw(). Створити класи Square та Rectangle (унаслідовані від Figure) в яких перегружена імплементація методу Draw() виводить на консоль назву класу в якому даний метод реалізований.



using System;

using System.Globalization;

using System.Threading;

namespace ConsoleApp1

{

abstract class Figure

{

public abstract void Draw();

}

class Square : Figure

{

public override void Draw()

{

Console.WriteLine("Квадрат");

}

}

class Rectangle : Figure

{

public override void Draw()

{

Console.WriteLine("Прямокутник");

}

}

class Program

{

static void Main(string[] args)

{

Rectangle rect = new Rectangle();

Square sq = new Square();

sq.Draw();

rect.Draw();

}

}

}

**Завдання** **2**. Використовуючи попереднє завдання добавити в клас Figure readonly проперті X та Y. Також, в клас Figure добавити параметризований конструктор що установляє значення в цих пропертях. Поправити унаслідовані класи щоб вони коректно працювали викликаючи базовий конструктор.



using System;

using System.Globalization;

using System.Threading;

namespace ConsoleApp1

{

abstract class Figure

{

public float X { get; private set; }

public float Y { get; private set; }

protected Figure(float X, float Y)

{

this.X = X;

this.Y = Y;

}

public abstract void Draw();

}

class Square : Figure

{

public Square(float X, float Y) : base(X, Y)

{

}

public override void Draw()

{

Console.WriteLine("Квадрат");

}

}

class Rectangle : Figure

{

public Rectangle(float X, float Y) : base(X, Y)

{

}

public override void Draw()

{

Console.WriteLine("Прямокутник");

}

}

class Program

{

static void Main(string[] args)

{

Rectangle rect = new Rectangle(2.1f, 3.4f);

Square sq = new Square(1.1f, 5.3f);

sq.Draw();

rect.Draw();

}

}

}

**Завдання 3.** Використовуючи попереднє завдання змінити абстрактний метод Draw() на віртуальний і реалізувати його аналогічно як у класах Square та Rectangle. Переконатись що об'єкт класу Figure тепер може бути ініціалізований і виклики методу Draw() на об'єктах створених класів приведуть до виклику трьох різних реалізацій.



using System;

using System.Globalization;

using System.Threading;

namespace ConsoleApp1

{

class Figure

{

public float X { get; private set; }

public float Y { get; private set; }

public Figure(float X, float Y)

{

this.X = X;

this.Y = Y;

}

public virtual void Draw()

{

Console.WriteLine("Фiгура");

}

}

class Square : Figure

{

public Square(float X, float Y) : base(X, Y)

{

}

public override void Draw()

{

Console.WriteLine("Квадрат");

}

}

class Rectangle : Figure

{

public Rectangle(float X, float Y) : base(X, Y)

{

}

public override void Draw()

{

Console.WriteLine("Прямокутник");

}

}

class Program

{

static void Main(string[] args)

{

Rectangle rect = new Rectangle(2.1f, 3.4f);

Square sq = new Square(1.1f, 5.3f);

Figure fig = new Figure(2.1f, 1.1f);

sq.Draw();

rect.Draw();

fig.Draw();

}

}

}

**Завдання 4.** Використовуючи попереднє завдання винести метод Draw() у інтерфейс IDrawable. Створити метод DrawAll(params IDrawable[] array) в який передати об'єкти створених класів і в циклі викликати метод Draw() для кожного з них. Переконатись що були викликані коректні реалізації методу Draw() для кожного з класів.



using System;

using System.Collections.Generic;

using System.Text;

namespace CSharp

{

interface IDrawable

{

void Draw();

}

class Figure : IDrawable

{

public float X { get; private set; }

public float Y { get; private set; }

public Figure(float X, float Y)

{

this.X = X;

this.Y = Y;

}

void IDrawable.Draw()

{

Console.WriteLine("Фiгура");

}

}

class Square : Figure, IDrawable

{

public Square(float X, float Y) : base(X, Y)

{

}

void IDrawable.Draw()

{

Console.WriteLine("Квадрат");

}

}

class Rectangle : Figure, IDrawable

{

public Rectangle(float X, float Y) : base(X, Y)

{

}

void IDrawable.Draw()

{

Console.WriteLine("Прямокутник");

}

}

class Program

{

static void DrawAll(params IDrawable[] array)

{

foreach (IDrawable drawable in array)

{

drawable.Draw();

}

}

static void Main(string[] args)

{

Rectangle rect = new Rectangle(2.1f, 3.4f);

Square sq = new Square(1.1f, 5.3f);

Figure fig = new Figure(2.1f, 1.1f);

IDrawable rectDrawable = (IDrawable)rect;

IDrawable sqDrawable = (IDrawable)sq;

IDrawable figDrawable = (IDrawable)fig;

DrawAll(rectDrawable, sqDrawable, figDrawable);

}

}

}