Лабораторная работа №3

Создание произвольного класса. Массивы объектов. Обработка текстовой информации.

Задание:

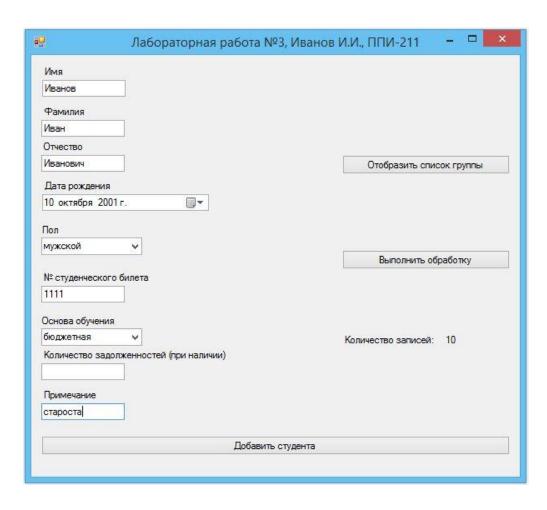
Создать программу реализующую обработку данных введенных пользователем по следующим алгоритмам:

- 1) Отбор данных(вводятся пользователем) с последующей сортировкой по указанным признакам(по вариантам)
- 2) Вывод всех элементов массива в алфавитном порядке

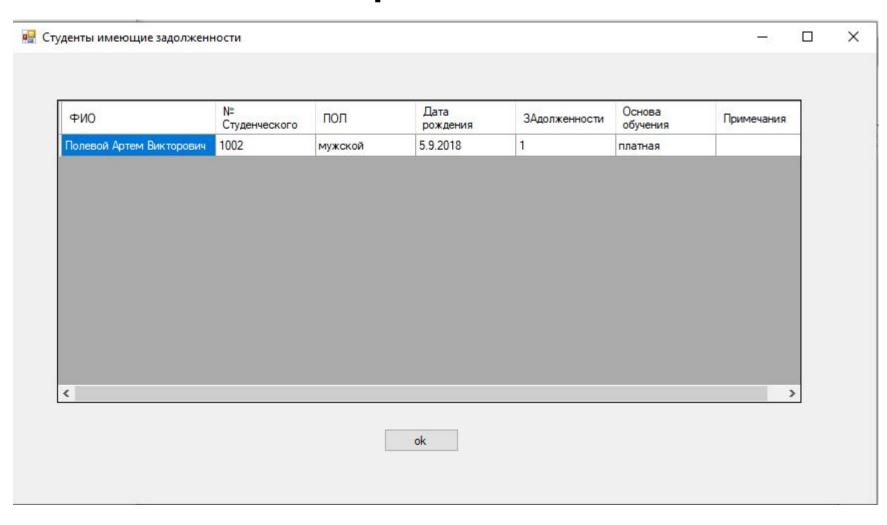
Требования к программе

- 1) Начальная форма приложения должна содержать 9 полей для ввода данных, поле для вывода текущего значения количества записей, кнопку по которой создается новый элемент массива из введенных пользователем данных и отдельную кнопку для реализации каждого действия из указанных в задании.
- 2) Заголовок начальной формы ФИО студента и номер группы, заголовок формы с выводом результата действий по названию действия(«Результат обработки», «Список группы» и т.д.)
- 3) Результат выполнения каждого из действий выводится на новой форме, имеющей табличное поле для вывода результата и кнопку ОК, реализующей закрытие формы. Предусмотреть блокирование начальной формы до закрытия форм с выводом результатов. На форме демонстрирующей все текущие элементы массива предусмотреть возможность редактирования и удаления записей.
- 4) Для передачи данных между объектами-формами использовать отдельный класс
- 5) Для ввода полей с ограниченным набором значений (пол, основа обучения) использовать элемент comboBox, для даты рождения dateTimePicker.
- 6) Предусмотреть проверку введенных пользователем данных с генерацией сообщения об ошибке в случае необходимости.
- 7) Обеспечить базовое наполнение массива 10 записями

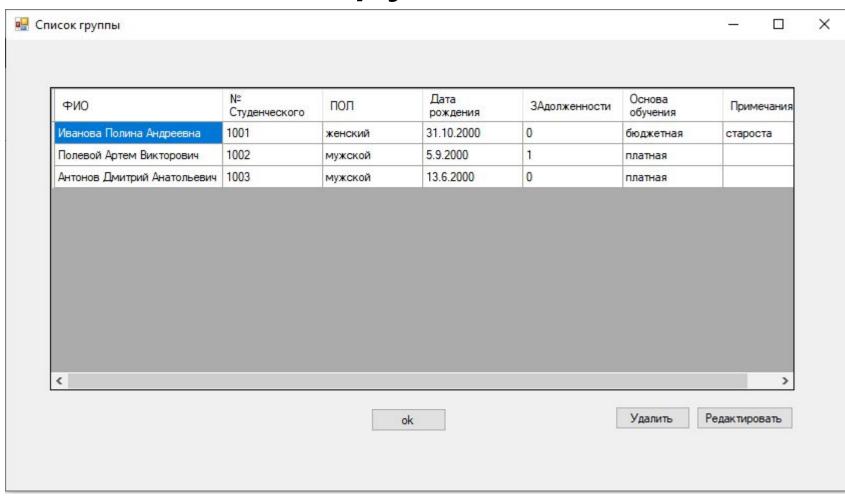
Пример основной формы



Пример формы с результатом обработки



Пример формы отображающей список группы



Пример конструкции пользовательского класса

```
public class Student
public string Surname; // Фамилия
public string Name;
                     // Имя
public string SName; // Отчество
public int BDateD; // Число ДР
public int BDateM; // Месяц ДР
public int BDateY; // Год ДР
public string Gender; // Пол
public int S_number; // № студенческого билета
public string L_base; // основа обучения
public int Debts; // количество задолженностей
public string Note; // примечание
```

Пример создания массива объектов пользовательского класса

Обозначени е массива

public Student[] array1 = new Student[20];

Модификато р прав доступа

Имя пользовательско го класса Имя создаваемог о массива

Создание нового объекта класса Student

Количество элементов в создаваемо м массиве

Варианты задания

Отбор данных:

- 1. Платная форма обучения
- 2. Бюджетная форма обучения
- 3. Наличие задолженностей
- 4. Отсутствие задолженностей
- 5. Мужской пол
- 6. Женский пол

Сортировка данных:

- 1. ФИО в алфавитном порядке
- 2. ФИО в обратном алфавитному порядке
- 3. Дата рождения по возрастанию
- 4. Дата рождения по убыванию
- 5. Номер студенческого билета по возрастанию
- 6. Номер студенческого билета по убыванию

Требования к отчету:

- Оформление по ГОСТ
- Содержание:
 - 1) формулировка задания
- 2) использованные способы передачи данных между объектами(с кратким описанием)
 - 3) блок схема программы (укрупненная)
- 4) скриншоты работы программы (начальная форма + результат каждого действия + сообщение об ошибке)
 - 5) листинг программы (форма1 + форма2 + форма 3)

Полезные ссылки и функции: