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 №8  (ВАРИАНТ №27)  ОПРЕДЕЛЕНИЕ И ВЫЗОВ СОБЫТИЙ. |
|  |
| 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:

Создать абстрактный класс Aircraft, который будет содержать общие свойства и методы для разных типов самолетов (например, название, модель, дальность, расход топлива и т. д.). Класс должен также включать абстрактные методы для получения информации и записи данных в файл.

Создать класс Airplane, который будет наследоваться от класса Aircraft. В нем необходимо реализовать статическое событие AirplaneAdded, которое будет генерироваться при добавлении нового самолета в систему. Событие должно вызывать цепочку обработчиков событий.

Реализовать обработчики событий. Необходимо протестировать работу статической и динамической подписки на событие, а также организовать цепочку обработчиков событий, которые будут выполняться при добавлении нового самолета. Каждый обработчик должен выводить соответствующие сообщения.

Реализовать методы для чтения и записи данных из файла. Класс Airplane должен уметь загружать информацию о самолетах из текстового файла, а также записывать информацию в файл. При чтении данных из файла необходимо генерировать событие AirplaneAdded, чтобы обработчики событий могли среагировать.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**

Программа реализует модель управления самолетами, где основная логика работы заключается в добавлении новых самолетов в список, отображении их информации, а также взаимодействии через события. Все самолетные объекты создаются на основе класса Airplane, который наследует абстрактный класс Aircraft. При добавлении нового самолета генерируется событие AirplaneAdded, которое затем вызывает обработчики событий. Для создания объектов используется форма с элементами управления, позволяющими пользователю вводить данные о самолетах. Событие AirplaneAdded поддерживает цепочку обработчиков, позволяя многократно обрабатывать событие. Подписка на событие может быть как статической, так и динамической, что позволяет гибко управлять реакцией на добавление самолета. Программа также позволяет загружать данные из текстового файла, где информация о самолетах хранится в формате CSV. Методы класса Airplane используют потоки ввода/вывода для чтения и записи данных в файл. В интерфейсе пользователю предоставляется возможность выбирать тип самолета (грузовой или пассажирский), вводить его характеристики, а также отображать данные о самолетах в элементе ListBox. Интерфейс поддерживает настройку внешнего вида (шрифт и цвет) через диалоговые окна.

**2.2 Листинг программы**

Класс Airplane на рисунках 1,2:

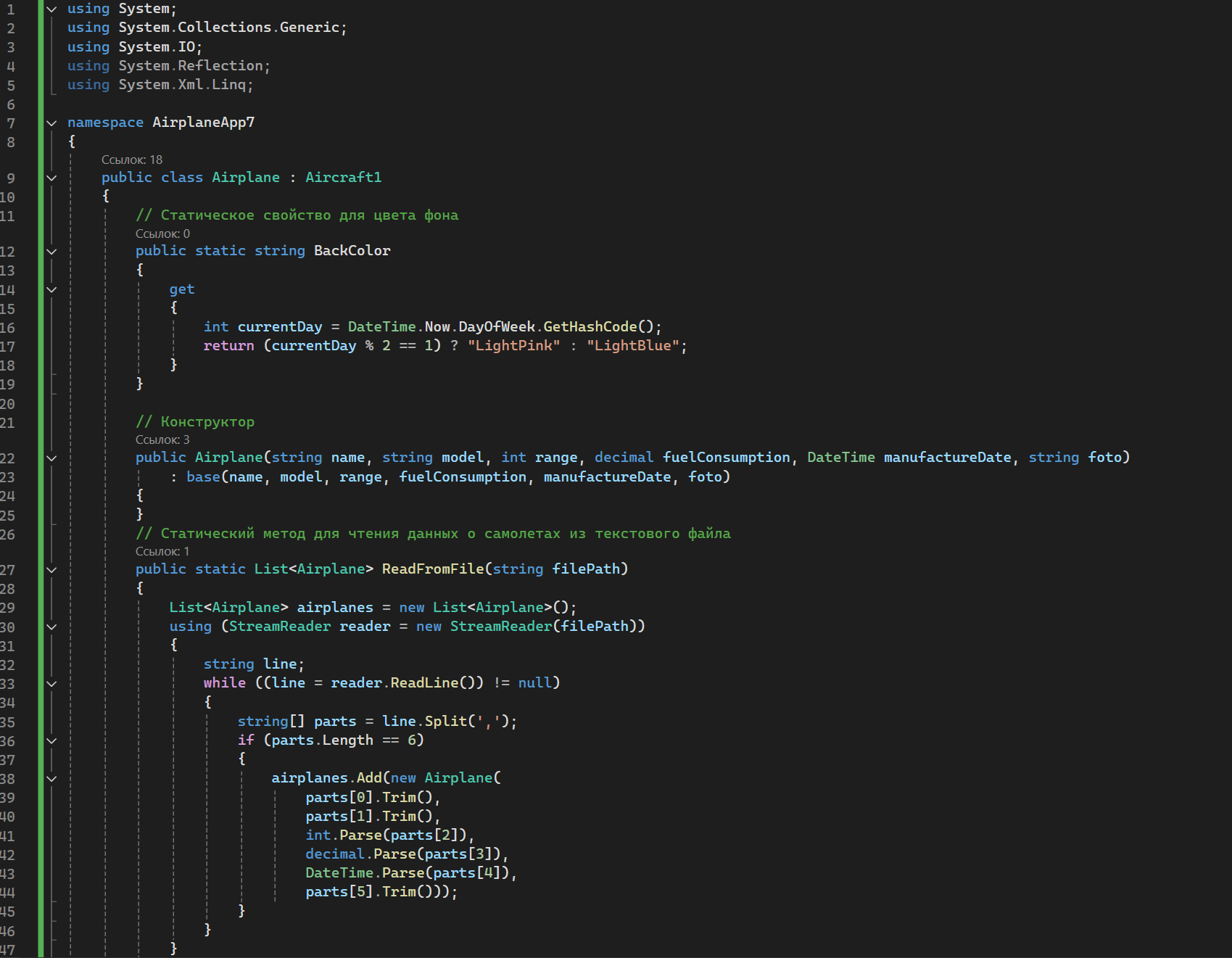


Рисунок 1 – Код класса Airplane

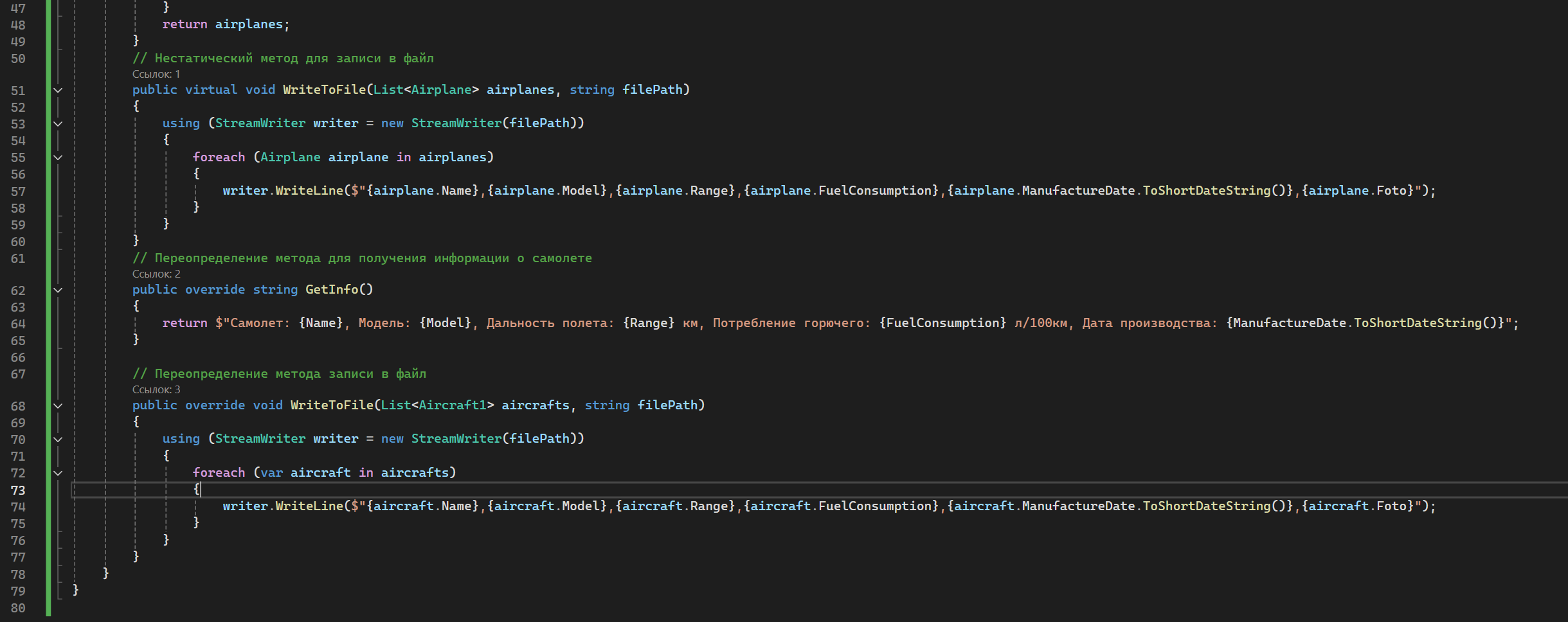


Рисунок 2 – Код класса Airplane

Код формы (Form1) на рисунках 3, 4, 5, 6:

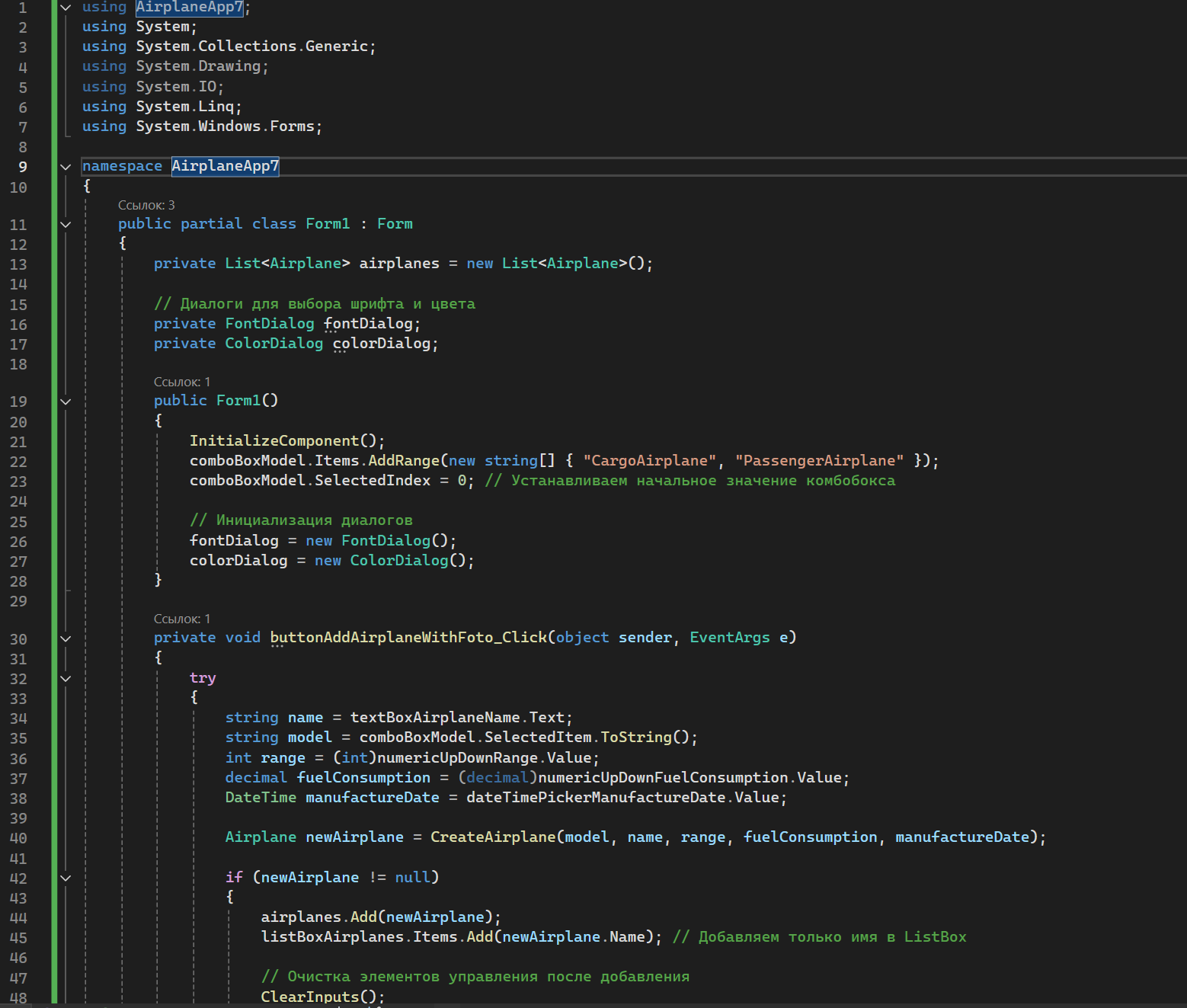


Рисунок 3 – Код формы (Form1)

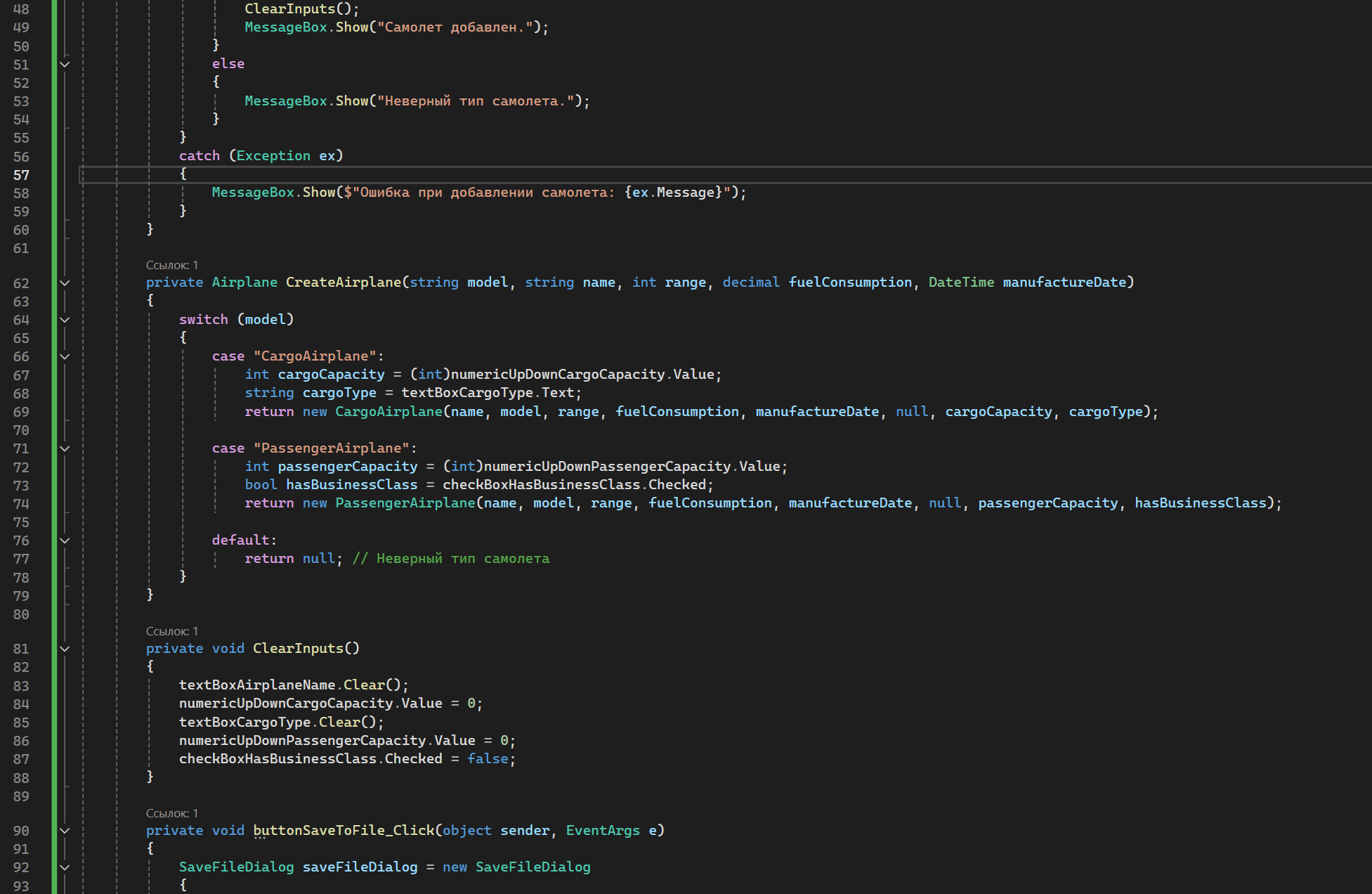


Рисунок 4 – Код формы (Form1)



Рисунок 5 – Код формы (Form1)

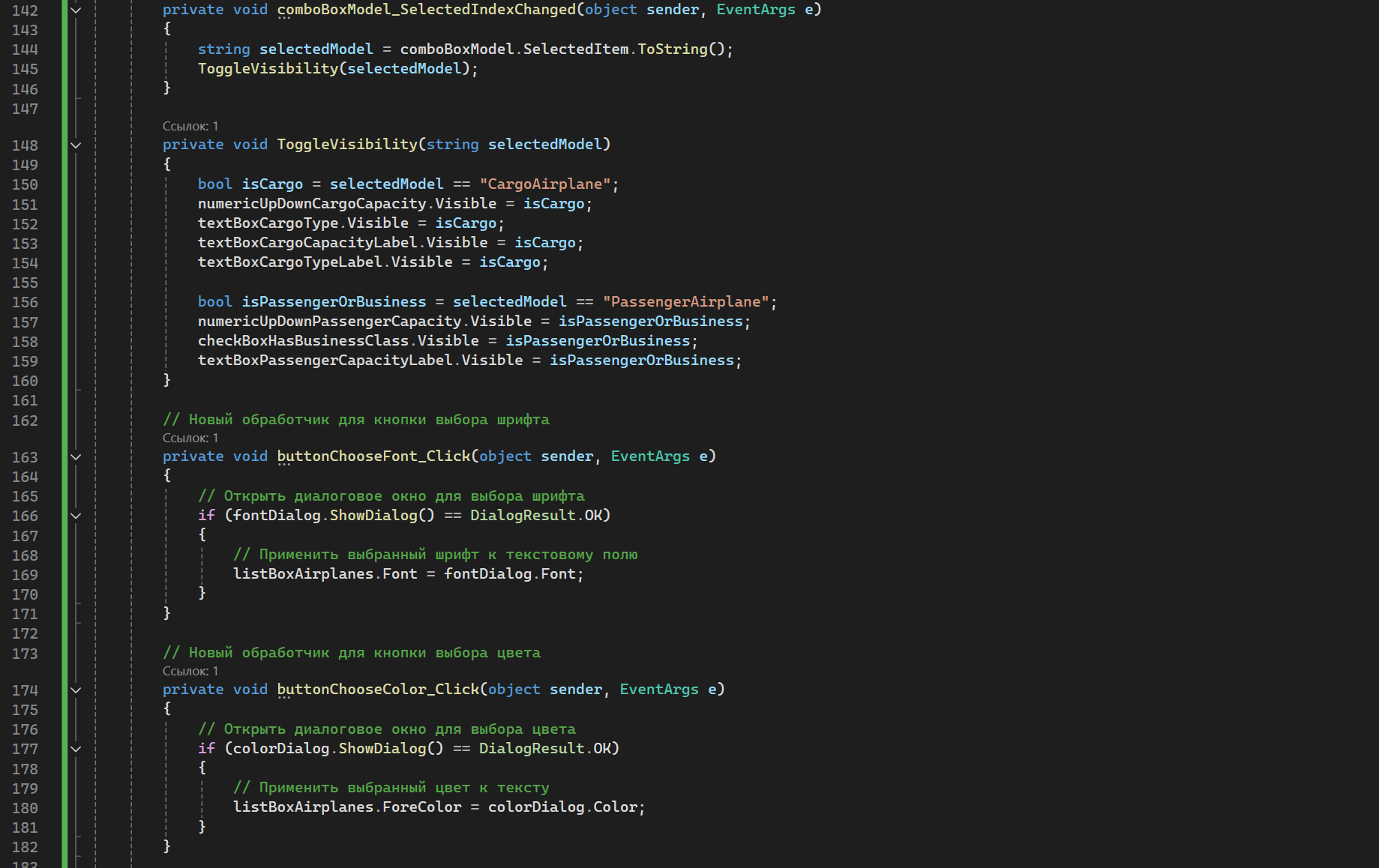


Рисунок 6 – Код формы (Form1)

На рисунке 7 изображен код дочернего класса CargoAirplane:

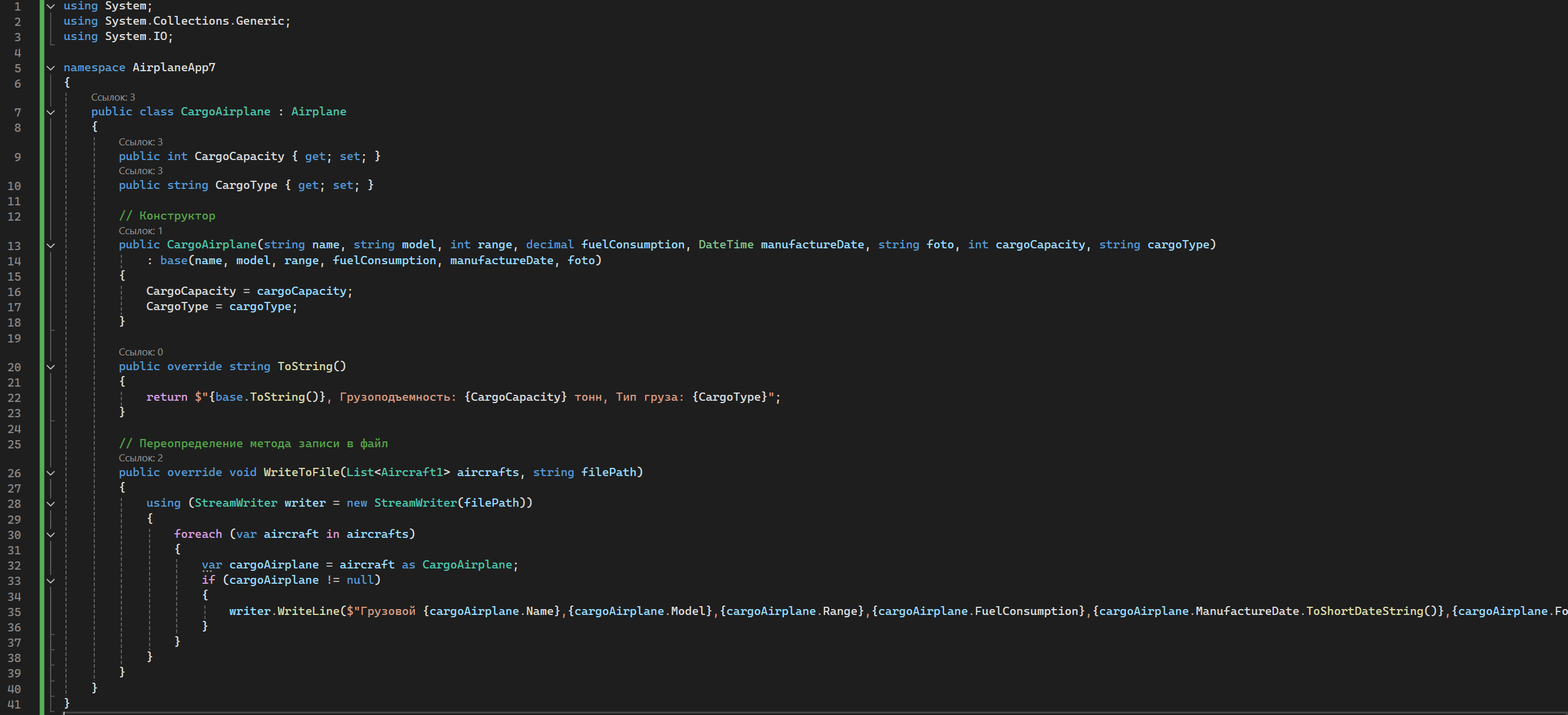


Рисунок 7 – Код дочернего класса CargoAirplane

На рисунке 8 изображен код дочернего класса PassengerAirplane:

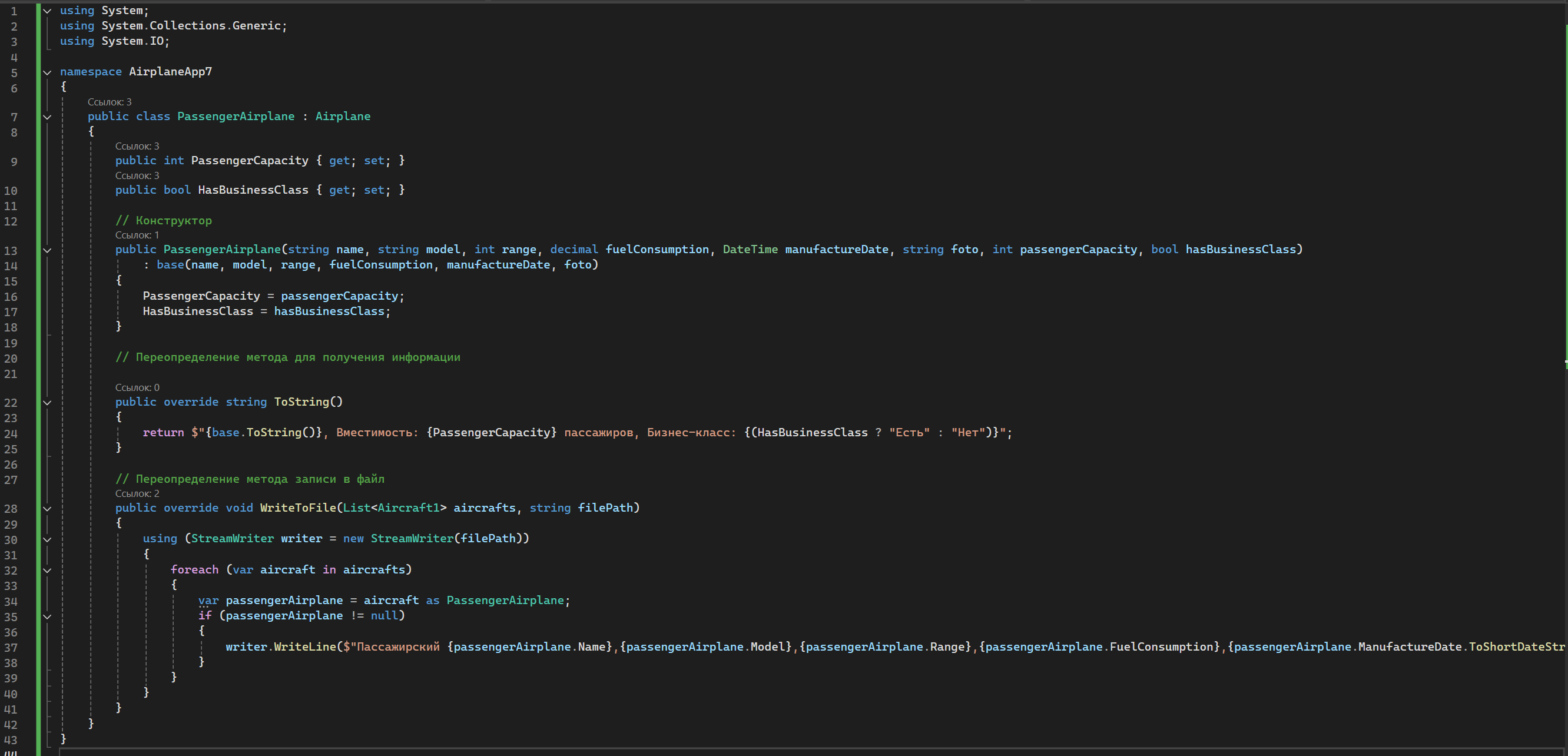


Рисунок 8 – Код дочернего класса PassengerAirplane

На рисунке 9 изображен код абстрактного класса Aircraft1:

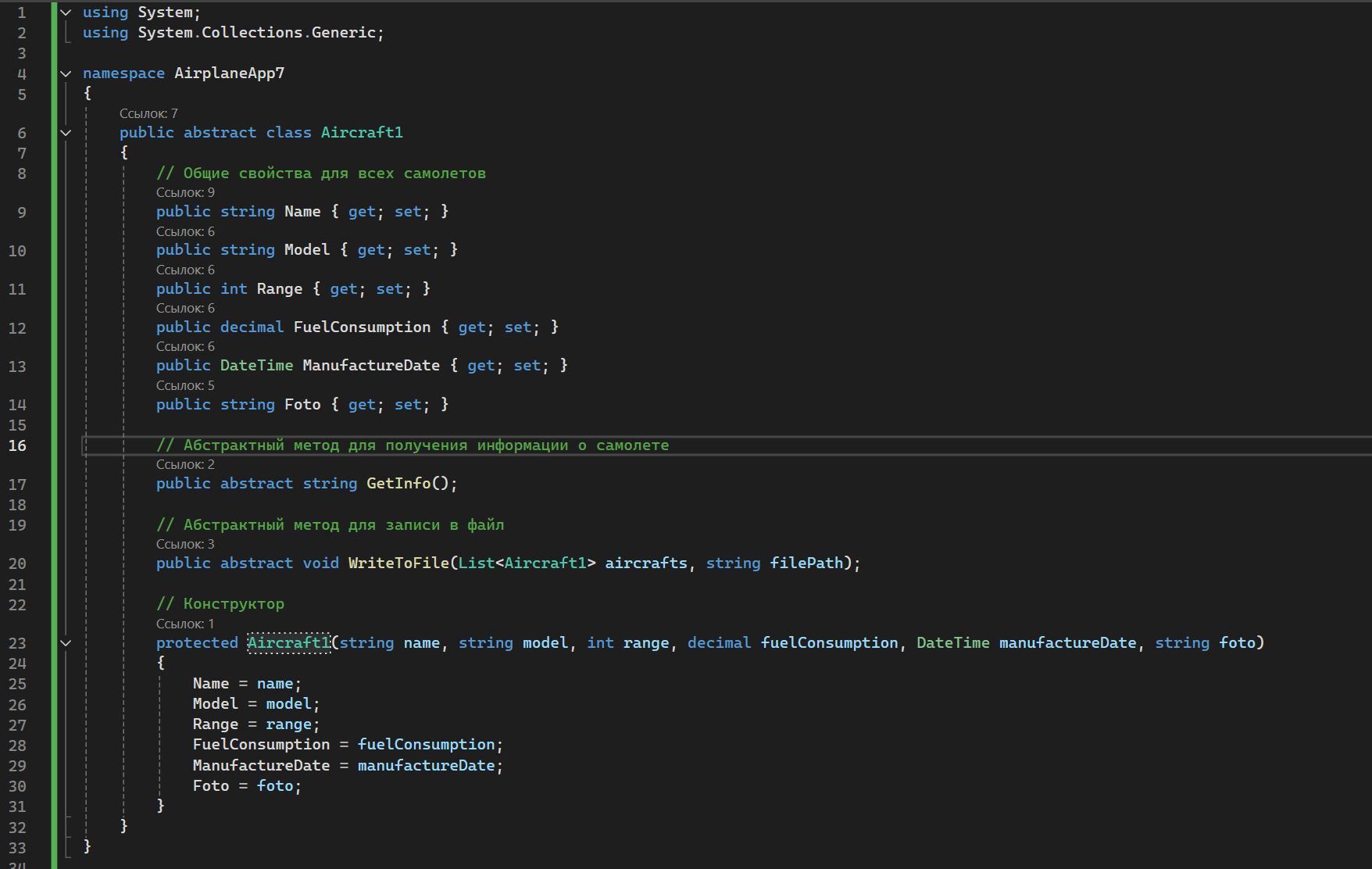


Рисунок 9 – Код абстрактного класса Aircraft1

**2.3 Тесты работы программы**

Программа была протестирована на следующих сценариях:

1. Добавление нового самолета

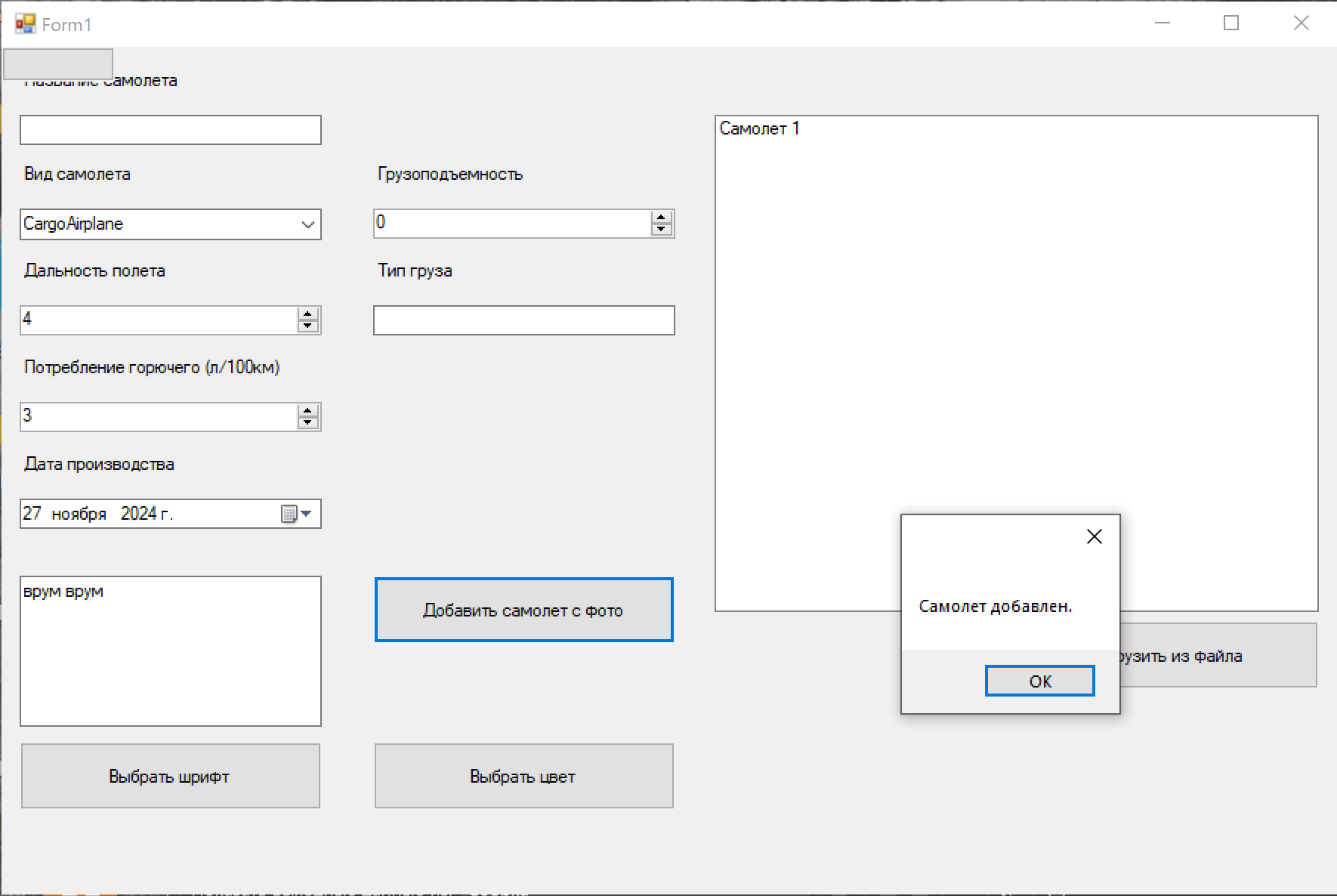


Рисунок 10 – Результат работы программы

* 1. **Интерфейс программы**

ComboBox (comboBoxModel) позволяет пользователю выбрать модель самолета (грузовой или пассажирский). В зависимости от выбранной модели появляются дополнительные поля для ввода данных, такие как грузоподъемность и тип груза для грузового самолета или пассажирская вместимость и наличие бизнес-класса для пассажирского самолета.

TextBox (textBoxAirplaneName) служит для ввода названия самолета.

NumericUpDown (numericUpDownRange) предназначен для ввода дальности полета самолета, а NumericUpDown (numericUpDownFuelConsumption) — для ввода расхода топлива.

DateTimePicker (dateTimePickerManufactureDate) позволяет выбрать дату производства самолета.

NumericUpDown (numericUpDownCargoCapacity) отображается, если выбран модель грузового самолета, и позволяет ввести грузоподъемность.

TextBox (textBoxCargoType) также доступен для модели "CargoAirplane" и служит для ввода типа груза.

NumericUpDown (numericUpDownPassengerCapacity) отображается, если выбрана модель пассажирского самолета, и позволяет ввести пассажирскую вместимость.

CheckBox (checkBoxHasBusinessClass) активируется для пассажирского самолета и дает возможность указать наличие бизнес-класса.

Button (buttonAddAirplaneWithFoto) используется для добавления нового самолета в список. После успешного добавления данные самолета сохраняются и выводится сообщение.

ListBox (listBoxAirplanes) отображает список имен добавленных самолетов. Пользователь может выбрать самолет из списка, чтобы увидеть подробную информацию о нем.

Button (buttonLoadFromFile) позволяет загрузить данные о самолетах из текстового файла. После загрузки данные отображаются в списке.

Button (buttonShowAirplaneInfo) используется для отображения подробной информации о выбранном самолете в текстовом поле.

Button (buttonChooseFont) открывает диалоговое окно для выбора шрифта текста, отображаемого в списке самолетов.

Button (buttonChooseColor) открывает диалоговое окно для выбора цвета текста в списке самолетов.

Label (textBoxCargoCapacityLabel, textBoxCargoTypeLabel, textBoxPassengerCapacityLabel, checkBoxHasBusinessClassLabel) отображает подписи для соответствующих полей ввода, и эти метки могут быть скрыты или отображены в зависимости от выбранной модели самолета.

**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.