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 №6  (ВАРИАНТ №27)  МЕХАНИЗМЫ НАССЛЕДОВАНИЯ: КОНСТРУКТОРЫ ПРОИЗВОДНЫХ КЛАССОВ. |
|  |
| 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:

* Добавить необходимые конструкторы классу наследнику, учитывая оптимальные версии конструкторов базового класса.
* Добавить в новые классы необходимые поля, свойства, конструкторы.
* Разработать иерархию классов приложения.

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**

В программе "AirplaneAp5" реализована иерархия классов для управления самолетами, которая включает базовый класс Airplane и три дочерних класса: CargoAirplane (грузовой самолет), PassengerAirplane (пассажирский самолет) и BusinessClassAirplane (самолет бизнес-класса). Базовый класс Airplane содержит общие свойства самолетов: название, модель, дальность полета, расход топлива, дату производства и путь к фотографии. Дочерние классы расширяют базовый класс, добавляя специфичные свойства: CargoAirplane - грузоподъемность и тип груза, PassengerAirplane - вместимость и наличие бизнес-класса, BusinessClassAirplane - вместимость бизнес-класса, наличие отдельного лаунжа и список удобств.

Проект содержит также форму (Form1), которая реализует интерфейс IAirplaneManager для управления самолетами. Form1 предоставляет функциональность для добавления новых самолетов (грузовых, пассажирских и бизнес-класса) в список, выбора фотографии для самолета, очистки полей формы для ввода данных и обновления списка самолетов в элементе ListBox.

Интерфейс IAirplaneManager определяет контракт для класса, управляющего самолетами, и содержит методы для добавления, удаления, обновления, получения списка всех самолетов, получения самолета по названию, выбора фотографии и обновления списка самолетов.

Проект включает также реализацию метода ShowFoto для отображения фотографии самолета в элементе PictureBox и метода ToString для вывода информации о самолете в текстовом формате.

В настоящее время проект не содержит функциональность для удаления самолетов из списка, обновления информации о существующих самолетах, сохранения и загрузки данных в файл. Также нет визуализации плана мест в самолете.

В дальнейшем в проект можно добавить функциональность для удаления, обновления, сохранения и загрузки данных, а также ввести визуализацию плана мест.

**2.2 Листинг программы**

Класс Airplane5 на рисунках 1,2:

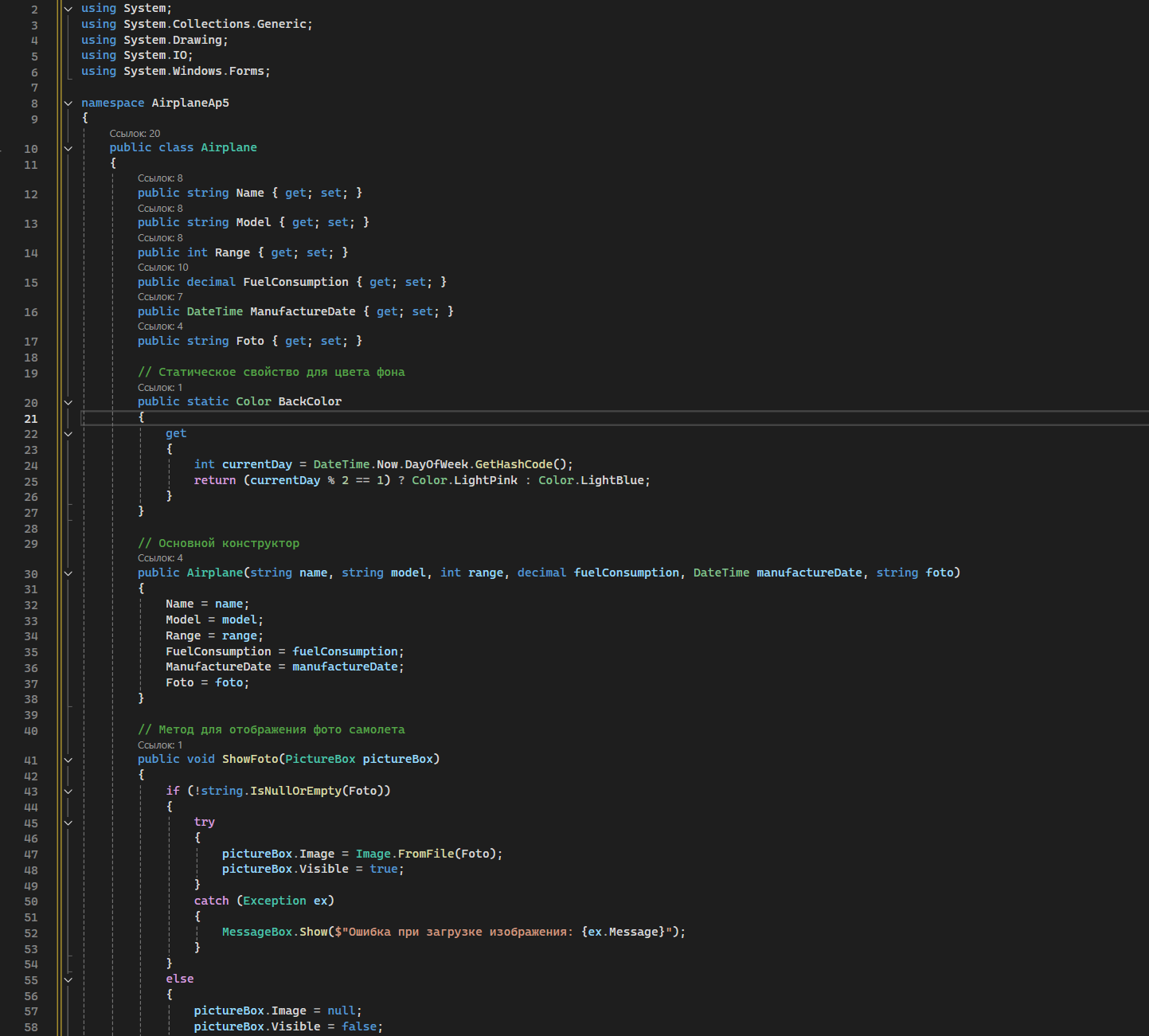


Рисунок 1 – Код класса Airplane

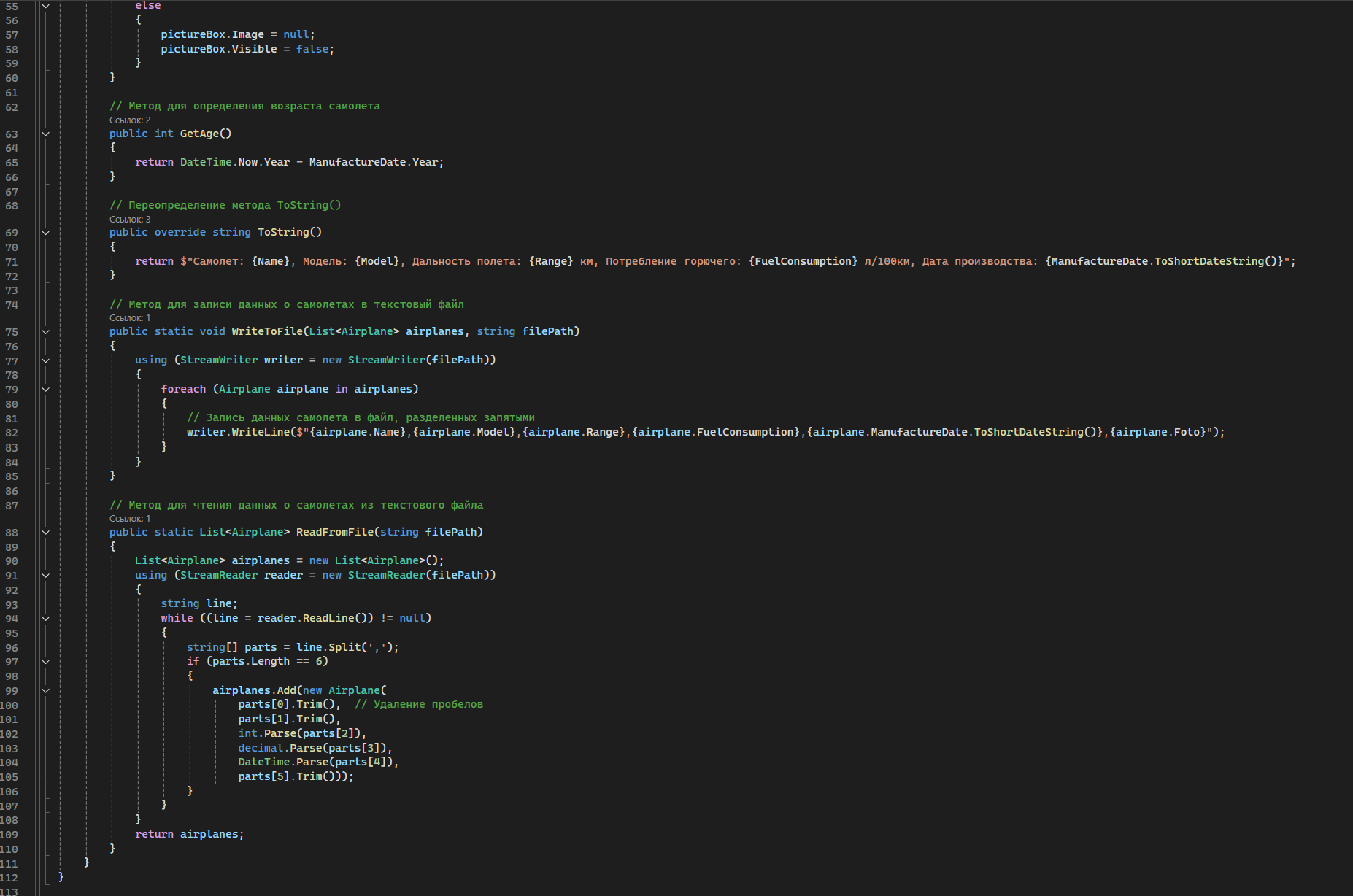


Рисунок 2 – Код класса Airplane

Код формы (Form1) на рисунках 3, 4, 5:



Рисунок 3 – Код формы (Form1)

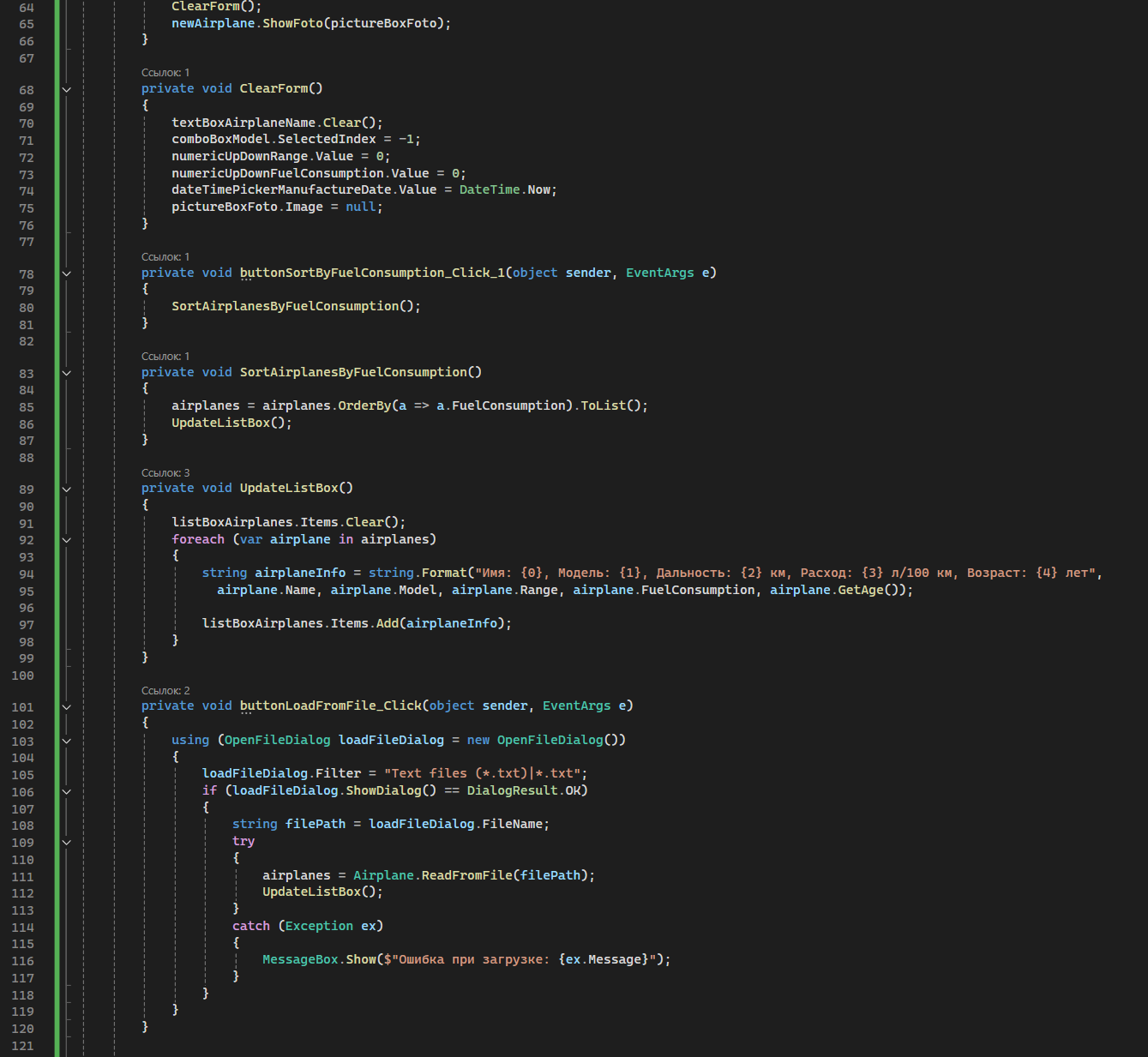


Рисунок 4 – Код формы (Form1)



Рисунок 5 – Код формы (Form1)

На рисунке 6 изображен код дочернего класса BusinessClassAirplane:

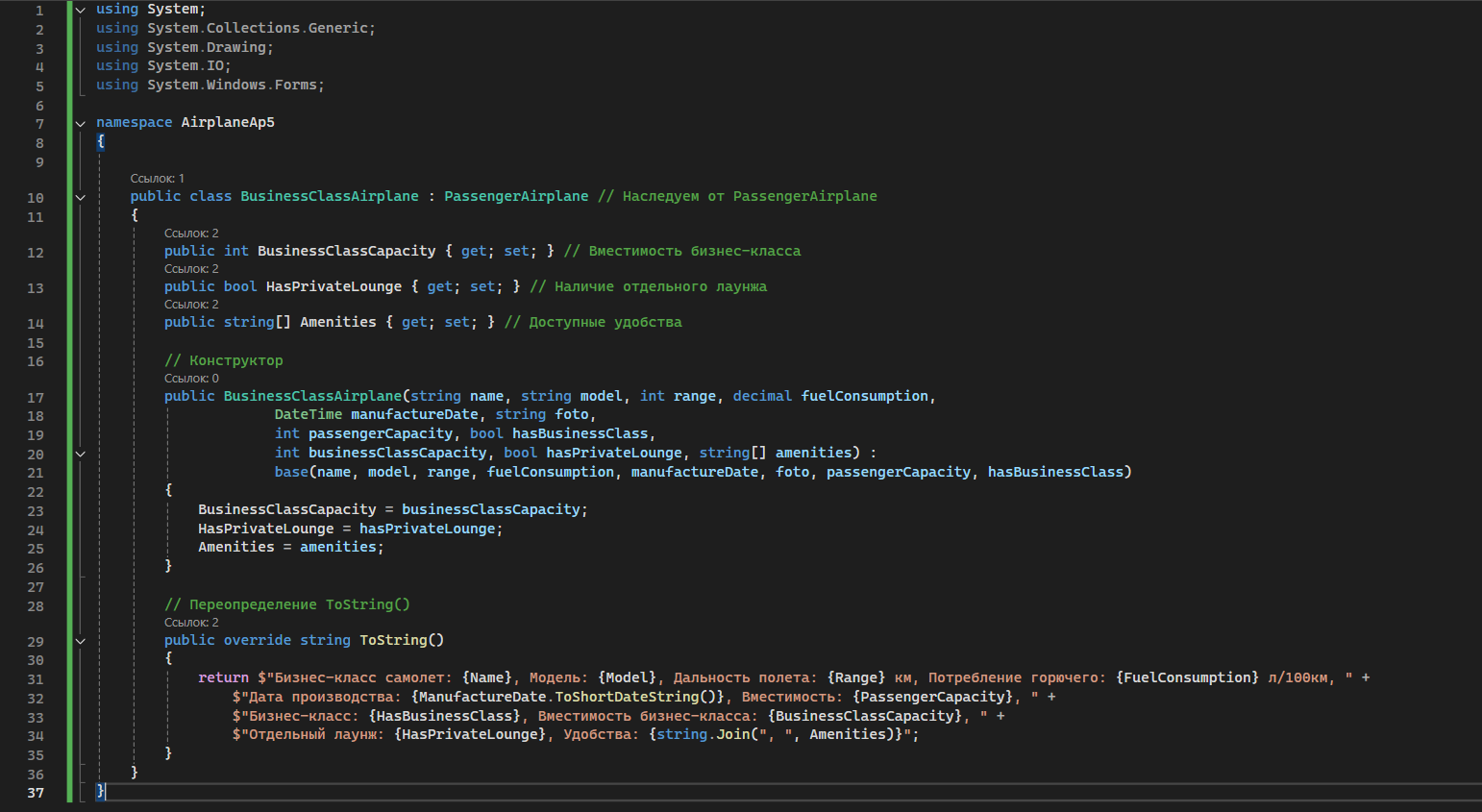


Рисунок 6 – Код дочернего класса BusinessClassAirplane

На рисунке 7 изображен код дочернего класса CargoAirplane:

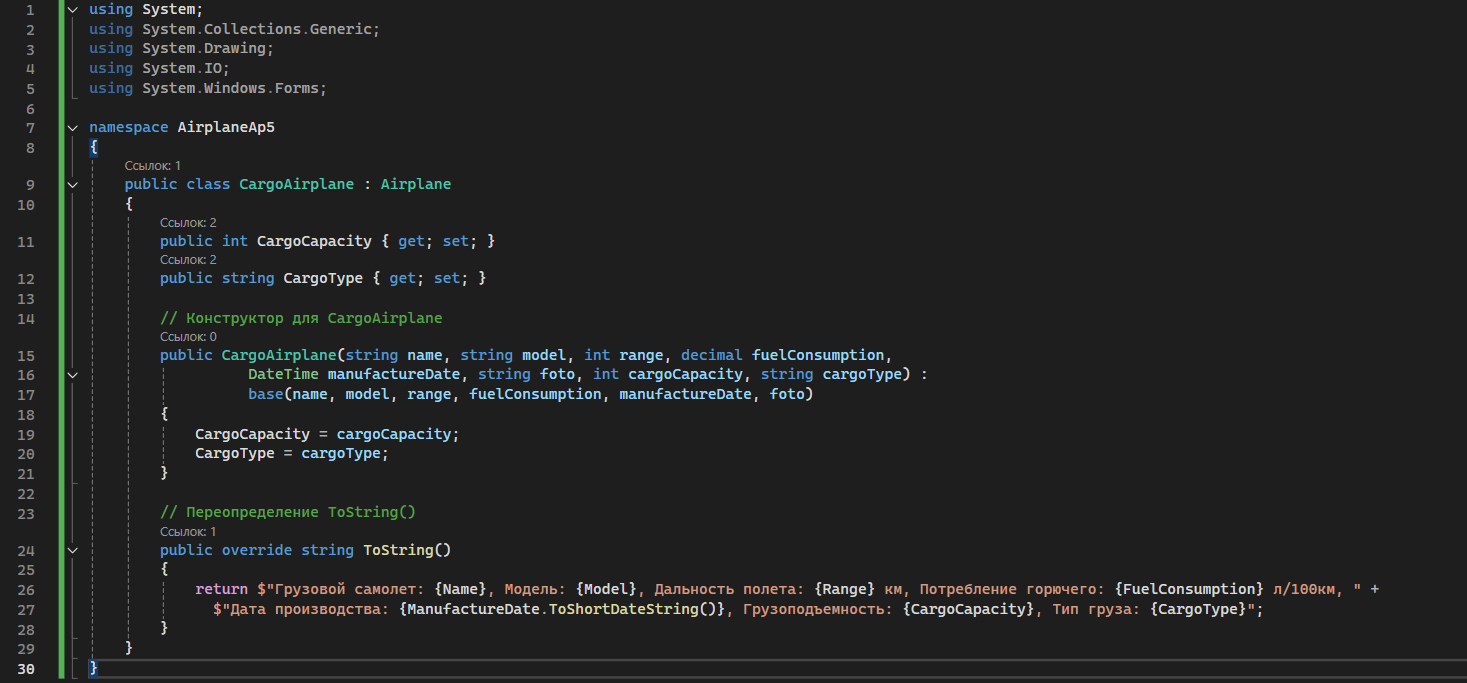


Рисунок 7 – Код дочернего класса CargoAirplane

На рисунке 8 изображен код дочернего класса PassengerAirplane:

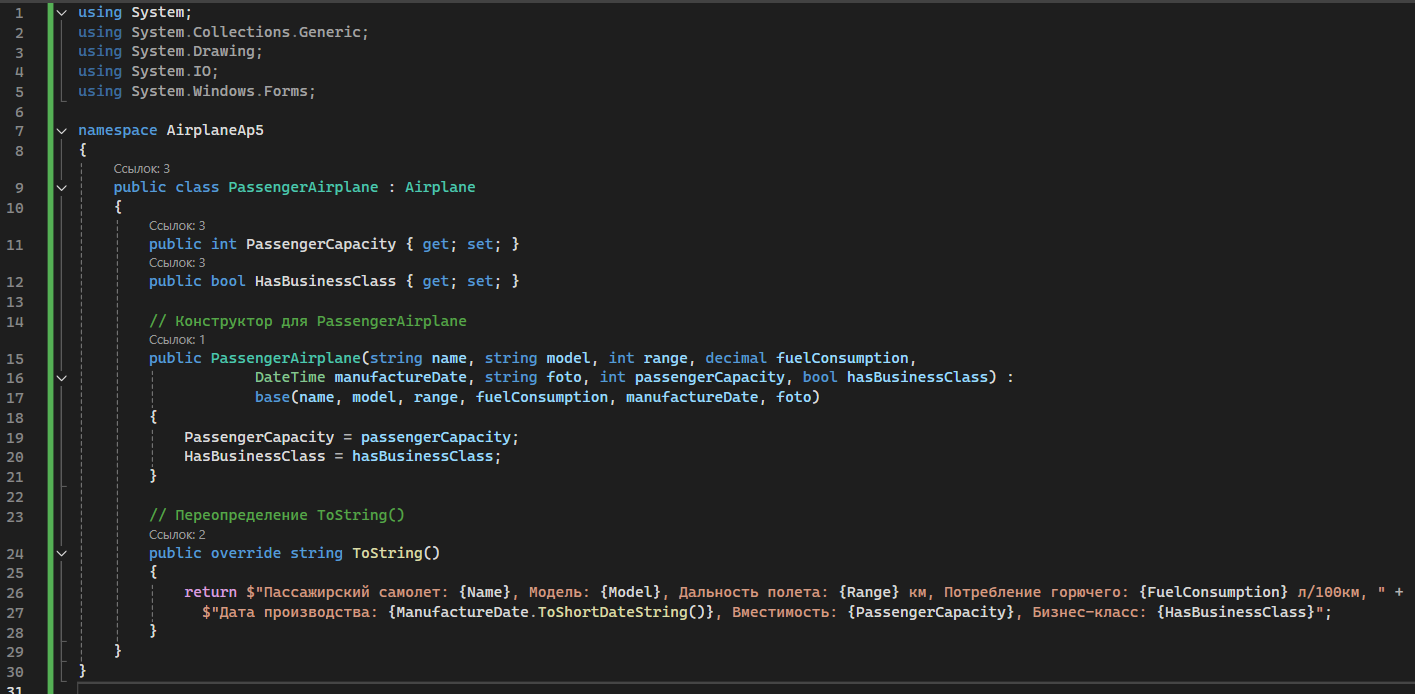


Рисунок 8 – Код дочернего класса PassengerAirplane

**2.3 Тесты работы программы**

Программа была протестирована на следующих сценариях:

1. Добавление данных о рразличных самолетах.

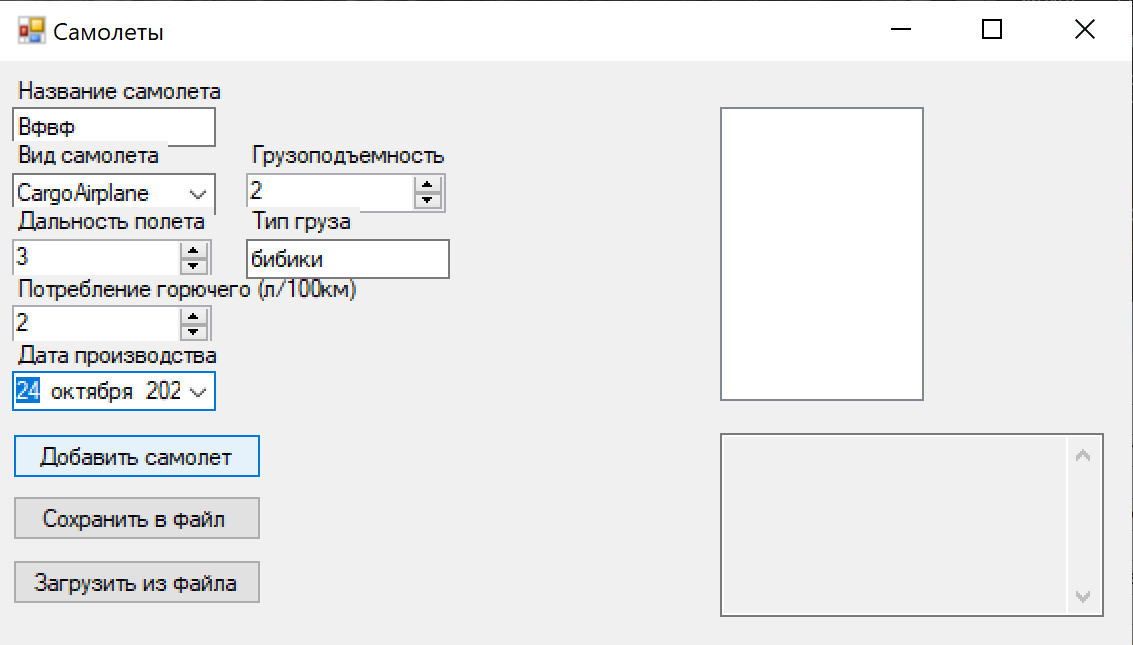


Рисунок 6 – Тест программы

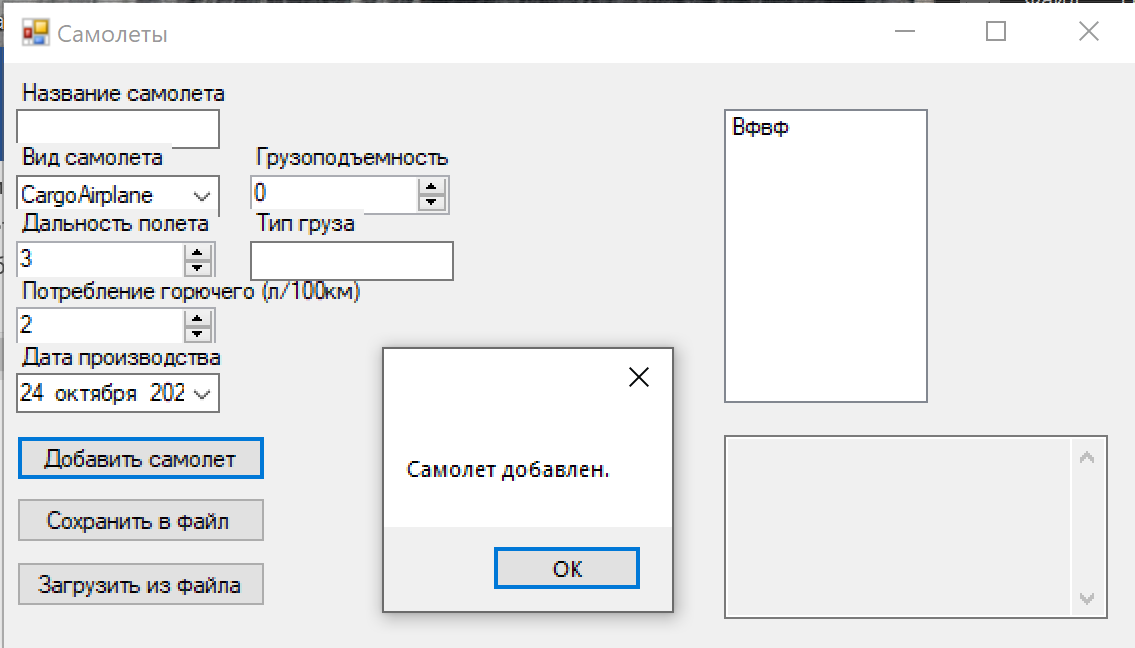


Рисунок 7 – Тест программы

На рисунке 8 изображена диаграмма классов.

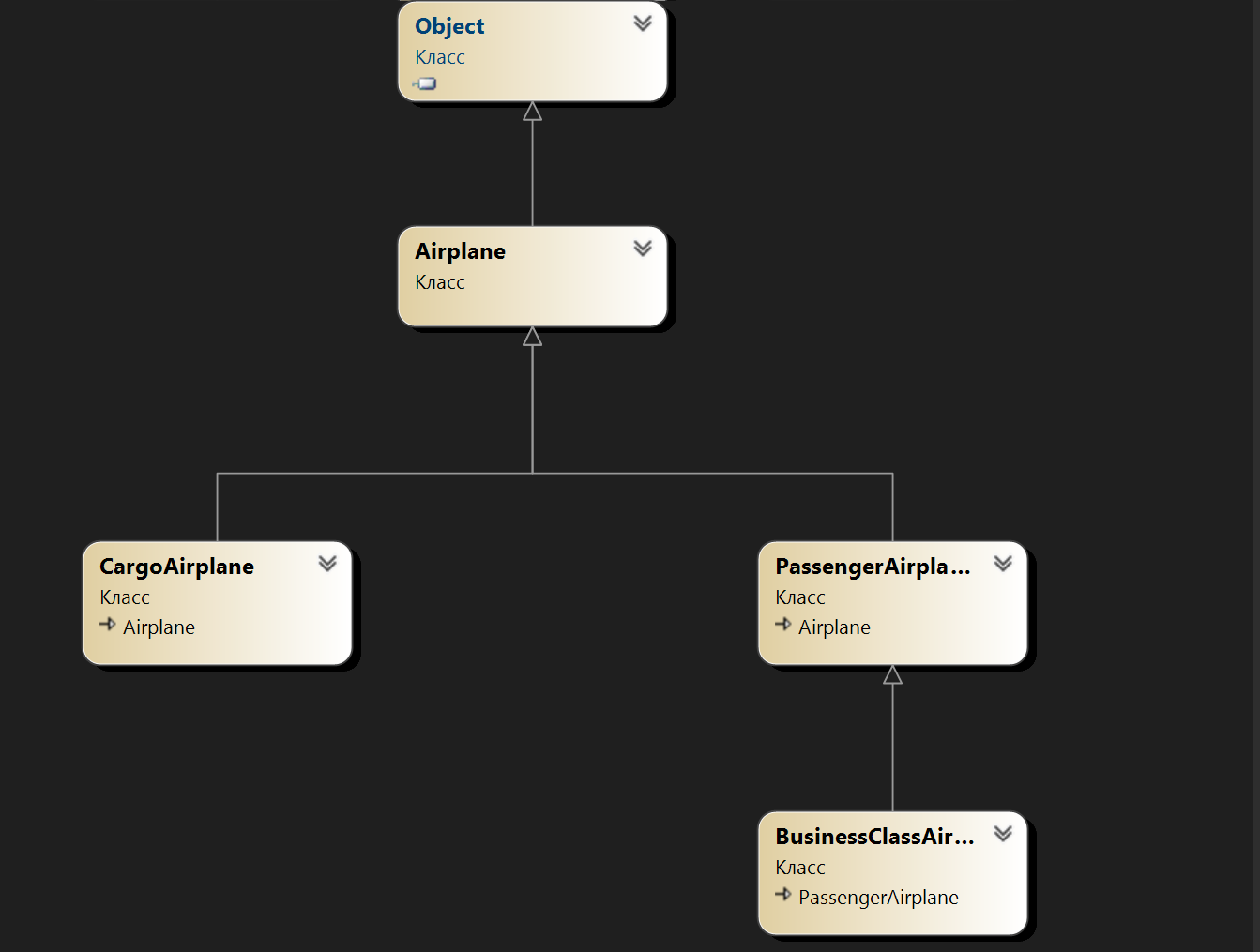


Рисунок 8 – Диаграмма классов

**2.4 Интерфейс программы**

Классы:

* Airplane: Базовый класс для всех типов самолетов. Он содержит основные характеристики: название, модель, дальность полета, расход топлива, дату производства и путь к фотографии.
* CargoAirplane: Дочерний класс, представляющий грузовой самолет. Наследует все свойства от Airplane и добавляет свойства: грузоподъемность (CargoCapacity) и тип груза (CargoType).
* PassengerAirplane: Дочерний класс, представляющий пассажирский самолет. Наследует все свойства от Airplane и добавляет свойства: вместимость (PassengerCapacity) и наличие бизнес-класса (HasBusinessClass).
* BusinessClassAirplane: Дочерний класс, представляющий самолет бизнес-класса. Наследует все свойства от PassengerAirplane и добавляет свойства: вместимость бизнес-класса (BusinessClassCapacity), наличие отдельного лаунжа (HasPrivateLounge) и список удобств (Amenities).

**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.