Министерство образования Республики Беларусь

Оршанский колледж ВГУ имени П.М.Машерова

**Отчет**

по учебной практике по программированию

«Модуль 4. Задания по интерфейсам»

Выполнил учащийся Казаченко Н.С.

группы 3ПОИС23 04.10.2025г.

Проверил Алейников М.А.

04.10.2025г.

Орша, 2025

**Цель:** разработать Модуль 4.

**Содержание работы**

**Модуль 4 Задания по интерфейсам:**

1. Создайте интерфейс "Фигура" с методами для вычисления площади и периметра. Затем реализуйте этот интерфейс в классах геометрических фигур (круг, прямоугольник, треугольник).

2. Разработайте приложение для учета продуктов в магазине с использованием интерфейсов. Создайте интерфейс "Товар" с методами для определения стоимости и остатка товара на складе. Реализуйте этот интерфейс в классах различных товаров.

3. Создайте систему учета студентов в университете с помощью интерфейсов. Создайте интерфейс "Студент" с методами для определения среднего балла и получения информации о курсе. Реализуйте этот интерфейс в классах студентов разных курсов.

4. Реализуйте приложение для работы с библиотекой книг с использованием интерфейсов. Создайте интерфейс "Книга" с методами для проверки доступности и выдачи книги. Реализуйте этот интерфейс в классах различных книг.

5. Создайте приложение для рисования на холсте с использованием интерфейсов. Создайте интерфейс "Рисунок" с методами для рисования линий, кругов и прямоугольников. Реализуйте этот интерфейс в классе для работы с холстом.

**Ход выполнения работы**

**Выполнения практических заданий**

**Задание 1**

Для выполнения программы требовалось написать программу, которая будет с помощью интерфейса вычислять периметр и площадь геометрических фигур.

Код написанной программы представлен в листинге 4.1.

Результат выполнения программы представлен на рисунке 4.1.

Листинг 4.1. Код программы

using System;

namespace \_4.\_0.\_0

{

// Интерфейс Фигура

public interface IShape

{

double CalculateArea();

double CalculatePerimeter();

string GetShapeName();

}

// Класс Круг

public class Circle : IShape

{

public double Radius { get; set; }

public Circle(double radius)

{

Radius = radius;

}

public double CalculateArea()

{

return Math.PI \* Radius \* Radius;

}

public double CalculatePerimeter()

{

return 2 \* Math.PI \* Radius;

}

public string GetShapeName()

{

return "Круг";

}

}

// Класс Прямоугольник

public class Rectangle : IShape

{

public double Width { get; set; }

public double Height { get; set; }

public Rectangle(double width, double height)

{

Width = width;

Height = height;

}

public double CalculateArea()

{

return Width \* Height;

}

public double CalculatePerimeter()

{

return 2 \* (Width + Height);

}

public string GetShapeName()

{

return "Прямоугольник";

}

}

// Класс Треугольник

public class Triangle : IShape

{

public double SideA { get; set; }

public double SideB { get; set; }

public double SideC { get; set; }

public Triangle(double sideA, double sideB, double sideC)

{

SideA = sideA;

SideB = sideB;

SideC = sideC;

}

public double CalculateArea()

{

// Используем формулу Герона

double p = CalculatePerimeter() / 2;

return Math.Sqrt(p \* (p - SideA) \* (p - SideB) \* (p - SideC));

}

public double CalculatePerimeter()

{

return SideA + SideB + SideC;

}

public string GetShapeName()

{

return "Треугольник";

}

public bool IsValidTriangle()

{

return SideA + SideB > SideC &&

SideA + SideC > SideB &&

SideB + SideC > SideA;

}

}

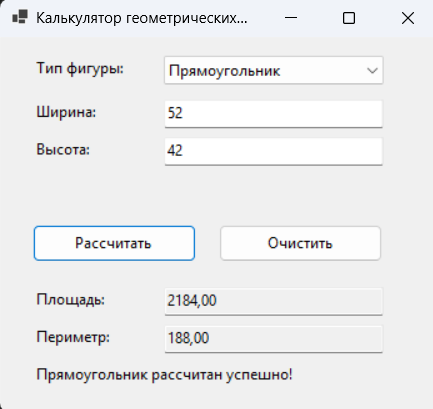


Рисунок 4.1 – Результат выполнения программы

**Задание 2**

Для выполнения задания требовалось написать программу, которая с помощью, которой будет реализовываться учет продуктов в магазине.

Код программы представлен в листинге 4.2.

Результат выполнения программы представлен на рисунке 4.2.

Листинг 4.2. Код программы.

namespace StoreInventoryApp

{

partial class Form1

{

private System.ComponentModel.IContainer components = null;

protected override void Dispose(bool disposing)

{

if (disposing && (components != null))

{

components.Dispose();

}

base.Dispose(disposing);

}

#region Windows Form Designer generated code

private void InitializeComponent()

{

this.cmbProductType = new System.Windows.Forms.ComboBox();

this.lblProductType = new System.Windows.Forms.Label();

this.lblName = new System.Windows.Forms.Label();

this.txtName = new System.Windows.Forms.TextBox();

this.lblCode = new System.Windows.Forms.Label();

this.txtCode = new System.Windows.Forms.TextBox();

this.lblPrice = new System.Windows.Forms.Label();

this.txtPrice = new System.Windows.Forms.TextBox();

this.lblStock = new System.Windows.Forms.Label();

this.txtStock = new System.Windows.Forms.TextBox();

this.lblExtra1 = new System.Windows.Forms.Label();

this.txtExtra1 = new System.Windows.Forms.TextBox();

this.lblExtra2 = new System.Windows.Forms.Label();

this.txtExtra2 = new System.Windows.Forms.TextBox();

this.lblExtra3 = new System.Windows.Forms.Label();

this.txtExtra3 = new System.Windows.Forms.TextBox();

this.btnAddProduct = new System.Windows.Forms.Button();

this.lstProducts = new System.Windows.Forms.ListBox();

this.lblTotalProducts = new System.Windows.Forms.Label();

this.lblTotalValue = new System.Windows.Forms.Label();

this.btnRemoveProduct = new System.Windows.Forms.Button();

this.txtSearch = new System.Windows.Forms.TextBox();

this.lblSearch = new System.Windows.Forms.Label();

this.btnSearch = new System.Windows.Forms.Button();

this.btnShowLowStock = new System.Windows.Forms.Button();

this.dtpExpiryDate = new System.Windows.Forms.DateTimePicker();

this.chkPerishable = new System.Windows.Forms.CheckBox();

this.lblTotalItems = new System.Windows.Forms.Label();

this.btnShowAll = new System.Windows.Forms.Button();

this.SuspendLayout();

//

// cmbProductType

//

this.cmbProductType.DropDownStyle = System.Windows.Forms.ComboBoxStyle.DropDownList;

this.cmbProductType.FormattingEnabled = true;

this.cmbProductType.Location = new System.Drawing.Point(120, 20);

this.cmbProductType.Name = "cmbProductType";

this.cmbProductType.Size = new System.Drawing.Size(200, 28);

this.cmbProductType.TabIndex = 0;

this.cmbProductType.SelectedIndexChanged += new System.EventHandler(this.cmbProductType\_SelectedIndexChanged);

//

// lblProductType

//

this.lblProductType.AutoSize = true;

this.lblProductType.Location = new System.Drawing.Point(20, 23);

this.lblProductType.Name = "lblProductType";

this.lblProductType.Size = new System.Drawing.Size(90, 20);

this.lblProductType.TabIndex = 1;

this.lblProductType.Text = "Тип товара:";

//

// lblName

//

this.lblName.AutoSize = true;

this.lblName.Location = new System.Drawing.Point(20, 60);

this.lblName.Name = "lblName";

this.lblName.Size = new System.Drawing.Size(130, 20);

this.lblName.TabIndex = 2;

this.lblName.Text = "Название товара:";

//

// txtName

//

this.txtName.Location = new System.Drawing.Point(160, 57);

this.txtName.Name = "txtName";

this.txtName.Size = new System.Drawing.Size(200, 27);

this.txtName.TabIndex = 3;

//

// lblCode

//

this.lblCode.AutoSize = true;

this.lblCode.Location = new System.Drawing.Point(20, 95);

this.lblCode.Name = "lblCode";

this.lblCode.Size = new System.Drawing.Size(100, 20);

this.lblCode.TabIndex = 4;

this.lblCode.Text = "Код товара:";

//

// txtCode

//

this.txtCode.Location = new System.Drawing.Point(160, 92);

this.txtCode.Name = "txtCode";

this.txtCode.Size = new System.Drawing.Size(200, 27);

this.txtCode.TabIndex = 5;

//

// lblPrice

//

this.lblPrice.AutoSize = true;

this.lblPrice.Location = new System.Drawing.Point(20, 130);

this.lblPrice.Name = "lblPrice";

this.lblPrice.Size = new System.Drawing.Size(116, 20);

this.lblPrice.TabIndex = 6;

this.lblPrice.Text = "Базовая цена:";

//

// txtPrice

//

this.txtPrice.Location = new System.Drawing.Point(160, 127);

this.txtPrice.Name = "txtPrice";

this.txtPrice.Size = new System.Drawing.Size(200, 27);

this.txtPrice.TabIndex = 7;

this.txtPrice.Text = "0";

//

// lblStock

//

this.lblStock.AutoSize = true;

this.lblStock.Location = new System.Drawing.Point(20, 165);

this.lblStock.Name = "lblStock";

this.lblStock.Size = new System.Drawing.Size(134, 20);

this.lblStock.TabIndex = 8;

this.lblStock.Text = "Количество (шт):";

//

// txtStock

//

this.txtStock.Location = new System.Drawing.Point(160, 162);

this.txtStock.Name = "txtStock";

this.txtStock.Size = new System.Drawing.Size(200, 27);

this.txtStock.TabIndex = 9;

this.txtStock.Text = "0";

//

// lblExtra1

//

this.lblExtra1.AutoSize = true;

this.lblExtra1.Location = new System.Drawing.Point(20, 200);

this.lblExtra1.Name = "lblExtra1";

this.lblExtra1.Size = new System.Drawing.Size(85, 20);

this.lblExtra1.TabIndex = 10;

this.lblExtra1.Text = "Доп. инфо:";

//

// txtExtra1

//

this.txtExtra1.Location = new System.Drawing.Point(160, 197);

this.txtExtra1.Name = "txtExtra1";

this.txtExtra1.Size = new System.Drawing.Size(200, 27);

this.txtExtra1.TabIndex = 11;

//

// lblExtra2

//

this.lblExtra2.AutoSize = true;

this.lblExtra2.Location = new System.Drawing.Point(20, 235);

this.lblExtra2.Name = "lblExtra2";

this.lblExtra2.Size = new System.Drawing.Size(85, 20);

this.lblExtra2.TabIndex = 12;

this.lblExtra2.Text = "Доп. инфо:";

//

// txtExtra2

//

this.txtExtra2.Location = new System.Drawing.Point(160, 232);

this.txtExtra2.Name = "txtExtra2";

this.txtExtra2.Size = new System.Drawing.Size(200, 27);

this.txtExtra2.TabIndex = 13;

//

// lblExtra3

//

this.lblExtra3.AutoSize = true;

this.lblExtra3.Location = new System.Drawing.Point(20, 270);

this.lblExtra3.Name = "lblExtra3";

this.lblExtra3.Size = new System.Drawing.Size(85, 20);

this.lblExtra3.TabIndex = 14;

this.lblExtra3.Text = "Доп. инфо:";

//

// txtExtra3

//

this.txtExtra3.Location = new System.Drawing.Point(160, 267);

this.txtExtra3.Name = "txtExtra3";

this.txtExtra3.Size = new System.Drawing.Size(200, 27);

this.txtExtra3.TabIndex = 15;

//

// btnAddProduct

//

this.btnAddProduct.Location = new System.Drawing.Point(20, 350);

this.btnAddProduct.Name = "btnAddProduct";

this.btnAddProduct.Size = new System.Drawing.Size(150, 40);

this.btnAddProduct.TabIndex = 16;

this.btnAddProduct.Text = "Добавить товар";

this.btnAddProduct.UseVisualStyleBackColor = true;

this.btnAddProduct.Click += new System.EventHandler(this.btnAddProduct\_Click);

//

// lstProducts

//

this.lstProducts.FormattingEnabled = true;

this.lstProducts.ItemHeight = 20;

this.lstProducts.Location = new System.Drawing.Point(400, 50);

this.lstProducts.Name = "lstProducts";

this.lstProducts.Size = new System.Drawing.Size(600, 344);

this.lstProducts.TabIndex = 17;

//

// lblTotalProducts

//

this.lblTotalProducts.AutoSize = true;

this.lblTotalProducts.Location = new System.Drawing.Point(400, 20);

this.lblTotalProducts.Name = "lblTotalProducts";

this.lblTotalProducts.Size = new System.Drawing.Size(85, 20);

this.lblTotalProducts.TabIndex = 18;

this.lblTotalProducts.Text = "Товаров: 0";

//

// lblTotalValue

//

this.lblTotalValue.AutoSize = true;

this.lblTotalValue.Location = new System.Drawing.Point(600, 20);

this.lblTotalValue.Name = "lblTotalValue";

this.lblTotalValue.Size = new System.Drawing.Size(143, 20);

this.lblTotalValue.TabIndex = 19;

this.lblTotalValue.Text = "Общая стоимость:";

//

// btnRemoveProduct

//

this.btnRemoveProduct.Location = new System.Drawing.Point(180, 350);

this.btnRemoveProduct.Name = "btnRemoveProduct";

this.btnRemoveProduct.Size = new System.Drawing.Size(150, 40);

this.btnRemoveProduct.TabIndex = 20;

this.btnRemoveProduct.Text = "Удалить товар";

this.btnRemoveProduct.UseVisualStyleBackColor = true;

this.btnRemoveProduct.Click += new System.EventHandler(this.btnRemoveProduct\_Click);

//

// txtSearch

//

this.txtSearch.Location = new System.Drawing.Point(400, 410);

this.txtSearch.Name = "txtSearch";

this.txtSearch.Size = new System.Drawing.Size(200, 27);

this.txtSearch.TabIndex = 21;

this.txtSearch.TextChanged += new System.EventHandler(this.txtSearch\_TextChanged);

//

// lblSearch

//

this.lblSearch.AutoSize = true;

this.lblSearch.Location = new System.Drawing.Point(400, 387);

this.lblSearch.Name = "lblSearch";

this.lblSearch.Size = new System.Drawing.Size(106, 20);

this.lblSearch.TabIndex = 22;

this.lblSearch.Text = "Поиск товара:";

//

// btnSearch

//

this.btnSearch.Location = new System.Drawing.Point(610, 410);

this.btnSearch.Name = "btnSearch";

this.btnSearch.Size = new System.Drawing.Size(100, 27);

this.btnSearch.TabIndex = 23;

this.btnSearch.Text = "Найти";

this.btnSearch.UseVisualStyleBackColor = true;

this.btnSearch.Click += new System.EventHandler(this.btnSearch\_Click);

//

// btnShowLowStock

//

this.btnShowLowStock.Location = new System.Drawing.Point(720, 410);

this.btnShowLowStock.Name = "btnShowLowStock";

this.btnShowLowStock.Size = new System.Drawing.Size(130, 27);

this.btnShowLowStock.TabIndex = 24;

this.btnShowLowStock.Text = "Низкий запас";

this.btnShowLowStock.UseVisualStyleBackColor = true;

this.btnShowLowStock.Click += new System.EventHandler(this.btnShowLowStock\_Click);

//

// dtpExpiryDate

//

this.dtpExpiryDate.Location = new System.Drawing.Point(160, 197);

this.dtpExpiryDate.Name = "dtpExpiryDate";

this.dtpExpiryDate.Size = new System.Drawing.Size(200, 27);

this.dtpExpiryDate.TabIndex = 25;

this.dtpExpiryDate.Value = DateTime.Now.AddDays(7);

//

// chkPerishable

//

this.chkPerishable.AutoSize = true;

this.chkPerishable.Location = new System.Drawing.Point(160, 230);

this.chkPerishable.Name = "chkPerishable";

this.chkPerishable.Size = new System.Drawing.Size(15, 14);

this.chkPerishable.TabIndex = 26;

this.chkPerishable.UseVisualStyleBackColor = true;

//

// lblTotalItems

//

this.lblTotalItems.AutoSize = true;

this.lblTotalItems.Location = new System.Drawing.Point(500, 20);

this.lblTotalItems.Name = "lblTotalItems";

this.lblTotalItems.Size = new System.Drawing.Size(75, 20);

this.lblTotalItems.TabIndex = 27;

this.lblTotalItems.Text = "Единиц: 0";

//

// btnShowAll

//

this.btnShowAll.Location = new System.Drawing.Point(860, 410);

this.btnShowAll.Name = "btnShowAll";

this.btnShowAll.Size = new System.Drawing.Size(140, 27);

this.btnShowAll.TabIndex = 28;

this.btnShowAll.Text = "Показать все";

this.btnShowAll.UseVisualStyleBackColor = true;

this.btnShowAll.Click += new System.EventHandler(this.btnShowAll\_Click);

//

// Form1

//

this.AutoScaleDimensions = new System.Drawing.SizeF(8F, 20F);

this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;

this.ClientSize = new System.Drawing.Size(1022, 453);

this.Controls.Add(this.btnShowAll);

this.Controls.Add(this.lblTotalItems);

this.Controls.Add(this.chkPerishable);

this.Controls.Add(this.dtpExpiryDate);

this.Controls.Add(this.btnShowLowStock);

this.Controls.Add(this.btnSearch);

this.Controls.Add(this.lblSearch);

this.Controls.Add(this.txtSearch);

this.Controls.Add(this.btnRemoveProduct);

this.Controls.Add(this.lblTotalValue);

this.Controls.Add(this.lblTotalProducts);

this.Controls.Add(this.lstProducts);

this.Controls.Add(this.btnAddProduct);

this.Controls.Add(this.txtExtra3);

this.Controls.Add(this.lblExtra3);

this.Controls.Add(this.txtExtra2);

this.Controls.Add(this.lblExtra2);

this.Controls.Add(this.txtExtra1);

this.Controls.Add(this.lblExtra1);

this.Controls.Add(this.txtStock);

this.Controls.Add(this.lblStock);

this.Controls.Add(this.txtPrice);

this.Controls.Add(this.lblPrice);

this.Controls.Add(this.txtCode);

this.Controls.Add(this.lblCode);

this.Controls.Add(this.txtName);

this.Controls.Add(this.lblName);

this.Controls.Add(this.lblProductType);

this.Controls.Add(this.cmbProductType);

this.Name = "Form1";

this.Text = "Система учета товаров в магазине";

this.ResumeLayout(false);

this.PerformLayout();

}

#endregion

private System.Windows.Forms.ComboBox cmbProductType;

private System.Windows.Forms.Label lblProductType;

private System.Windows.Forms.Label lblName;

private System.Windows.Forms.TextBox txtName;

private System.Windows.Forms.Label lblCode;

private System.Windows.Forms.TextBox txtCode;

private System.Windows.Forms.Label lblPrice;

private System.Windows.Forms.TextBox txtPrice;

private System.Windows.Forms.Label lblStock;

private System.Windows.Forms.TextBox txtStock;

private System.Windows.Forms.Label lblExtra1;

private System.Windows.Forms.TextBox txtExtra1;

private System.Windows.Forms.Label lblExtra2;

private System.Windows.Forms.TextBox txtExtra2;

private System.Windows.Forms.Label lblExtra3;

private System.Windows.Forms.TextBox txtExtra3;

private System.Windows.Forms.Button btnAddProduct;

private System.Windows.Forms.ListBox lstProducts;

private System.Windows.Forms.Label lblTotalProducts;

private System.Windows.Forms.Label lblTotalValue;

private System.Windows.Forms.Button btnRemoveProduct;

private System.Windows.Forms.TextBox txtSearch;

private System.Windows.Forms.Label lblSearch;

private System.Windows.Forms.Button btnSearch;

private System.Windows.Forms.Button btnShowLowStock;

private System.Windows.Forms.DateTimePicker dtpExpiryDate;

private System.Windows.Forms.CheckBox chkPerishable;

private System.Windows.Forms.Label lblTotalItems;

private System.Windows.Forms.Button btnShowAll;

}

}

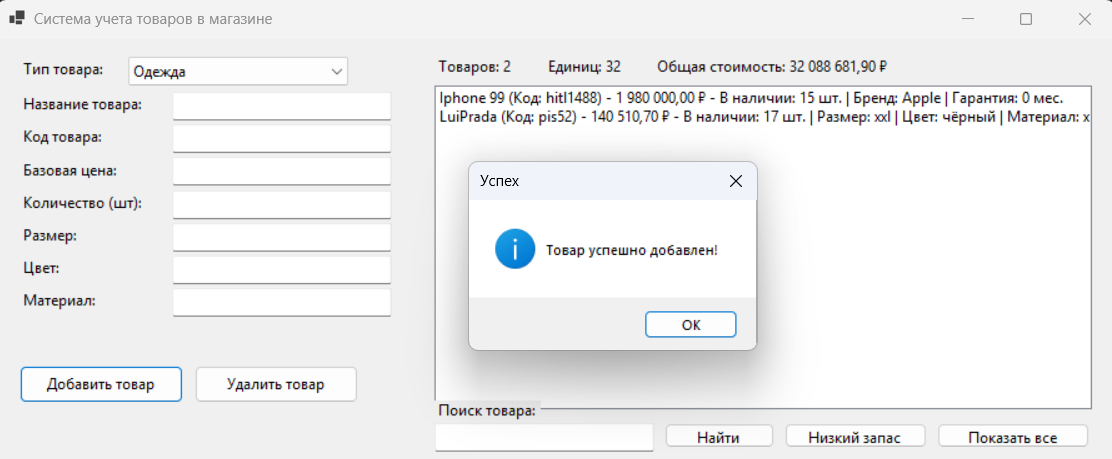


Рисунок 4.2 – Результат учета товаров

**Задание 3.**

Для выполнения задания требовалось написать программу, которая будет организовывать систему учета студентов в университете.

Код программы представлен на рисунке 4.3.

Результат выполнения программы представлен на рисунке 4.3.

Листинг 4.3.Код программы

namespace \_4.3

{

public interface IStudent

{

double GetAverageGrade(); // Определение среднего балла

string GetCourseInfo(); // Получение информации о курсе

}

// Студент 1 курса

public class FirstYearStudent : IStudent

{

private string name;

private double averageGrade;

private string faculty;

public FirstYearStudent(string name, double averageGrade, string faculty)

{

this.name = name;

this.averageGrade = averageGrade;

this.faculty = faculty;

}

public double GetAverageGrade()

{

return averageGrade;

}

public string GetCourseInfo()

{

return $"Студент 1 курса: {name}, Факультет: {faculty}";

}

}

// Студент 2 курса

public class SecondYearStudent : IStudent

{

private string name;

private double averageGrade;

private string specialization;

public SecondYearStudent(string name, double averageGrade, string specialization)

{

this.name = name;

this.averageGrade = averageGrade;

this.specialization = specialization;

}

public double GetAverageGrade()

{

return averageGrade;

}

public string GetCourseInfo()

{

return $"Студент 2 курса: {name}, Специализация: {specialization}";

}

}

// Студент 3 курса

public class ThirdYearStudent : IStudent

{

private string name;

private double averageGrade;

private string researchTopic;

public ThirdYearStudent(string name, double averageGrade, string researchTopic)

{

this.name = name;

this.averageGrade = averageGrade;

this.researchTopic = researchTopic;

}

public double GetAverageGrade()

{

return averageGrade;

}

public string GetCourseInfo()

{

return $"Студент 3 курса: {name}, Тема исследования: {researchTopic}";

}

}

// Студент 4 курса

public class FourthYearStudent : IStudent

{

private string name;

private double averageGrade;

private string diplomaTopic;

public FourthYearStudent(string name, double averageGrade, string diplomaTopic)

{

this.name = name;

this.averageGrade = averageGrade;

this.diplomaTopic = diplomaTopic;

}

public double GetAverageGrade()

{

return averageGrade;

}

public string GetCourseInfo()

{

return $"Студент 4 курса: {name}, Тема диплома: {diplomaTopic}";

}

}

// Демонстрация работы

class Program

{

static void Main()

{

// Создаем студентов разных курсов

List<IStudent> students = new List<IStudent>

{

new FirstYearStudent("Иван Иванов", 4.2, "Информатика"),

new SecondYearStudent("Петр Петров", 4.7, "Программная инженерия"),

new ThirdYearStudent("Мария Сидорова", 4.9, "Машинное обучение"),

new FourthYearStudent("Анна Козлова", 4.5, "Разработка мобильных приложений")

};

// Выводим информацию о студентах

Console.WriteLine("Система учета студентов университета:");

Console.WriteLine("======================================");

foreach (IStudent student in students)

{

string courseInfo = student.GetCourseInfo();

double averageGrade = student.GetAverageGrade();

Console.WriteLine($"{courseInfo}");

Console.WriteLine($"Средний балл: {averageGrade:F1}");

Console.WriteLine("------------------------");

}

// Подсчитываем общий средний балл

Console.WriteLine("\nОбщая статистика:");

double totalAverage = 0;

foreach (IStudent student in students)

{

totalAverage += student.GetAverageGrade();

}

double overallAverage = totalAverage / students.Count;

Console.WriteLine($"Средний балл всех студентов: {overallAverage:F1}");

}

}

}

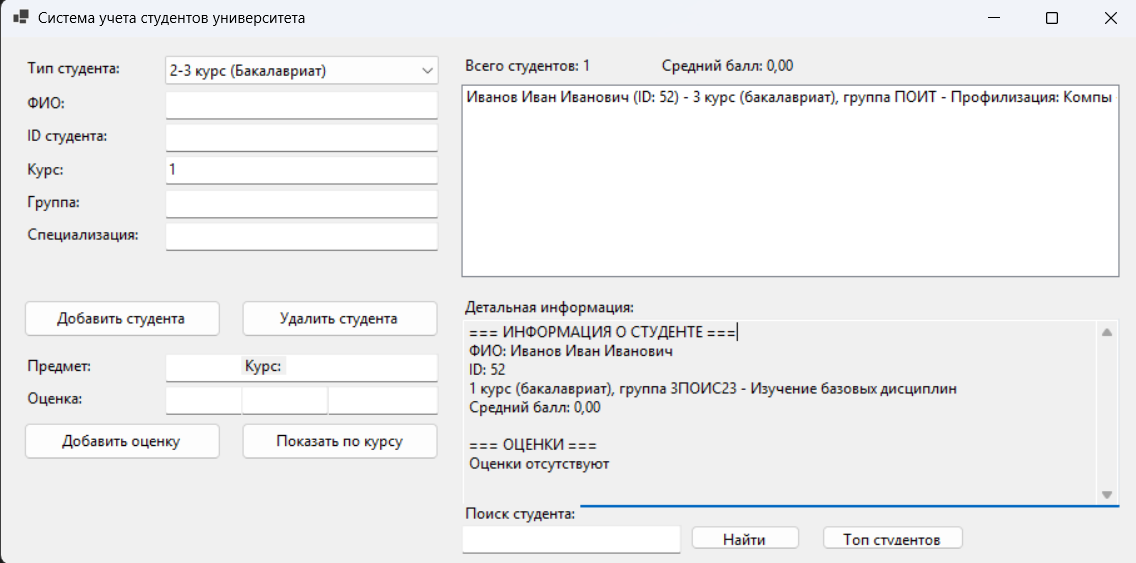


Рисунок 4.3 – Результат выполнения программы

**Задание 4.**

Для выполнения программы требуется написать программу для учета книг в библиотеке с использованием интерфейсов.

Код написанной программы представлен в листинге 4.4.

Результат выполнения программы представлен на рисунке 4.4

Листинге 4.4. Код программы

namespace \_4.\_4

{

public interface IBook

{

bool IsAvailable();

void BorrowBook();

}

public class FictionBook : IBook

{

private string title;

private string author;

private bool isAvailable;

public FictionBook(string title, string author)

{

this.title = title;

this.author = author;

this.isAvailable = true;

}

public bool IsAvailable()

{

return isAvailable;

}

public void BorrowBook()

{

if (isAvailable)

{

isAvailable = false;

Console.WriteLine($"Книга \"{title}\" автора {author} выдана читателю");

}

else

{

Console.WriteLine($"Книга \"{title}\" уже выдана");

}

}

}

public class ScientificBook : IBook

{

private string title;

private string field;

private bool isAvailable;

public ScientificBook(string title, string field)

{

this.title = title;

this.field = field;

this.isAvailable = true;

}

public bool IsAvailable()

{

return isAvailable;

}

public void BorrowBook()

{

if (isAvailable)

{

isAvailable = false;

Console.WriteLine($"Научная книга \"{title}\" (область: {field}) выдана читателю");

}

else

{

Console.WriteLine($"Научная книга \"{title}\" уже выдана");

}

}

}

public class Textbook : IBook

{

private string title;

private string subject;

private bool isAvailable;

public Textbook(string title, string subject)

{

this.title = title;

this.subject = subject;

this.isAvailable = true;

}

public bool IsAvailable()

{

return isAvailable;

}

public void BorrowBook()

{

if (isAvailable)

{

isAvailable = false;

Console.WriteLine($"Учебник \"{title}\" (предмет: {subject}) выдан читателю");

}

else

{

Console.WriteLine($"Учебник \"{title}\" уже выдан");

}

}

}

class Program

{

static void Main()

{

List<IBook> books = new List<IBook>

{

new FictionBook("Преступление и наказание", "Ф.М. Достоевский"),

new ScientificBook("Краткая история времени", "Физика"),

new Textbook("Основы программирования", "Информатика"),

new FictionBook("Мастер и Маргарита", "М.А. Булгаков")

};

Console.WriteLine("Библиотека книг:");

Console.WriteLine("================");

for (int i = 0; i < books.Count; i++)

{

string status = books[i].IsAvailable() ? "Доступна" : "Выдана";

Console.WriteLine($"{i + 1}. {status}");

}

Console.WriteLine("\nПроцесс выдачи книг:");

Console.WriteLine("====================");

books[0].BorrowBook();

books[2].BorrowBook();

books[0].BorrowBook();

Console.WriteLine("\nОбновленная информация о книгах:");

Console.WriteLine("================================");

for (int i = 0; i < books.Count; i++)

{

string status = books[i].IsAvailable() ? "Доступна" : "Выдана";

Console.WriteLine($"{i + 1}. {status}");

}

Console.WriteLine("\nСтатистика библиотеки:");

Console.WriteLine("======================");

int availableCount = 0;

foreach (IBook book in books)

{

if (book.IsAvailable())

availableCount++;

}

Console.WriteLine($"Доступно книг: {availableCount} из {books.Count}");

}

}

}

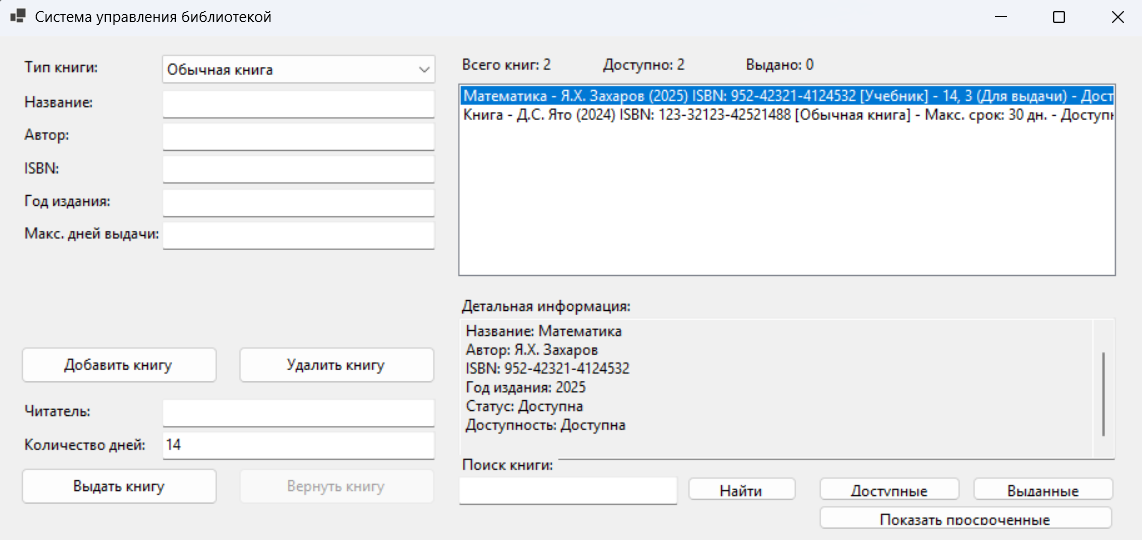


Рисунок 4.4 – Результат выполнения программы.

**Задание 5.**

Для выполнения программы требовалось написать программу, которая позволяет рисовать на холсте с использованием интерфейсов.

Код программы представлен в листинге 4.5.

Результат выполнения программы представлен на рисунке 4.5.

Листинг 4.5. Код программы.

namespace \_4.5

{

public interface IDrawing

{

void DrawLine(int x1, int y1, int x2, int y2);

void DrawCircle(int x, int y, int radius);

void DrawRectangle(int x, int y, int width, int height);

}

public class Canvas : IDrawing

{

private List<string> shapes;

public Canvas()

{

shapes = new List<string>();

Console.WriteLine("Холст создан");

}

public void DrawLine(int x1, int y1, int x2, int y2)

{

string line = $"Линия от ({x1},{y1}) до ({x2},{y2})";

shapes.Add(line);

Console.WriteLine($"Нарисована: {line}");

}

public void DrawCircle(int x, int y, int radius)

{

string circle = $"Круг в точке ({x},{y}) радиусом {radius}";

shapes.Add(circle);

Console.WriteLine($"Нарисован: {circle}");

}

public void DrawRectangle(int x, int y, int width, int height)

{

string rectangle = $"Прямоугольник в ({x},{y}) размером {width}x{height}";

shapes.Add(rectangle);

Console.WriteLine($"Нарисован: {rectangle}");

}

public void DisplayAllShapes()

{

Console.WriteLine("\nВсе фигуры на холсте:");

Console.WriteLine("======================");

if (shapes.Count == 0)

{

Console.WriteLine("Холст пуст");

return;

}

for (int i = 0; i < shapes.Count; i++)

{

Console.WriteLine($"{i + 1}. {shapes[i]}");

}

}

public void ClearCanvas()

{

shapes.Clear();

Console.WriteLine("Холст очищен");

}

}

class Program

{

static void Main()

{

Canvas canvas = new Canvas();

Console.WriteLine("\nПроцесс рисования:");

Console.WriteLine("===================");

IDrawing drawing = canvas;

drawing.DrawLine(0, 0, 100, 100);

drawing.DrawCircle(50, 50, 25);

drawing.DrawRectangle(10, 10, 80, 60);

drawing.DrawLine(100, 100, 200, 50);

drawing.DrawCircle(150, 75, 30);

canvas.DisplayAllShapes();

Console.WriteLine("\nОчистка холста:");

Console.WriteLine("================");

canvas.ClearCanvas();

canvas.DisplayAllShapes();

Console.WriteLine("\nРисуем заново:");

Console.WriteLine("==============");04=

drawing.DrawRectangle(5, 5, 50, 30);

drawing.DrawCircle(25, 25, 15);

canvas.DisplayAllShapes();

}

}

}

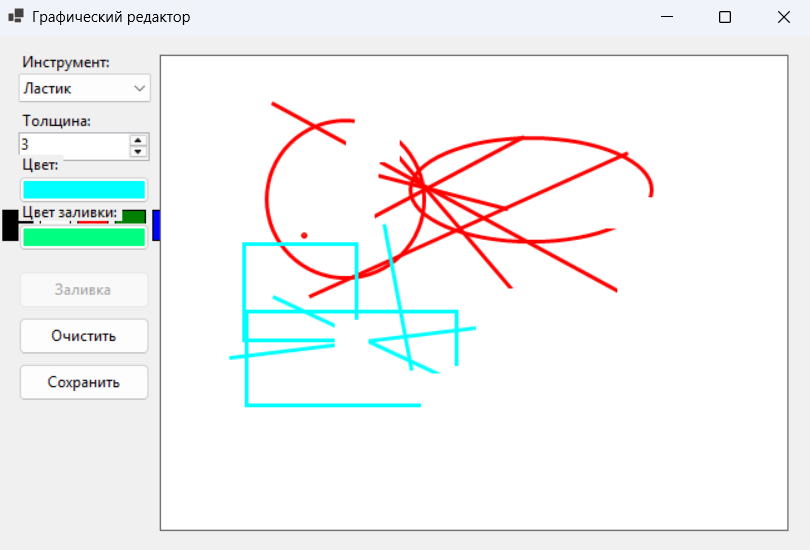


Рисунок 4.5 – Результат выполнения программы

**Вывод:** в ходе выполнения практического задания был разработан модуль 4.