

Министерство образования и молодежной политики Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по учебной практике

УП 01.01

Выполнил: Мартынов Арсений

Группа: ПР-31

Преподаватель: Мирошниченко Г.В.

2024

Содержание

[1. **Задание №1 Мобильное приложение «Дневник тренировок»** 3](#_Toc118960339)

[1.1 Описание задачи 3](#_Toc118960340)

[1.2 Структура проекта 3](#_Toc118960341)

[1.3 Описание разработанных функций 3](#_Toc118960342)

[1.4 Алгоритм решения 4](#_Toc118960343)

[1.5 Используемые библиотеки 4](#_Toc118960344)

[1.6 Тестовые случаи 5](#_Toc118960345)

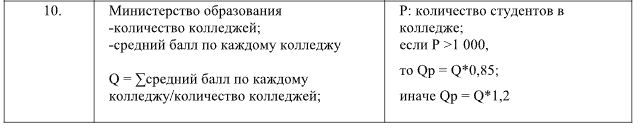
[1.7 Используемые инструменты 7](#_Toc118960346)

[1.8 Описание пользовательского интерфейса 7](#_Toc118960347)

[1.9 Приложение (pr screen экранов) 7](#_Toc118960348)

# 1. **Задание №1 Приложение «Министерство образования»**

## Описание задачи

Выполнить задание по варианту, в WindowsForms с использованием коллекций и методов Linq

## Структура проекта

(Привести изображение со структурой решения. Описать проекты, входящие в него, объяснить их назначение)

## 1.3 Описание разработанных функций

**Базовый класс  
//** метод для вычисления качестваGetQ **//** возвращает среднее значение средних баллов

// метод для получения информации о колледже   
GetInfo  
// возвращает строку с информацией о колледже

//метод добавления колледжа с тремя перегрузками  
 AddCollege   
1-я перегрузка входные средн балл  
2-я перегрузка входные имя колледжа и средн балл  
3-я перегрузка входные средн балл и кол-во студентов  
// увеличивает количество колледжей и добавляет средний балл в список

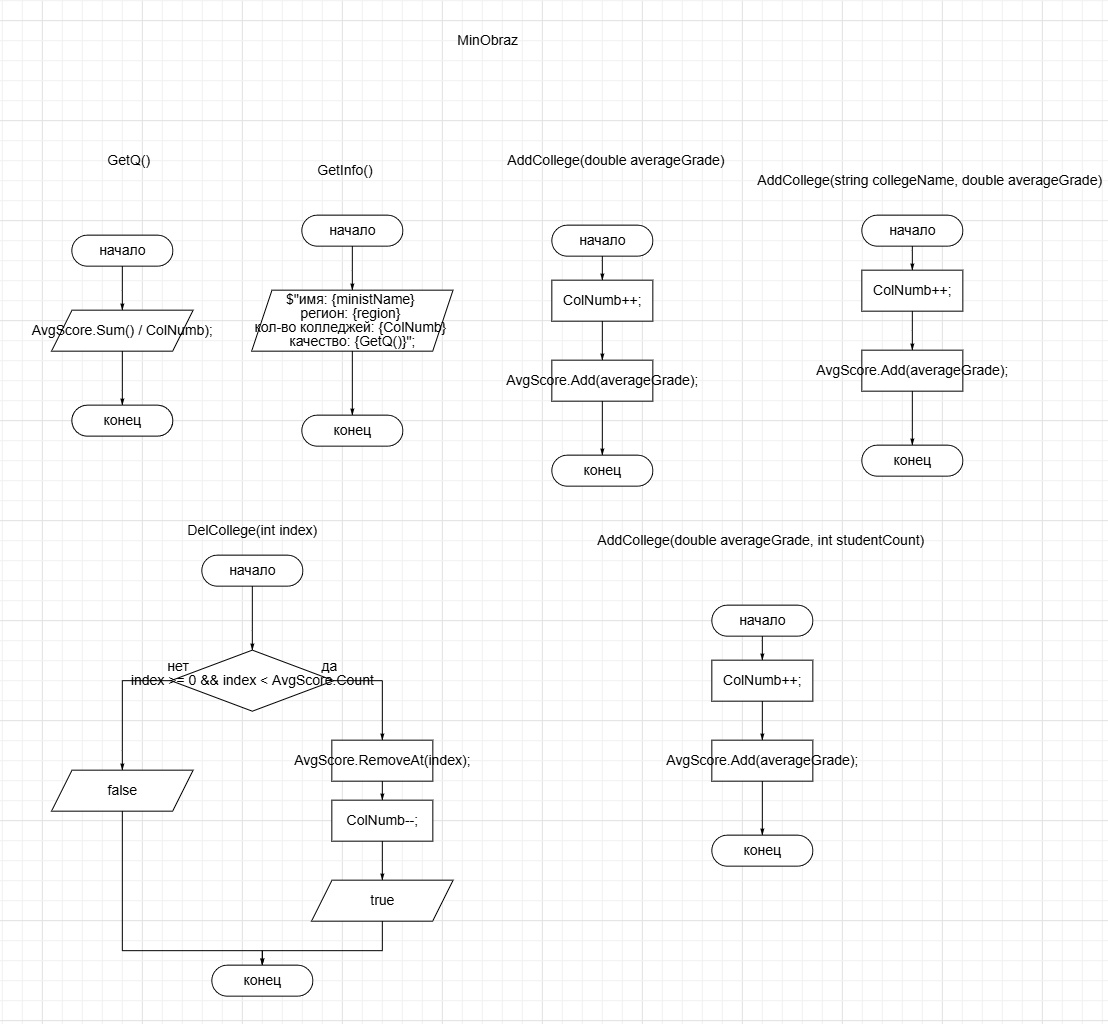
// метод для удаления колледжа по индексу  
DelCollege  
//входные индекс   
// возвращает тру если удаление прошло успешно  
// возвращает фолс если индекс некорректен  
// возвращает фолс если ошибки

**Класс наследник  
//** переопределение метода для вычисления качества  
GetQ  
**//** возвращает качество в зависимости от количества студентов

**//** переопределение метода для получения информации  
GetInfo  
**//** возвращает строку с информацией о колледже и количестве студентов

**//** метод для вычисления соотношения студентов к колледжам  
StudRatio  
**//** возвращает отношение количества студентов к количеству колледжей

**//** метод для получения имени колледжа  
GetName  
**//** возвращает имя колледжа

1.4 Алгоритм решения

## 

## 1.5 Используемые библиотеки

## System;

## System.Collections.Generic;

## System.ComponentModel;

## System.Data;

## System.Drawing;

## System.Linq;

## System.Text;

## System.Threading.Tasks;

## System.Windows.Forms;

## 1.6 Тестовые случаи

**1. TestMinObrGetQ**

* **Назначение теста**: Проверка корректности вычисления качества (Q) для объекта MinObraz.
* **Входные данные**:
  + avgScore: список средних оценок { 4.5, 3.8 }
  + Конструктор MinObraz принимает параметры: 10, avgScore, "имя1", "регион1".
* **Ожидаемый результат**: 0.83m (значение качества).

**2. TestMinObrGetInfo**

* **Назначение теста**: Проверка корректности формирования строки информации о объекте MinObraz.
* **Входные данные**:
  + avgScore: список средних оценок { 4.5, 3.8 }
  + Конструктор MinObraz принимает параметры: 10, avgScore, "имя1", "регион1".
* **Ожидаемый результат**: строка "имя: имя1 регион: регион1 кол-во колледжей: 10 качество: 0,83".

**3. TestMinObrAdd**

* **Назначение теста**: Проверка корректности добавления колледжа в объект MinObraz.
* **Входные данные**:
  + averageGrade: 85.5
  + studentCount: 100
  + Конструктор MinObraz принимает параметры: 10, avgScore, "имя1", "регион1".
* **Ожидаемый результат**: 11 (количество колледжей после добавления).

**4. TestMinObrDel**

* **Назначение теста**: Проверка корректности удаления колледжа из объекта MinObraz.
* **Входные данные**:
  + Конструктор MinObraz принимает параметры: 10, avgScore, "имя1", "регион1".
  + Удаляется колледж с индексом 1.
* **Ожидаемый результат**: 9 (количество колледжей после удаления).

**5. TestMinDopGetQ**

* **Назначение теста**: Проверка корректности вычисления качества (Q) для объекта MinDop.
* **Входные данные**:
  + avgScore: список средних оценок { 4.5, 3.8 }
  + Конструктор MinDop принимает параметры: 10, avgScore, "имя2", "регион2", 1001.
* **Ожидаемый результат**: 0.7055.

**6. TestMinDopGetInfo**

* **Назначение теста**: Проверка корректности формирования строки информации о объекте MinDop.
* **Входные данные**:
  + avgScore: список средних оценок { 4.5, 3.8 }
  + Конструктор MinDop принимает параметры: 10, avgScore, "имя2", "регион2", 1001.
* **Ожидаемый результат**: строка "имя: имя2 регион: регион2 кол-во колледжей: 10 качество: 0,7055 всего студентов: 1001".

**7. TestMinDopGetStudRatio**

* **Назначение теста**: Проверка корректности вычисления соотношения студентов для объекта MinDop.
* **Входные данные**:
  + avgScore: список средних оценок { 4.5, 3.8 }
  + Конструктор MinDop принимает параметры: 10, avgScore, "имя2", "регион2", 1001.
* **Ожидаемый результат**: 100.1.

**8. TestMinDopGetName**

* **Назначение теста**: Проверка корректности получения имени для объекта MinDop.
* **Входные данные**:
  + avgScore: список средних оценок { 4.5, 3.8 }
  + Конструктор MinDop принимает параметры: 10, avgScore, "имя2", "регион2", 1001.
* **Ожидаемый результат**: строка "имя2".

## 1.7 Используемые инструменты

**Используемый язык программирования:** C#  
**Среда разработки:** Visual Studio 2022  
**Фреймворк:** .NET Framework 4.7.2

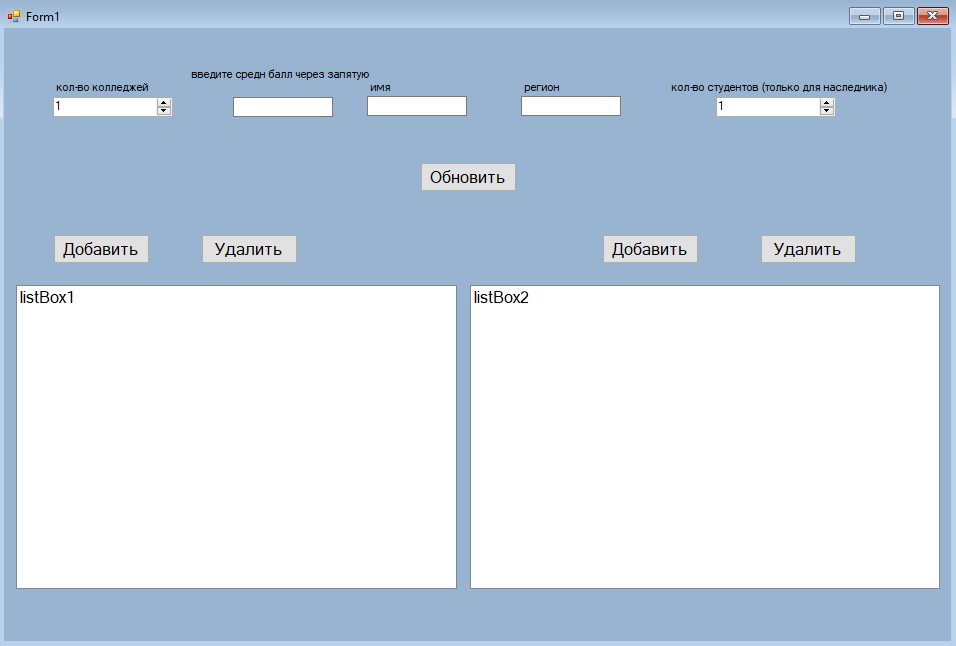
## 1.8 Описание пользовательского интерфейса

**Основные элементы интерфейса:**

1. **Поля ввода**:
   * **Кол-во колледжей**: Элемент управления NumericUpDown
   * **Средний балл**: Текстовое поле для ввода среднего балла через запятую.
   * **Имя**: Текстовое поле для ввода имени.
   * **Регион**: Текстовое поле для ввода региона.
   * **Кол-во студентов (только для наследника)**: Элемент управления NumericUpDown
2. **Кнопка "Обновить"**:
   * **Назначение**: Обновляет данные на основе введенной информации.
3. **Списки (ListBox1 и ListBox2)**:
   * **Назначение**: Отображают списки элементов базового класса и класса наследника
4. **Кнопки "Добавить" и "Удалить"**:
   * **Назначение**:
     + **Добавить**: Добавляет новый элемент в соответствующий список.
     + **Удалить**: Удаляет выбранный элемент из списка.

**Как это должно работать:**

* Пользователь вводит данные в соответствующие поля и выбирает значения из выпадающих списков.
* Нажатие на кнопку "Обновить" применяет изменения и обновляет отображаемую информацию.
* Кнопки "Добавить" и "Удалить" позволяют управлять элементами в списках, добавляя новые или удаляя существующие записи.

1.9 Приложение