Задача: разработать программу на языке C#, которая будет функционировать как редактор информации о студентах вуза. Студенты организованы в группы, каждая группа имеет список студентов и старосту. Программа должна предоставлять два режима работы: отображение данных о студенте/группе без возможности редактирования и редактирование данных. Для реализации этих режимов создадим отдельные классы, используя механизм наследования.

Требования к программе:

1. Создать класс Person, представляющий общие характеристики для студента и старосты. В этом классе должны быть поля для имени, фамилии, возраста и других общих данных.
2. Создать класс Student, который наследует от класса Person. Student должен иметь дополнительные поля, такие как номер студенческого билета и средний балл. Реализовать методы для отображения данных студента и редактирования их.
3. Создать класс Group, который также наследует от класса Person, так как староста - это тоже студент. В классе Group должны быть поля для названия группы и списка студентов, а также методы для добавления/удаления студентов и назначения старосты.
4. Создать класс Editor, представляющий редактор данных. В этом классе должны быть методы для переключения между режимами отображения и редактирования данных студентов и групп.
5. Реализовать главный класс Program, который будет использовать остальные классы для управления данными о студентах и группах вуза.

Этот план задания описывает основные классы и функциональность программы. Вы можете дополнить его, например, добавив методы для сохранения и загрузки данных, или реализовав более сложные операции в редакторе данных.

Приложение 1.

Пример использования программы:

// Создание объектов студентов и групп

Student student1 = new Student("Иван", "Иванов", 20, "12345", 4.5);

Student student2 = new Student("Петр", "Петров", 21, "54321", 3.8);

Group groupA = new Group("Группа А");

groupA.AddStudent(student1);

groupA.AddStudent(student2);

groupA.SetMonitor(student1);

// Создание редактора данных

Editor editor = new Editor();

// Режим отображения данных

editor.DisplayMode();

groupA.DisplayGroupInfo(); // Отображение данных о группе

student1.DisplayStudentInfo(); // Отображение данных о студенте

// Режим редактирования данных

editor.EditMode();

student1.EditStudentInfo(); // Редактирование данных студента

// Снова режим отображения данных

editor.DisplayMode();

student1.DisplayStudentInfo(); // Отображение отредактированных данных студента