МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное образовательное учреждение высшего образования

САНКТ – ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ

КАФЕДРА № 42

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ассистент |  |  |  | Д.О. Шевяков |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №2  (ВАРИАНТ №27)  КОНСТРУКТОРЫ КЛАССА |
|  |
| по курсу: ОСНОВЫ ПРОГРАММИРОВАНИЯ |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4321 |  |  |  | П.А. Евстафьева |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2024

**СОДЕРЖАНИЕ**

[1 Постановка задачи 3](#_gjdgxs)

[2 Выполнение задачи](#_30j0zll) 3

[2.1 Описание реализованной функции](#_3znysh7) 3

[2.2 Листинг программы](#_2et92p0) 4

[2.3 Тесты работы программы](#_tyjcwt) 6

2.4 Интерфейс программы 7

[3 Выводы](#_3dy6vkm) 7

**1 Постановка задачи**

Необходимо разработать Windows Forms приложение на языке C# для управления списком самолётов. Программа должна поддерживать следующие функции:

* создание объектов самолётов с указанием модели, дальности полёта, авиакомпании и расхода топлива;
* сортировка самолётов по расходу топлива;
* отображение списка самолётов и возможности поиска самолёта по параметрам расхода топлива;
* построение диаграммы классов;
* использование структур;
* создание и использование конструктора без параметров;
* реализовать настройку цвета фона формы;

Таблица 1 – Индивидуальное задание по варианту №27

|  |  |  |
| --- | --- | --- |
| Задание | Вход | Выход |
| **Данные:**   * Модель самолета; * Авиакомпания; * Дальность полета самолета; * Запас горючего.   **Задача:**  Провести сортировку самолетов компании по дальности полета. |  |  |

**2 Выполнение задачи**

**2.1 Описание реализованной функции**

Основной класс приложения — Airplane. Он содержит информацию о самолёте, включая модель, дальность полёта, авиакомпанию и расход топлива. В приложении реализованы функции для добавления самолёта в список, сортировки по расходу топлива, а также вывода списка самолётов на форму.

Основные методы:

* AddAirplane: добавление самолёта в список;
* SortAirplanesByFuelConsumption: сортировка самолётов по расходу топлива;
* UpdateListBox: обновление списка на форме.

**2.2 Листинг программы**

Класс Airplane:

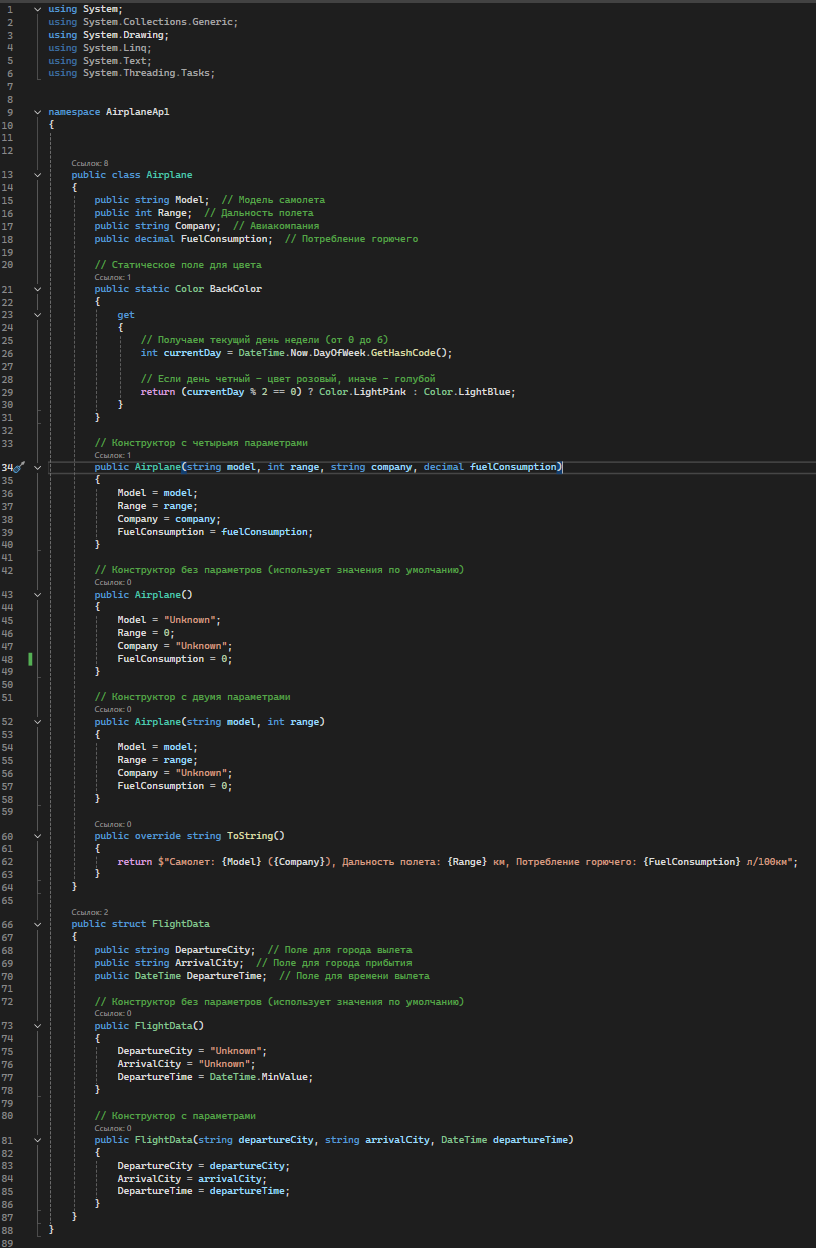


Рисунок 1 – Код класса Airplane

Код формы (Form1):

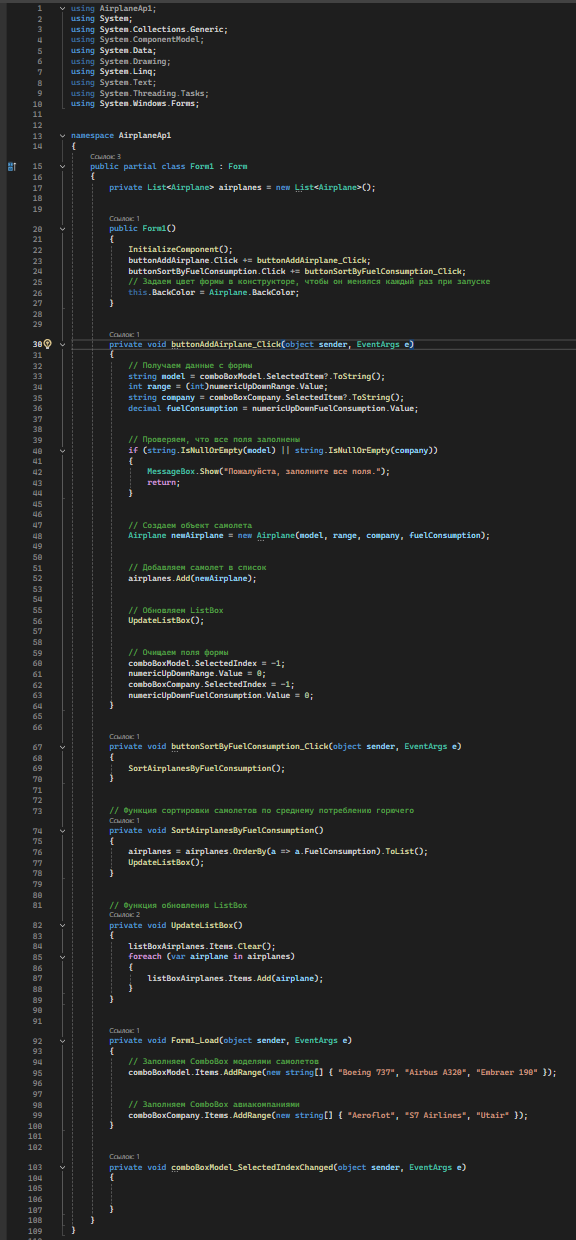


Рисунок 2 – Код формы (Form1)

**2.3 Тесты работы программы**

Программа была протестирована на следующих сценариях:

1. Добавление нескольких самолётов в список.

2. Сортировка списка по расходу топлива.

3. Запуск проекта в четные/нечетные дни месяца.

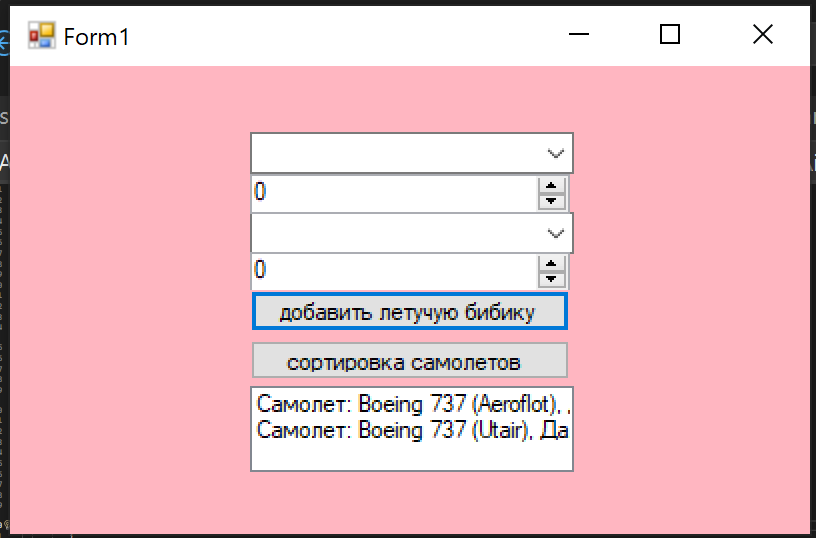


Рисунок 3 – Тест программы (до сортировки)

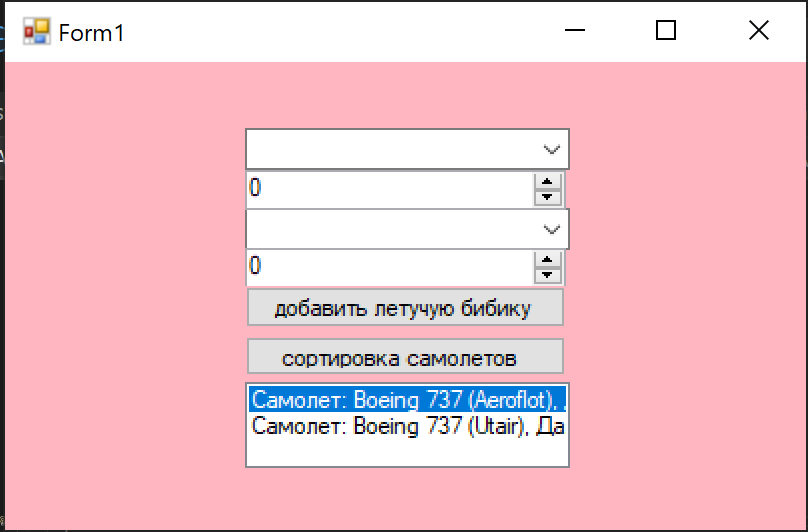


Рисунок 4 – Тест программы (после сортировки)

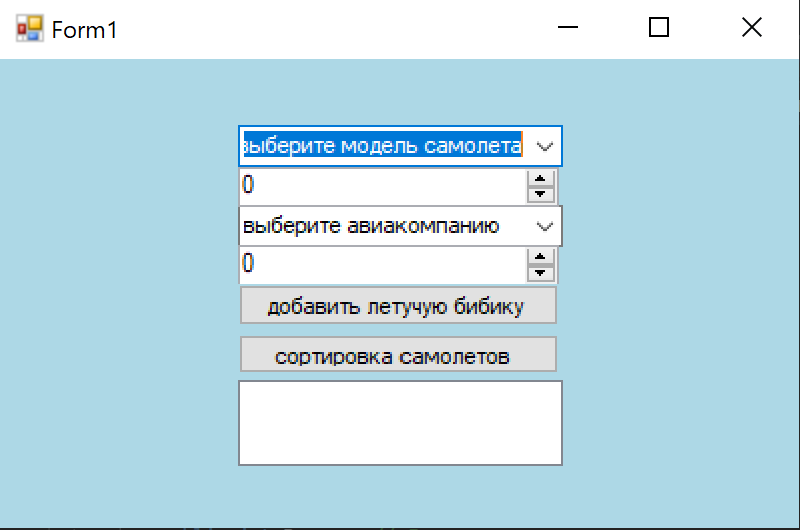


Рисунок 5 – Тест программы (в нечетный день)

**2.4 Интерфейс программы**

Приложение содержит элементы для добавления и сортировки самолётов:

ComboBoxModel — для выбора модели самолёта;

ComboBoxCompany — для выбора авиакомпании;

NumericUpDownRange — для ввода дальности полёта;

NumericUpDownFuelConsumption — для ввода расхода топлива;

ListBoxAirplanes — для отображения списка самолётов;

ButtonAddAirplane — для добавления самолёта;

ButtonSortByFuelConsumption — для сортировки самолётов.

В программе реализован конструктор без параметров в классе Airplane, который задаёт значения полей по умолчанию. Он может быть использован для создания самолётов без указания конкретных данных, что может быть полезно для тестирования или создания временных объектов.

public Airplane() : this("Unknown", 0, "Unknown", 0) { }

Данный конструктор присваивает модели самолёта значение "Unknown", дальность полёта — 0, авиакомпанию — "Unknown", и расход топлива — 0. Это упрощает создание объектов с минимальными данными, которые могут быть позднее изменены или дополнены.

Также в программе определена структура FlightData, которая хранит данные о вылете самолёта: город вылета, город прибытия и время вылета. Структуры, в отличие от классов, не поддерживают наследование, но могут быть полезны для хранения простых данных.

**3 Выводы**

В ходе лабораторной работы было создано Windows Forms приложение, которое реализует функции управления списком самолётов, их сортировки и отображения. Программа демонстрирует основные принципы объектно-ориентированного программирования, включая использование классов, перегрузку конструкторов, а также использование структур для хранения данных.