**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ**

**ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ**

**УНИВЕРСИТЕТ ИМЕНИ П.О.СУХОГО**

Факультет автоматизированных и информационных систем

Кафедра «Информационные технологии»

Дисциплина «Объектно-ориентированное проектирование и программирование»

ОТЧЕТ

Лабораторная работа № 1

Работа с файловой системой. Потоки ввода/вывода

Выполнил студент

Группы ИТД-21

Чайдаков И.М.

Принял преподаватель

Башаримов Ю. С.

Гомель 2025

Работа с файловой системой. Потоки ввода/вывода

**Цель работы:** опробовать различные метода инкапсуляции в разработке и использовать разные подходы к организации доступа.

**Задание:** Реализовать учет выполняемых упражнений на тренировке по

фитнесcу, необходимо:

Вывести количество выполненных упражнений за указанный день;

За указанный период по каждому виду упражнений, вывести их общее количество выполнений;

Вывести наиболее популярное упражнение за указанный день.

**Ход выполнения задания:**

**1.** Создаем класс Fitness который будет осуществлять работу со списками упражнений (см. Приложение А).

**2.** Описываем модульные тесты для верификации класса (см. Приложение B).

**3.** Создаем графический интерфейс для пользовательского доступа к функциям класса (см. Приложение С).

**4.** Запустим исполняемый файл и увидим перед собой созданный интерфейс с набором функций (Рисунок 1).

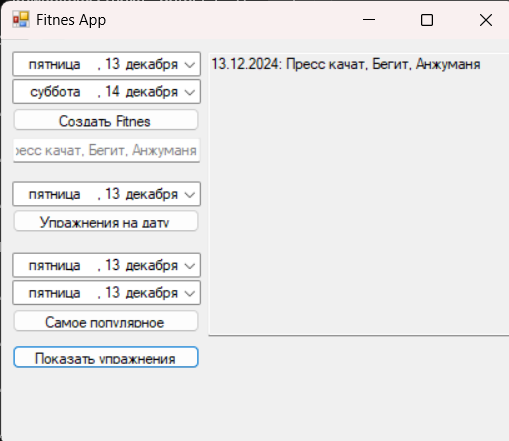


Рисунок 1 — Интерфейс программы.

**5.** Запустим тестовое приложение и убедимся что созданные нами классы работают корректно (рисунок 2).

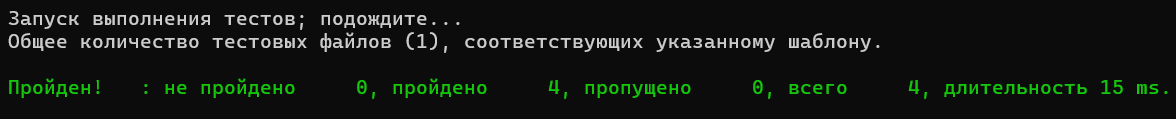


Рисунок 2 — Просмотр тестов

**Вывод:** Инкапсуляция позволяет ограничивать пользователя от вредоносных манипуляций с данными повышая надежность программы. Графические интерфейсы увеличивают читаемость функций программы и позволяют пользователю свободно использовать программу.

**ПРИЛОЖЕНИЕ А**

**Библиотека классов**

**Fitness**

using System.Globalization;

namespace Library;

/// <summary>

/// Represents a fitness tracker that logs exercises performed on specific dates.

/// </summary>

public class Fitnes

{

    /// <summary>

    /// Gets the array of dates tracked in the fitness log.

    /// </summary>

    public DateTime[] Dates { get; private set; }

    /// <summary>

    /// Gets or sets the list of exercises (fits) performed on each date.

    /// </summary>

    public List<string>[] Fits { get; set; }

    /// <summary>

    /// Initializes a new instance of the <see cref="Fitnes"/> class.

    /// </summary>

    /// <param name="startTime">The start date of the fitness log.</param>

    /// <param name="endTime">The end date of the fitness log.</param>

    /// <param name="fits">The exercises performed for each date.</param>

    public Fitnes(DateTime startTime, DateTime endTime, List<string>[] fits)

    {

        if (startTime > endTime)

        {

            Console.WriteLine("Data error. Wrong Time.");

            return;

        }

        int totalDays = (endTime - startTime).Days + 1;

        Dates = new DateTime[totalDays];

        for (int i = 0; i < totalDays; i++)

        {

            Dates[i] = startTime.AddDays(i);

        }

        Fits = fits;

    }

    /// <summary>

    /// Counts the number of exercises performed on a specific date.

    /// </summary>

    /// <param name="date">The date to check.</param>

    /// <returns>The count of exercises performed on the specified date. Returns 0 if the date is not tracked.</returns>

    public int CountFits(DateTime date)

    {

        int index = Array.IndexOf(Dates, date);

        if (index == -1)

        {

            return 0;

        }

        else

        {

            return index;

        }

    }

    /// <summary>

    /// Counts the occurrences of each exercise performed between two dates.

    /// </summary>

    /// <param name="startTime">The start date of the range.</param>

    /// <param name="endTime">The end date of the range.</param>

    public void CountFits(DateTime startTime, DateTime endTime)

    {

        if (startTime > endTime || Array.IndexOf(Dates, startTime) == -1 || Array.IndexOf(Dates, endTime) == -1)

        {

            Console.WriteLine("Data error. Wrong Time.");

            return;

        }

        List<string> fits = new List<string>();

        List<int> count = new List<int>();

        for (int i = (startTime - Dates[0]).Days; i < (endTime - Dates[0]).Days; i++)

        {

            for (int j = 0; j < this.Fits[i].Count; j++)

            {

                if (fits.IndexOf(this.Fits[i][j]) == -1)

                {

                    fits.Add(this.Fits[i][j]);

                    count.Add(1);

                }

                else

                {

                    count[fits.IndexOf(this.Fits[i][j])]++;

                }

            }

        }

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

        {

            Console.WriteLine($"Exercise {fits[i]} occurred {count[i]} times.");

        }

    }

    /// <summary>

    /// Finds the most frequently performed exercise within a specified date range.

    /// </summary>

    /// <param name="startTime">The start date of the range.</param>

    /// <param name="endTime">The end date of the range.</param>

    /// <returns>The name of the most frequently performed exercise, or an error message if the input is invalid.</returns>

    public string MostPopularFit(DateTime startTime, DateTime endTime)

    {

        if (startTime > endTime || Array.IndexOf(Dates, startTime) == -1 || Array.IndexOf(Dates, endTime) == -1)

        {

            return "Data error. Wrong Time.";

        }

        List<string> fits = new List<string>();

        List<int> count = new List<int>();

        for (int i = (startTime - Dates[0]).Days; i < (endTime - Dates[0]).Days; i++)

        {

            for (int j = 0; j < this.Fits[i].Count; j++)

            {

                if (fits.IndexOf(this.Fits[i][j]) == -1)

                {

                    fits.Add(this.Fits[i][j]);

                    count.Add(1);

                }

                else

                {

                    count[fits.IndexOf(this.Fits[i][j])]++;

                }

            }

        }

        return fits[count.IndexOf(count.Max())];

    }

}

**ПРИЛОЖЕНИЕ B**

**Модульные тесты**

namespace Test;

using Library;

public class FitnesTests

{

    [Fact]

    public void ConstructorTest()

    {

        DateTime startTime = new DateTime(2023, 12, 1);

        DateTime endTime = new DateTime(2023, 12, 5);

        var fits = new List<string>[]

        {

            new List<string> { "Push-up", "Squat" },

            new List<string> { "Sit-up" },

            new List<string> { "Push-up" },

            new List<string> { "Squat" },

            new List<string> { "Pull-up" }

        };

        var fitnes = new Fitnes(startTime, endTime, fits);

        Assert.Equal(5, fitnes.Dates.Length);

        Assert.Equal(new DateTime(2023, 12, 1), fitnes.Dates[0]);

        Assert.Equal(new DateTime(2023, 12, 5), fitnes.Dates[4]);

        Assert.Equal(fits, fitnes.Fits);

    }

    [Fact]

    public void CountFitsTest1()

    {

        DateTime startTime = new DateTime(2023, 12, 1);

        DateTime endTime = new DateTime(2023, 12, 5);

        var fits = new List<string>[]

        {

            new List<string> { "Push-up", "Squat" },

            new List<string> { "Sit-up" },

            new List<string> { "Push-up" },

            new List<string> { "Squat" },

            new List<string> { "Pull-up" }

        };

        var fitnes = new Fitnes(startTime, endTime, fits);

        int count = fitnes.CountFits(new DateTime(2023, 12, 3));

        Assert.Equal(2, count); // "Push-up" is on the 3rd day.

    }

    [Fact]

    public void MostPopularFitTest1()

    {

        DateTime startTime = new DateTime(2023, 12, 1);

        DateTime endTime = new DateTime(2023, 12, 5);

        var fits = new List<string>[]

        {

            new List<string> { "Push-up", "Squat" },

            new List<string> { "Sit-up" },

            new List<string> { "Push-up" },

            new List<string> { "Squat" },

            new List<string> { "Pull-up" }

        };

        var fitnes = new Fitnes(startTime, endTime, fits);

        string mostPopular = fitnes.MostPopularFit(new DateTime(2023, 12, 1), new DateTime(2023, 12, 3));

        Assert.Equal("Push-up", mostPopular);

    }

    [Fact]

    public void MostPopularFitTest2()

    {

        DateTime startTime = new DateTime(2023, 12, 1);

        DateTime endTime = new DateTime(2023, 12, 5);

        var fits = new List<string>[]

        {

            new List<string> { "Push-up", "Squat" },

            new List<string> { "Sit-up" },

            new List<string> { "Push-up" },

            new List<string> { "Squat" },

            new List<string> { "Pull-up" }

        };

        var fitnes = new Fitnes(startTime, endTime, fits);

        string result = fitnes.MostPopularFit(new DateTime(2023, 11, 30), new DateTime(2023, 12, 6));

        Assert.Equal("Data error. Wrong Time.", result);

    }

}

**ПРИЛОЖЕНИЕ C**

**Интерфейс WindowsForm**

**FitnessForm**

namespace WindowsForm;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Windows.Forms;

using Library;

public partial class FitnessForm : Form

{

    private Fitnes fitnes;

    public FitnessForm()

    {

        InitializeComponent();

    }

    private void btnCreate\_Click(object sender, EventArgs e)

    {

        DateTime startDate = dtpStart.Value;

        DateTime endDate = dtpEnd.Value;

        // Инициализация данных для примера

        List<string>[] fits = new List<string>[(endDate - startDate).Days + 1];

        for (int i = 0; i < fits.Length; i++)

        {

            fits[i] = new List<string>(txtExercises.Text.Split(',').Select(s => s.Trim()));

        }

        fitnes = new Fitnes(startDate, endDate, fits);

        MessageBox.Show("Объект Fitnes создан!");

    }

    private void btnCountFits\_Click(object sender, EventArgs e)

    {

        if (fitnes == null)

        {

            MessageBox.Show("Сначала создайте объект Fitnes.");

            return;

        }

        DateTime selectedDate = dtpSingleDate.Value;

        int count = fitnes.CountFits(selectedDate);

        MessageBox.Show($"На {selectedDate.ToShortDateString()} упражнений: {count}");

    }

    private void btnPopularFit\_Click(object sender, EventArgs e)

    {

        if (fitnes == null)

        {

            MessageBox.Show("Сначала создайте объект Fitnes.");

            return;

        }

        DateTime startDate = dtpRangeStart.Value;

        DateTime endDate = dtpRangeEnd.Value;

        string mostPopularFit = fitnes.MostPopularFit(startDate, endDate);

        MessageBox.Show($"Самое популярное упражнение: {mostPopularFit}");

    }

    private void btnShowExercises\_Click(object sender, EventArgs e)

    {

        if (fitnes == null)

        {

            MessageBox.Show("Сначала создайте объект Fitnes.");

            return;

        }

        // Отображаем список упражнений на каждый день

        txtDisplayExercises.Clear();

        for (int i = 0; i < fitnes.Dates.Length; i++)

        {

            txtDisplayExercises.AppendText($"{fitnes.Dates[i].ToShortDateString()}: {string.Join(", ", fitnes.Fits[i])}\n");

        }

    }

}

**FitnessForm.Designer**

namespace WindowsForm;

partial class FitnessForm

{

    private System.ComponentModel.IContainer components = null;

    protected override void Dispose(bool disposing)

    {

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

        {

            components.Dispose();

        }

        base.Dispose(disposing);

    }

    private void InitializeComponent()

    {

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

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

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

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

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

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

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

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

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

        this.txtExercises = new System.Windows.Forms.TextBox(); // Инициализация txtExercises

        this.txtDisplayExercises = new System.Windows.Forms.TextBox(); // Инициализация txtDisplayExercises

        this.SuspendLayout();

        // Start DateTimePicker

        this.dtpStart.Location = new System.Drawing.Point(12, 12);

        this.dtpStart.Name = "dtpStart";

        this.dtpStart.Size = new System.Drawing.Size(200, 22);

        this.dtpStart.TabIndex = 0;

        // End DateTimePicker

        this.dtpEnd.Location = new System.Drawing.Point(12, 40);

        this.dtpEnd.Name = "dtpEnd";

        this.dtpEnd.Size = new System.Drawing.Size(200, 22);

        this.dtpEnd.TabIndex = 1;

        // Create Button

        this.btnCreate.Location = new System.Drawing.Point(12, 68);

        this.btnCreate.Name = "btnCreate";

        this.btnCreate.Size = new System.Drawing.Size(200, 23);

        this.btnCreate.TabIndex = 2;

        this.btnCreate.Text = "Создать Fitnes";

        this.btnCreate.UseVisualStyleBackColor = true;

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

        // Single Date DateTimePicker

        this.dtpSingleDate.Location = new System.Drawing.Point(12, 140);

        this.dtpSingleDate.Name = "dtpSingleDate";

        this.dtpSingleDate.Size = new System.Drawing.Size(200, 22);

        this.dtpSingleDate.TabIndex = 3;

        // Count Fits Button

        this.btnCountFits.Location = new System.Drawing.Point(12, 168);

        this.btnCountFits.Name = "btnCountFits";

        this.btnCountFits.Size = new System.Drawing.Size(200, 23);

        this.btnCountFits.TabIndex = 4;

        this.btnCountFits.Text = "Упражнения на дату";

        this.btnCountFits.UseVisualStyleBackColor = true;

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

        // Range Start DateTimePicker

        this.dtpRangeStart.Location = new System.Drawing.Point(12, 210);

        this.dtpRangeStart.Name = "dtpRangeStart";

        this.dtpRangeStart.Size = new System.Drawing.Size(200, 22);

        this.dtpRangeStart.TabIndex = 5;

        // Range End DateTimePicker

        this.dtpRangeEnd.Location = new System.Drawing.Point(12, 238);

        this.dtpRangeEnd.Name = "dtpRangeEnd";

        this.dtpRangeEnd.Size = new System.Drawing.Size(200, 22);

        this.dtpRangeEnd.TabIndex = 6;

        // Popular Fit Button

        this.btnPopularFit.Location = new System.Drawing.Point(12, 266);

        this.btnPopularFit.Name = "btnPopularFit";

        this.btnPopularFit.Size = new System.Drawing.Size(200, 23);

        this.btnPopularFit.TabIndex = 7;

        this.btnPopularFit.Text = "Самое популярное упражнение";

        this.btnPopularFit.UseVisualStyleBackColor = true;

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

        // txtExercises

        this.txtExercises.Location = new System.Drawing.Point(12, 97);

        this.txtExercises.Name = "txtExercises";

        this.txtExercises.Size = new System.Drawing.Size(200, 22);

        this.txtExercises.TabIndex = 3;

        this.txtExercises.Text = "Введите упражнения"; // Подсказка в тексте

        this.txtExercises.ForeColor = System.Drawing.Color.Gray; // Цвет подсказки

        // Обработчик события "Enter", когда пользователь начинает вводить

        this.txtExercises.Enter += new System.EventHandler(this.TxtExercises\_Enter);

        // Обработчик события "Leave", когда пользователь покидает поле

        this.txtExercises.Leave += new System.EventHandler(this.TxtExercises\_Leave);

        //

        // btnShowExercises

        //

        this.btnShowExercises.Location = new System.Drawing.Point(12, 302);

        this.btnShowExercises.Name = "btnShowExercises";

        this.btnShowExercises.Size = new System.Drawing.Size(200, 23);

        this.btnShowExercises.TabIndex = 9;

        this.btnShowExercises.Text = "Показать упражнения";

        this.btnShowExercises.UseVisualStyleBackColor = true;

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

        //

        // txtDisplayExercises

        //

        this.txtDisplayExercises.Location = new System.Drawing.Point(220, 12);

        this.txtDisplayExercises.Multiline = true;

        this.txtDisplayExercises.Name = "txtDisplayExercises";

        this.txtDisplayExercises.Size = new System.Drawing.Size(400, 280);

        this.txtDisplayExercises.TabIndex = 10;

        this.txtDisplayExercises.ReadOnly = true;

        // MainForm

        this.AutoScaleDimensions = new System.Drawing.SizeF(8F, 16F);

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

        this.ClientSize = new System.Drawing.Size(550, 400);

        this.Controls.Add(this.btnPopularFit);

        this.Controls.Add(this.dtpRangeEnd);

        this.Controls.Add(this.dtpRangeStart);

        this.Controls.Add(this.btnCountFits);

        this.Controls.Add(this.dtpSingleDate);

        this.Controls.Add(this.btnCreate);

        this.Controls.Add(this.dtpEnd);

        this.Controls.Add(this.dtpStart);

        this.Controls.Add(this.btnShowExercises);

        this.Controls.Add(this.txtExercises);

        this.Controls.Add(this.txtDisplayExercises);

        this.Name = "MainForm";

        this.Text = "Fitnes App";

        this.ResumeLayout(false);

    }

    private void TxtExercises\_Enter(object sender, EventArgs e)

    {

        if (txtExercises.Text == "Введите упражнения (через запятую)")

        {

            txtExercises.Text = "";

            txtExercises.ForeColor = System.Drawing.Color.Black; // Цвет текста

        }

    }

    private void TxtExercises\_Leave(object sender, EventArgs e)

    {

        if (string.IsNullOrWhiteSpace(txtExercises.Text))

        {

            txtExercises.Text = "Введите упражнения (через запятую)";

            txtExercises.ForeColor = System.Drawing.Color.Gray; // Цвет подсказки

        }

    }

    private System.Windows.Forms.DateTimePicker dtpStart;

    private System.Windows.Forms.DateTimePicker dtpEnd;

    private System.Windows.Forms.Button btnCreate;

    private System.Windows.Forms.DateTimePicker dtpSingleDate;

    private System.Windows.Forms.Button btnCountFits;

    private System.Windows.Forms.DateTimePicker dtpRangeStart;

    private System.Windows.Forms.DateTimePicker dtpRangeEnd;

    private System.Windows.Forms.Button btnShowExercises;

    private System.Windows.Forms.Button btnPopularFit;

    private System.Windows.Forms.TextBox txtDisplayExercises;

    private System.Windows.Forms.TextBox txtExercises;

}