

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра «ОБЧИСЛЮВАЛЬНА ТЕХНІКА ТА ПРОГРАМУВАННЯ»

«Програмна технологія DOTNET»

Звіт з лабораторної роботи №3

Тема: «Обробка колекцій об'єктів. Робота з файлами»

Виконав:
ст. гр. КІТ-118в
Соколенко Д.Г.

Перевірив:
Бартош М.В.

Харків – 2020

Мета: отримати практичний досвід у розробці програм на С# і консольних додатків через додавання у попередню роботу можливість обробки класу-контейнеру і роботу з файлами.

Загальне завдання:

- Збереження введених даних у файлі для подальшого редагування;
- Відновлення раніше збережених даних з файлу;
- Пошук та редагування даних студента;
- Видалення даних особистої справи студента;
- У звіті привести опис використовуваних методів і алгоритмів роботи з файлами

Хід роботи

1. Спочатку був доданий метод запису інформації з контейнера у файл, що викликається 1 раз у кінці програми (рис.1).

```
ссылка: 1
public static void WriteStContainerToFile(string path, StudentContainer writtenContainer)
{
    try
    {
        using (StreamWriter sw = new StreamWriter(path, false, System.Text.Encoding.Default))
        {
            foreach (var i in writtenContainer)
            {
                sw.WriteLine(i.ToInfoString());
            }
        }
    }
    catch (Exception e)
    {
        Console.WriteLine(e.Message);
    }
}
```

Рисунок 1 – метод запису

2. Далі був доданий метод читання інформації з файлу у контейнер, що викликається 1 раз у початку програми (рис.2).

```
ссылка: 1
public static void ReadStContainerFromFile(string path, StudentContainer readContainer)
{
    try
    {
        using (StreamReader sr = new StreamReader(path, System.Text.Encoding.Default))
        {
            string line;
            while ((line = sr.ReadLine()) != null)
            {
                var infoStudent = line.Split(new char[] { '|' });

                readContainer.AddStudent(new Student(infoStudent[0],
                    infoStudent[1],
                    infoStudent[2],
                    DateTime.Parse(infoStudent[3]),
                    DateTime.Parse(infoStudent[4]),
                    char.Parse(infoStudent[5]),
                    infoStudent[6],
                    infoStudent[7],
                    double.Parse(infoStudent[8])));
            }
        }
    }
}
```

Рисунок 2 – метод читання

3. Наступним скором були реалізовані методи пошуку та редагування даних студента (рис.3 – 4).

```
Make a choice: 4
Input the index: 0
Last name: F
First name: I
Patronymic: Alexandrovich
BirthDate: 03.04.0005
EnterDate: 11.11.2011
Group Index: b
Faculty: v
Specialization: v
Performance: 1
```

Рисунок 4 – 0-ий елемент перед редагуванням

```
Choose what you want to do:
1 - Show data for all students
2 - Add new student
3 - Delete student by index
4 - Show by index
5 - Edit the student

0 - Exit
Make a choice: 5
Input the index: 0
Choose what you want to edit:

1 - Last name
2 - First name
3 - Patronymic
4 - Birth date
5 - Enter date
6 - Group index
7 - Faculty
8 - Specialization
9 - Performance
9
Performance: 100
```

Рисунок 4 – редагування 0-ого елемента, зміна поля Performance.

4. Далі продемонстрований метод видалення елемента з контейнеру – пункт 3 меню (рис.5).

```
1 - Show data for all students
2 - Add new student
3 - Delete student by index
4 - Show by index
5 - Edit the student
```

Рисунок 5 – видалення елемента

5. Також показаний зв'язок класів у програмі (рис.6).

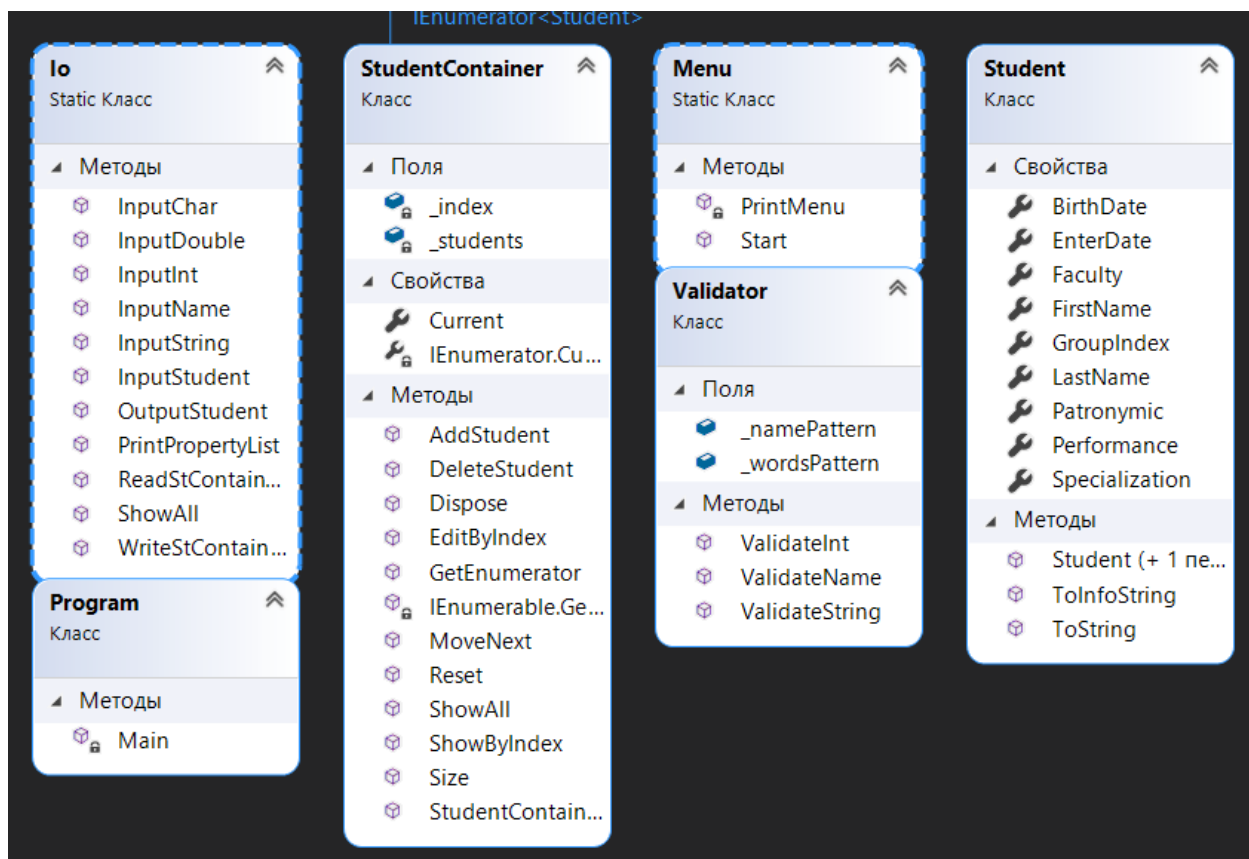


Рисунок 6 – Зв'язок класів у програмі

Висновки: отримали практичний досвід у розробці програм на C# і консольних додатків через додавання у попередню роботу можливість обробки класу-контейнеру і роботу з файлами.