Министерство науки и высшего образования Российской Федерации

ФГБОУ ВО «Кубанский Государственный Технологический Университет»

Кафедра информационных систем и программирования

Отчет по лабораторной работе №9

По дисциплине «Рефакторинг и работа с унаследованным кодом»

«Выделение бизнес-логики и применение различных вариантов визуального представления»

Выполнил студент

Группы 21-КБ-ПР2

Жулькин Д.П.

Преподаватель:

Степанова Елизавета Владимировна

Краснодар

2024

Цель работы: изучить техники рефакторинга и способы совместного использования общего кода в различных приложениях.

Задание:

1. Создать новое оконное приложение и библиотеку классов в соответствии с предложенными в пункте 2.4 изменениями, применив описанные в теории методы рефакторинга.
2. Проверить согласованность результатов и корректность работы нового варианта программы с результатами старой версии.
3. Оформить отчёт.

Процесс выполнения работы:

1) Создать новый проект C# Class Library.

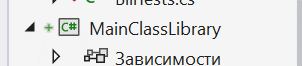


Рисунок 1 – Создан новый проект MainClassLibrary

2) Перенести в него основные классы старого консольного приложения (все кроме Program).

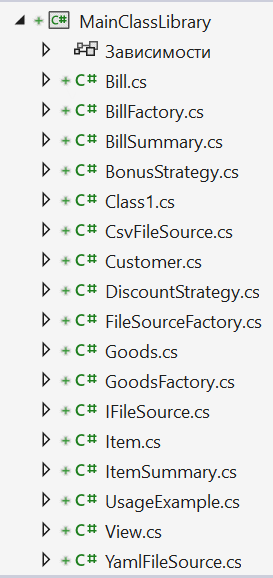


Рисунок 2 – Файлы старого проекта перенесены

3) Добавить ссылку на новую библиотеку в консольном приложении.

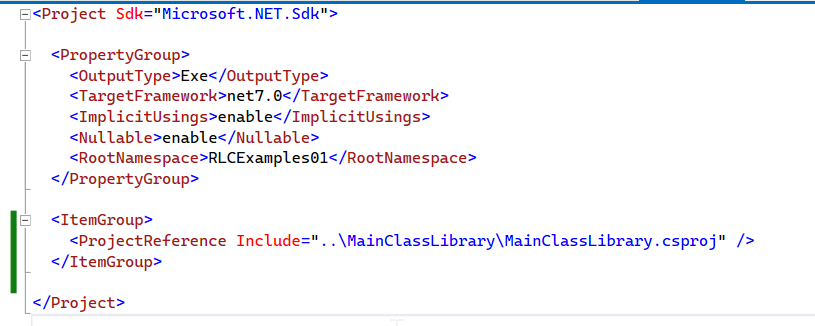


Рисунок 3 – Добавлена ссылка на библиотеку

4) Добавить пространство имен новой библиотеки в консольное приложение.



Рисунок 4 – Добавлено пространство имен библиотеки

5) Обновить ссылки на тестируемые классы в тестовом проекте.

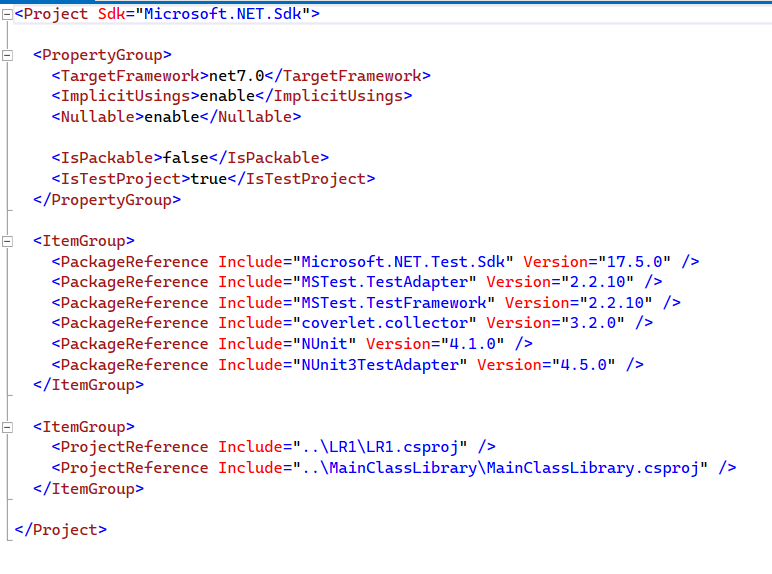


Рисунок 5 – Обновлены ссылки на проект

6) Создать новый проект оконного приложения C# Windows Forms Application.

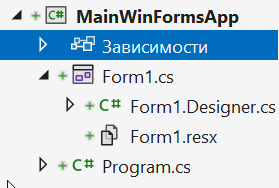


Рисунок 6 – Добавлен новый проект оконного приложения

7) Добавить ссылку на новую библиотеку в оконном приложении.

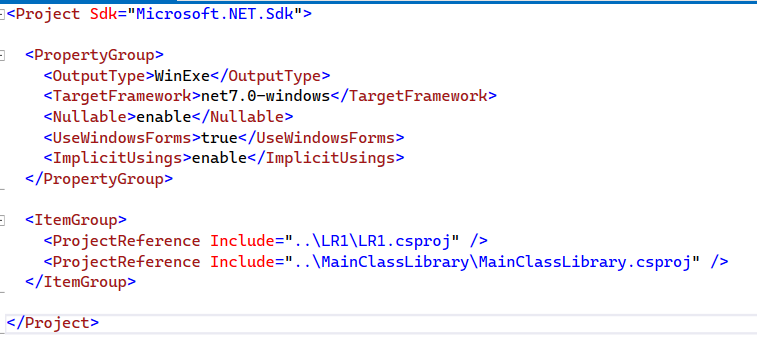


Рисунок 7 – Ссылка на новую библиотеку

8) Добавить пространство имен новой библиотеки в оконное приложение.

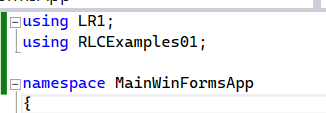


Рисунок 8 – Пространство имен новой библиотеки

9) Добавить необходимые визуальные компоненты (диалоги для выбора загружаемого файла и результатов работы, поле для вывода результатов генерации чека, кнопки управления работой (запуск генерации, выход и т.д.), переключатели для выбора формата вывода (HTML/TXT) и т.д.)

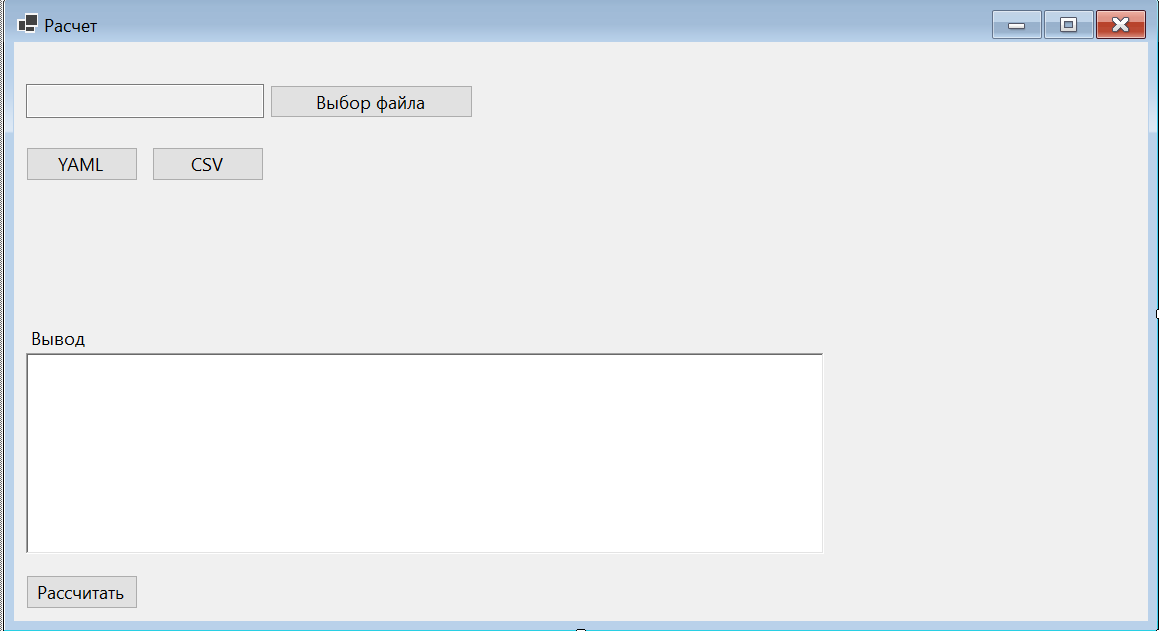


Рисунок 9 – Добавлены визуальные компоненты

10) Связать визуальные компоненты с кодом библиотеки.

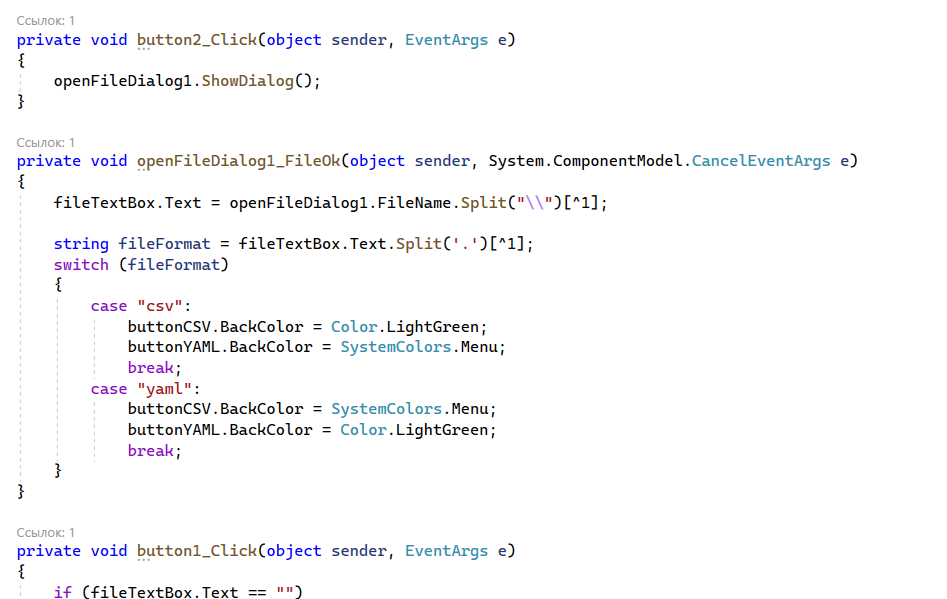


Рисунок 10 – Визуальные компоненты связаны с кодом библиотеки

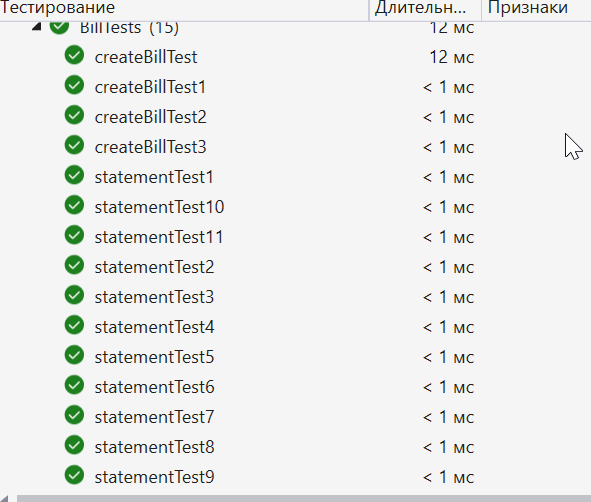


Рисунок 11 – Тесты выполняются успешно

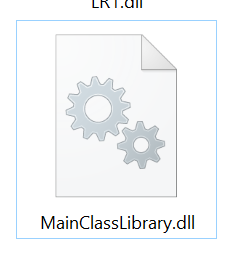


Рисунок 12 – Результат выполнения лабораторной работы

Вывод: задания выполнены успешно, освоены продвинутые практики рефакторинга.

Листинг:

using LR1;

using RLCExamples01;

namespace MainWinFormsApp

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void button2\_Click(object sender, EventArgs e)

{

openFileDialog1.ShowDialog();

}

private void openFileDialog1\_FileOk(object sender, System.ComponentModel.CancelEventArgs e)

{

fileTextBox.Text = openFileDialog1.FileName.Split("\\")[^1];

string fileFormat = fileTextBox.Text.Split('.')[^1];

switch (fileFormat)

{

case "csv":

buttonCSV.BackColor = Color.LightGreen;

buttonYAML.BackColor = SystemColors.Menu;

break;

case "yaml":

buttonCSV.BackColor = SystemColors.Menu;

buttonYAML.BackColor = Color.LightGreen;

break;

}

}

private void button1\_Click(object sender, EventArgs e)

{

if (fileTextBox.Text == "")

{

MessageBox.Show("Выберите файл!");

return;

}

StreamReader sr = new StreamReader(openFileDialog1.OpenFile());

BillFactory billFactory = new BillFactory(

fileSource: FileSourceFactory.Create(fileTextBox.Text)

);

string bill = billFactory.CreateBill(sr);

outputTextBox.Text = bill;

}

private void button3\_Click(object sender, EventArgs e)

{

outputTextBox.Text = "";

}

}

}