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)  ИЕРАРХИЯ КЛАССОВ. АБСТАКТНЫЕ КЛАССЫ. |
|  |
| 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 в собственную разрабатываемую библиотеку новый класс с

модификатором abstract. Описать в абстрактном классе любые, общие для всей иерархии, поля, свойства и методы, в том числе абстрактные методы и свойства.

Внести необходимые изменения в описание методов и свойств классов

наследников. Протестировать все внесенные изменения.

Разработайте собственные варианты переопределения метода ToString() в классах иерархии.

Протестировать метод NameText с возможностью выбора шрифта из диалогового окна

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 данной программе реализована функциональность для управления самолетами, включая добавление, сохранение и загрузку данных о самолетах, а также изменение их отображения в интерфейсе. Используется абстрактный класс Aircraft1, который задает общие свойства и методы для всех типов самолетов, таких как Airplane, PassengerAirplane и CargoAirplane. В этом классе определены абстрактные методы GetInfo() и WriteToFile(), которые реализуются в дочерних классах для отображения информации о самолете и записи данных в файл. Каждый тип самолета переопределяет метод ToString() для представления своей специфической информации: для пассажирских самолетов выводится информация о вместимости и наличии бизнес-класса, для грузовых — о грузоподъемности и типе груза. В форме пользователя реализованы элементы управления для ввода данных о самолете, выбора модели (пассажирский или грузовой), а также кнопки для добавления самолета в список, сохранения и загрузки данных из файлов. Пользователь также может выбирать шрифт для отображения списка самолетов с помощью FontDialog, что позволяет изменять шрифт в listBoxAirplanes. В коде предусмотрены обработчики ошибок для корректной работы с файлами и ввода данных.

**2.2 Листинг программы**

Класс Airplane на рисунках 1,2:

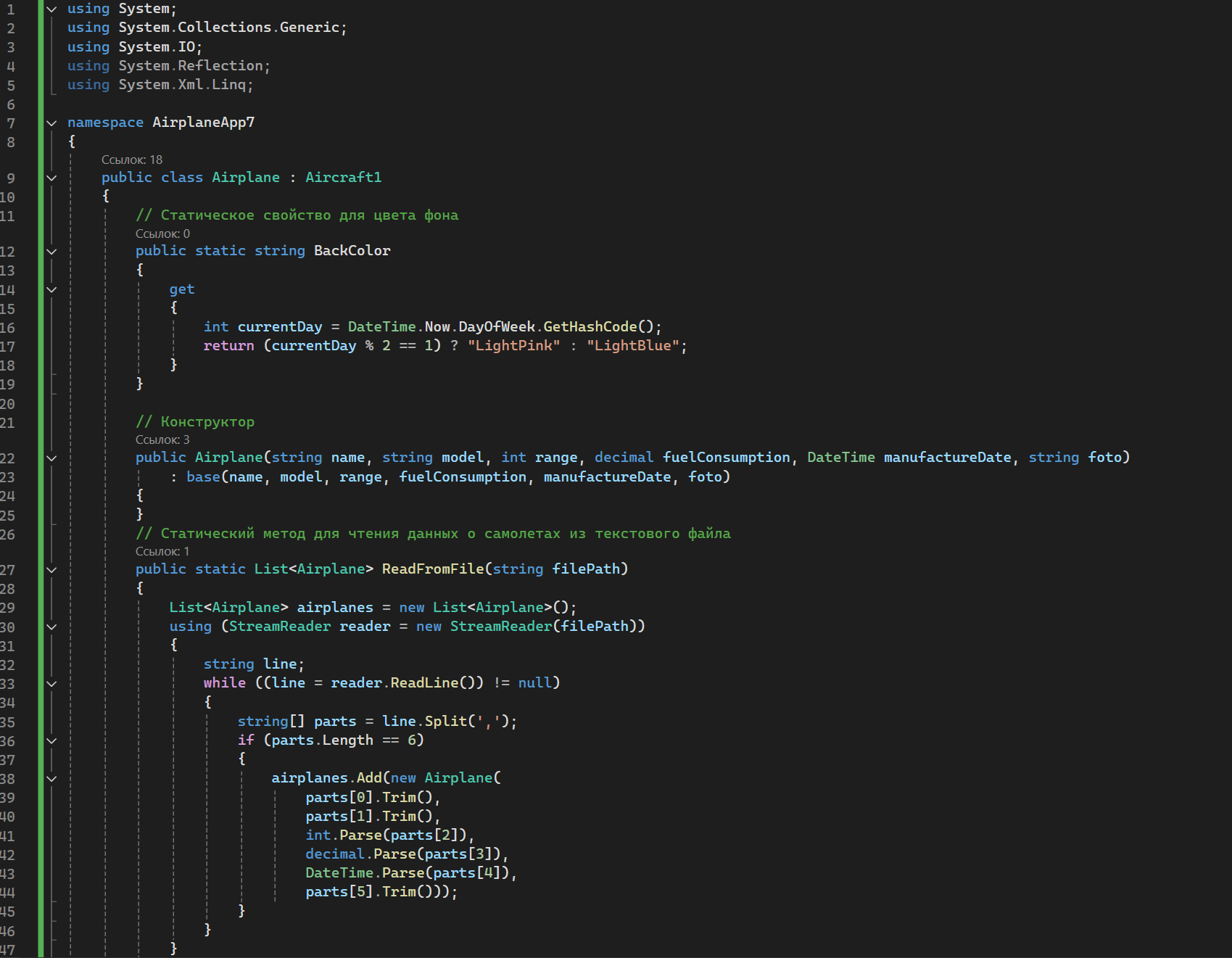


Рисунок 1 – Код класса Airplane

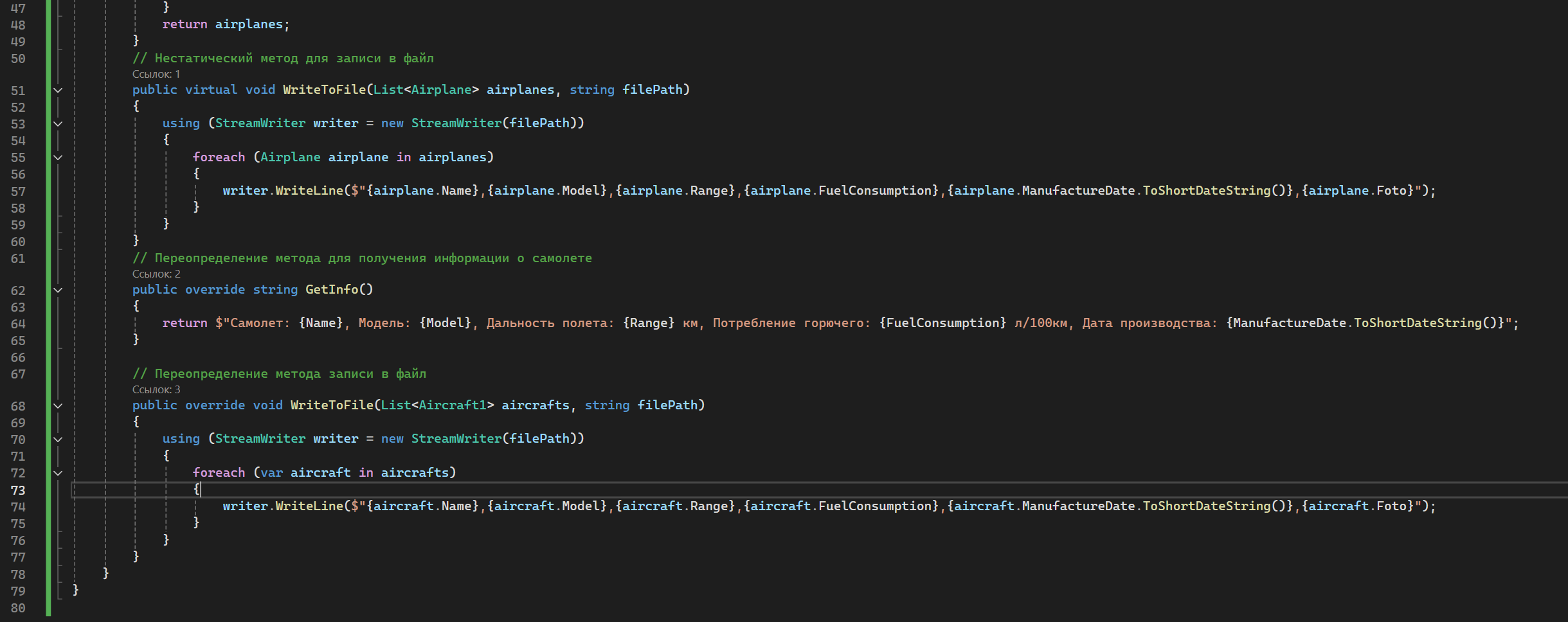


Рисунок 2 – Код класса Airplane

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

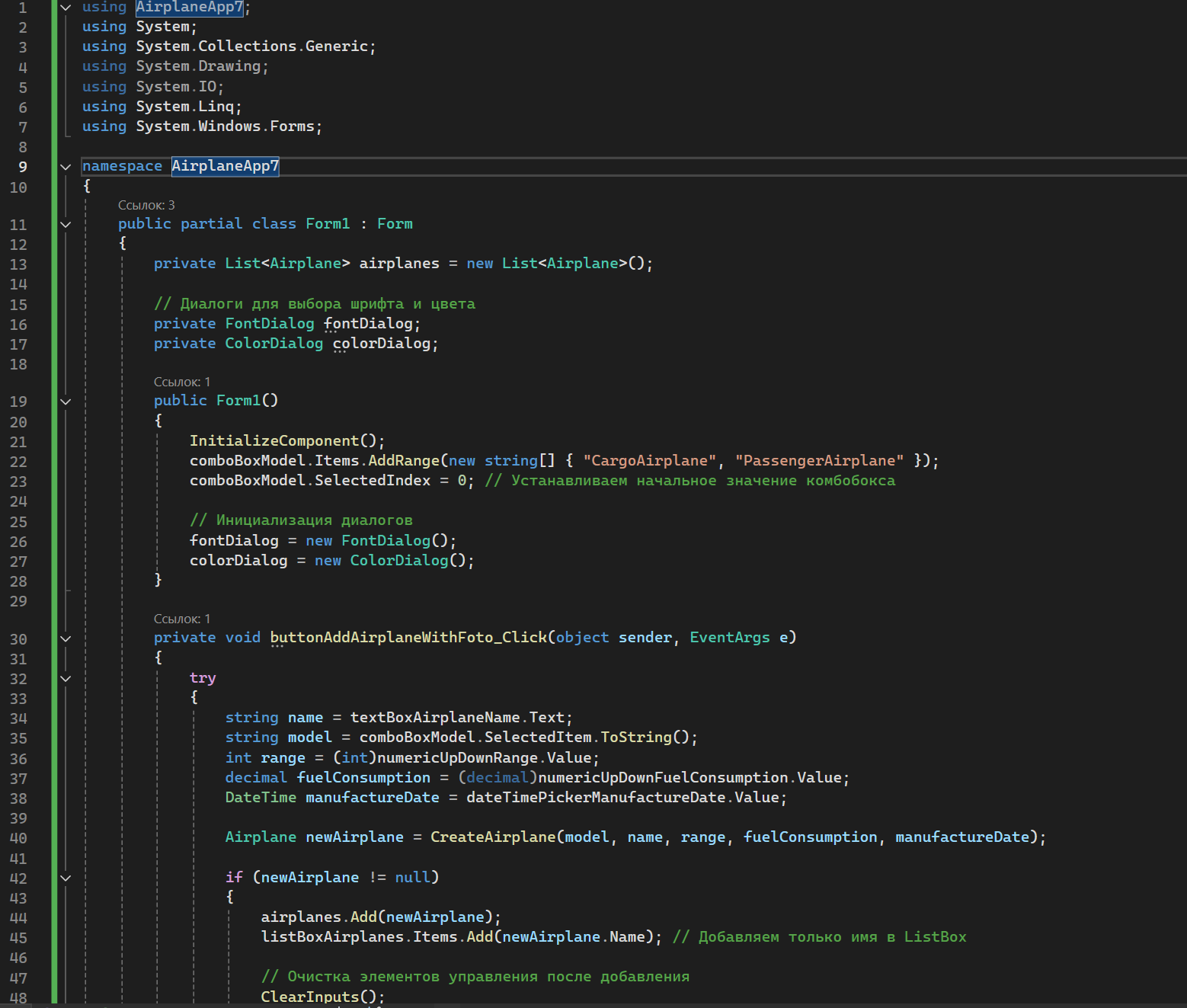


Рисунок 3 – Код формы (Form1)

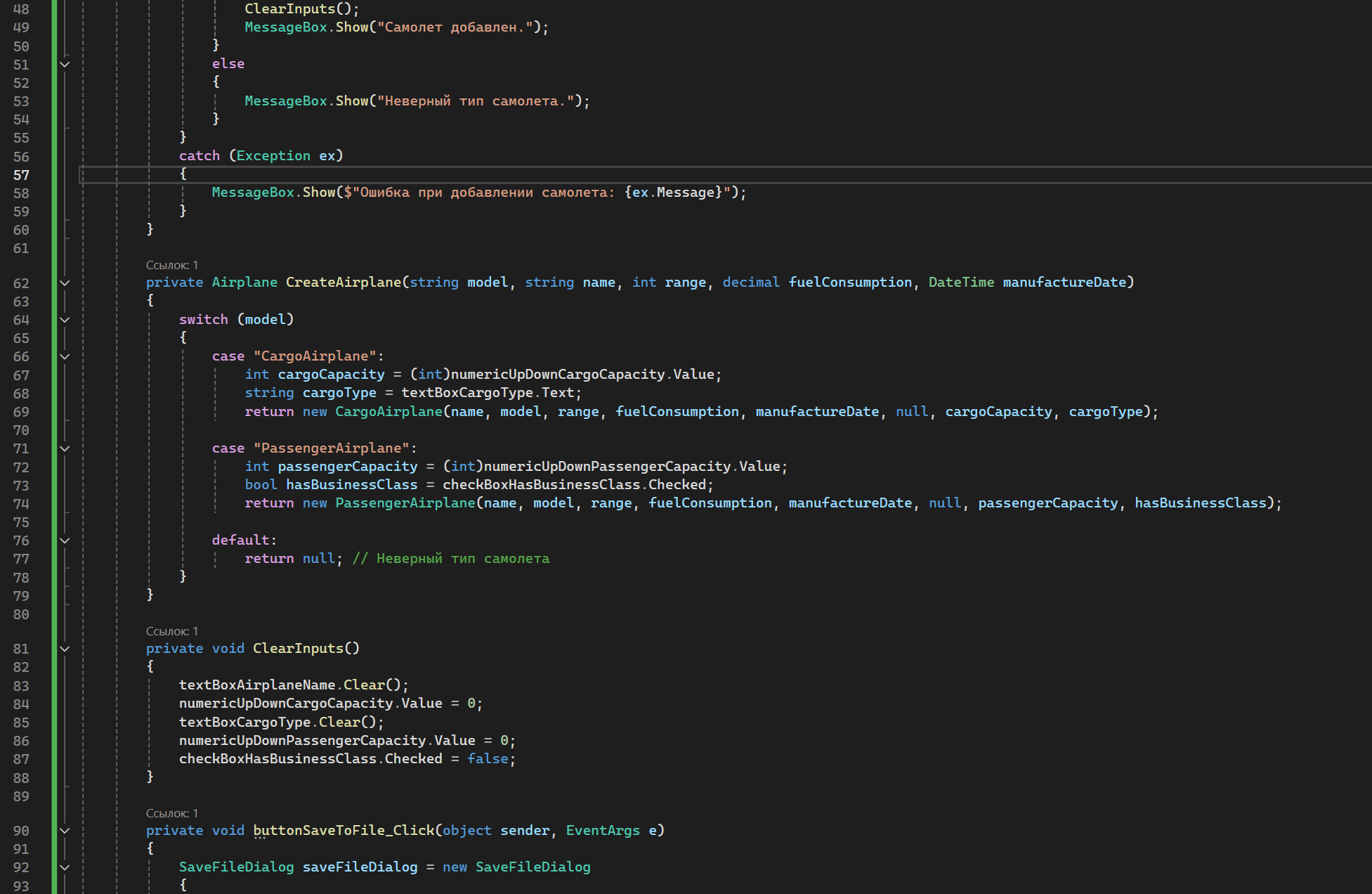


Рисунок 4 – Код формы (Form1)



Рисунок 5 – Код формы (Form1)

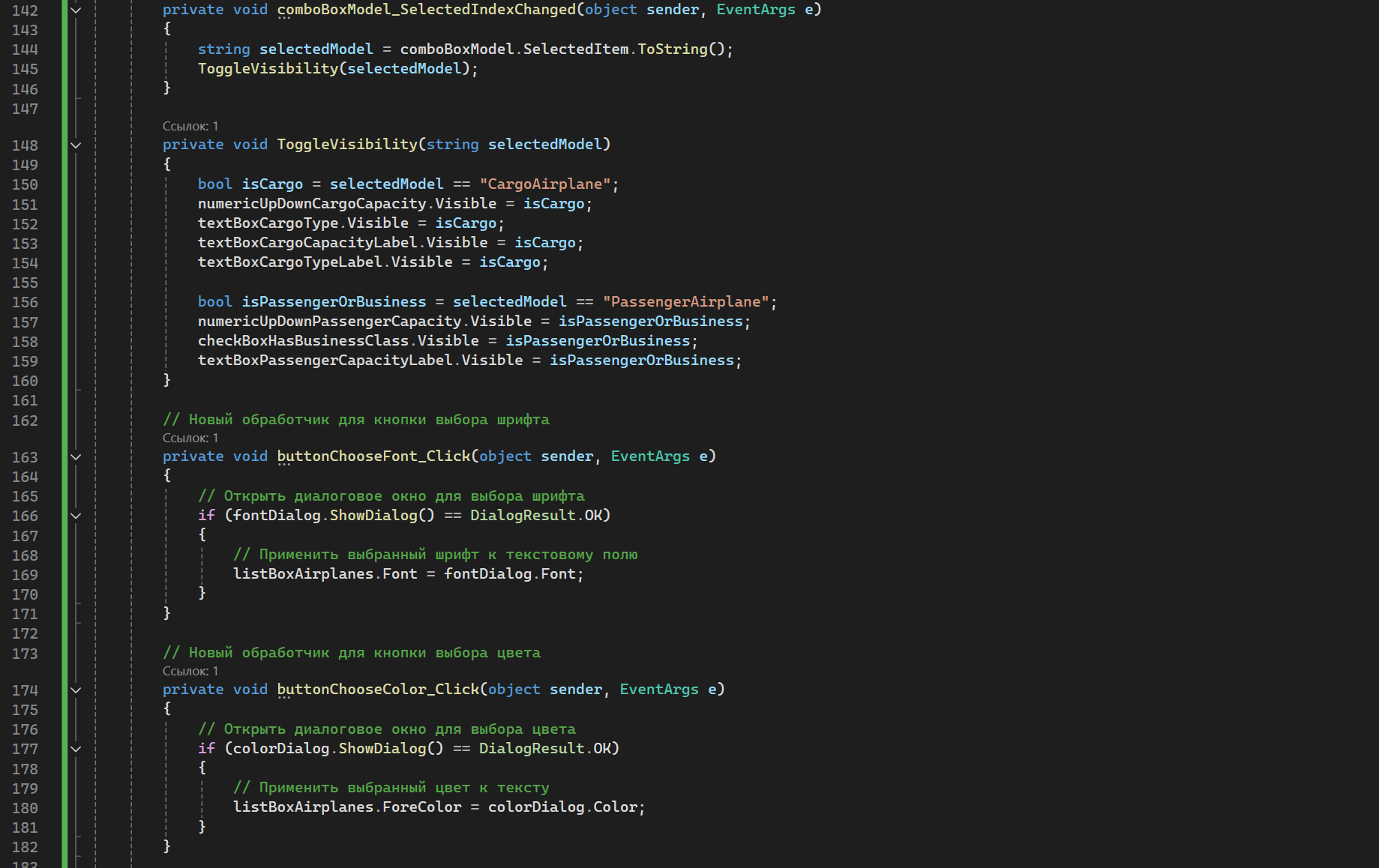


Рисунок 6 – Код формы (Form1)

На рисунке 7 изображен код дочернего класса CargoAirplane:

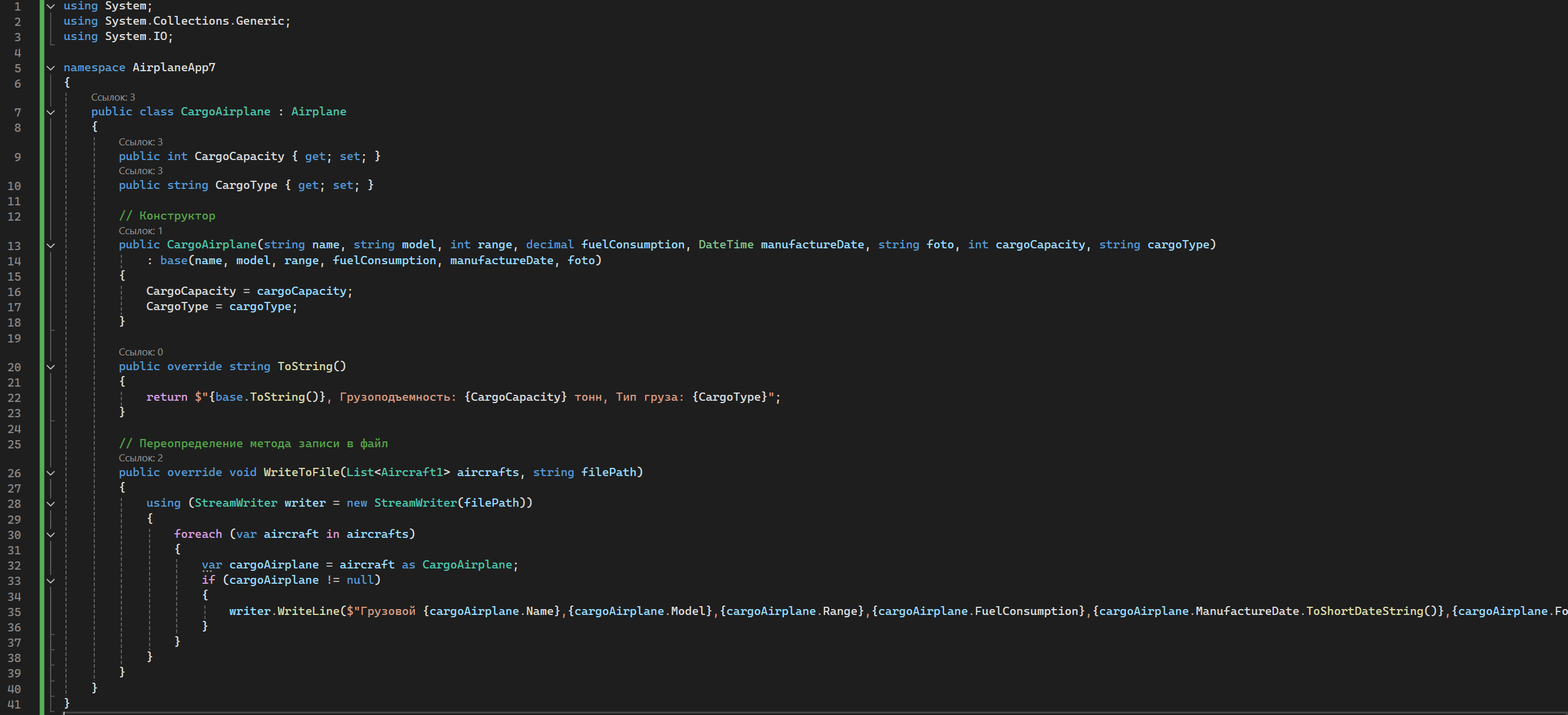


Рисунок 7 – Код дочернего класса CargoAirplane

На рисунке 8 изображен код дочернего класса PassengerAirplane:

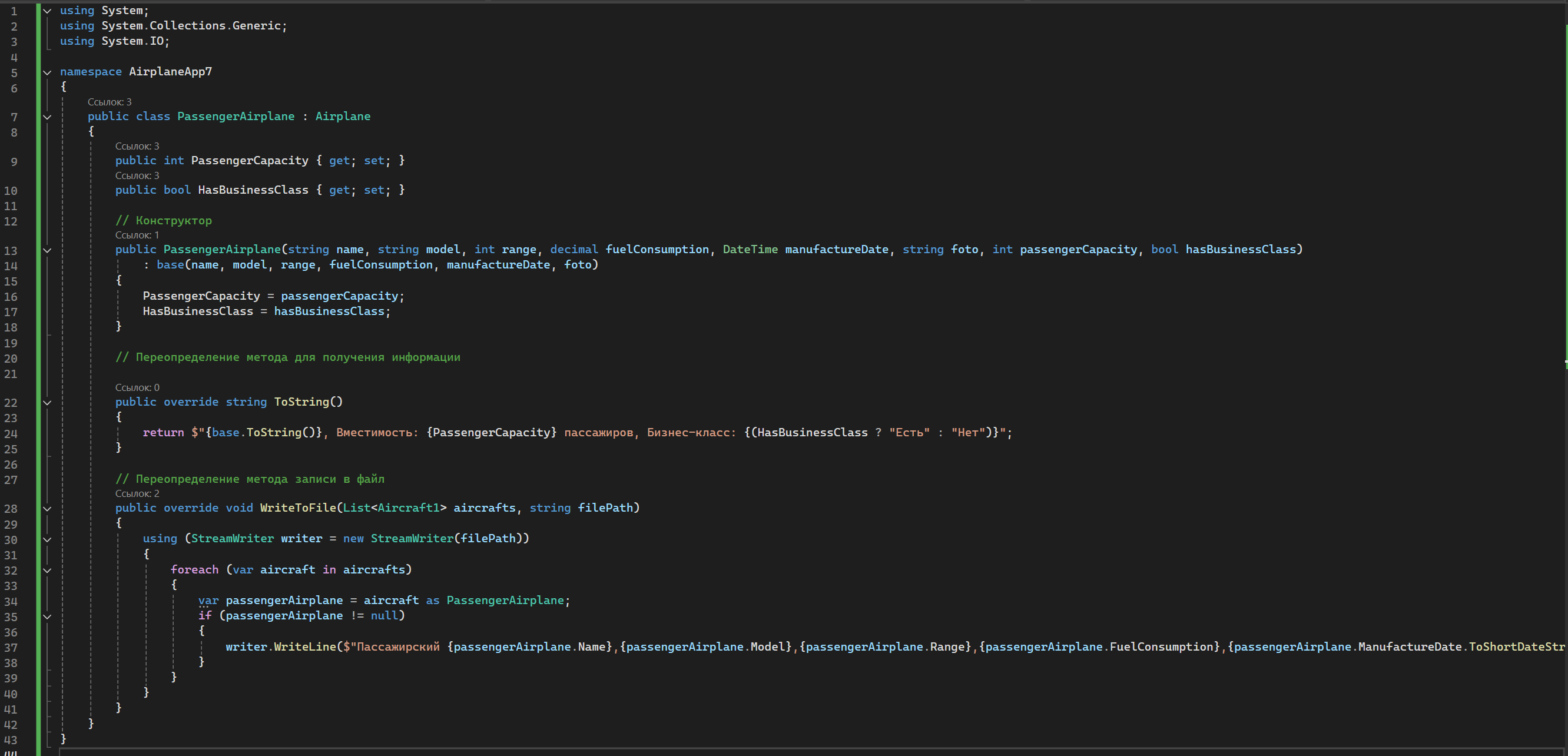


Рисунок 8 – Код дочернего класса PassengerAirplane

На рисунке 9 изображен код абстрактного класса Aircraft1:

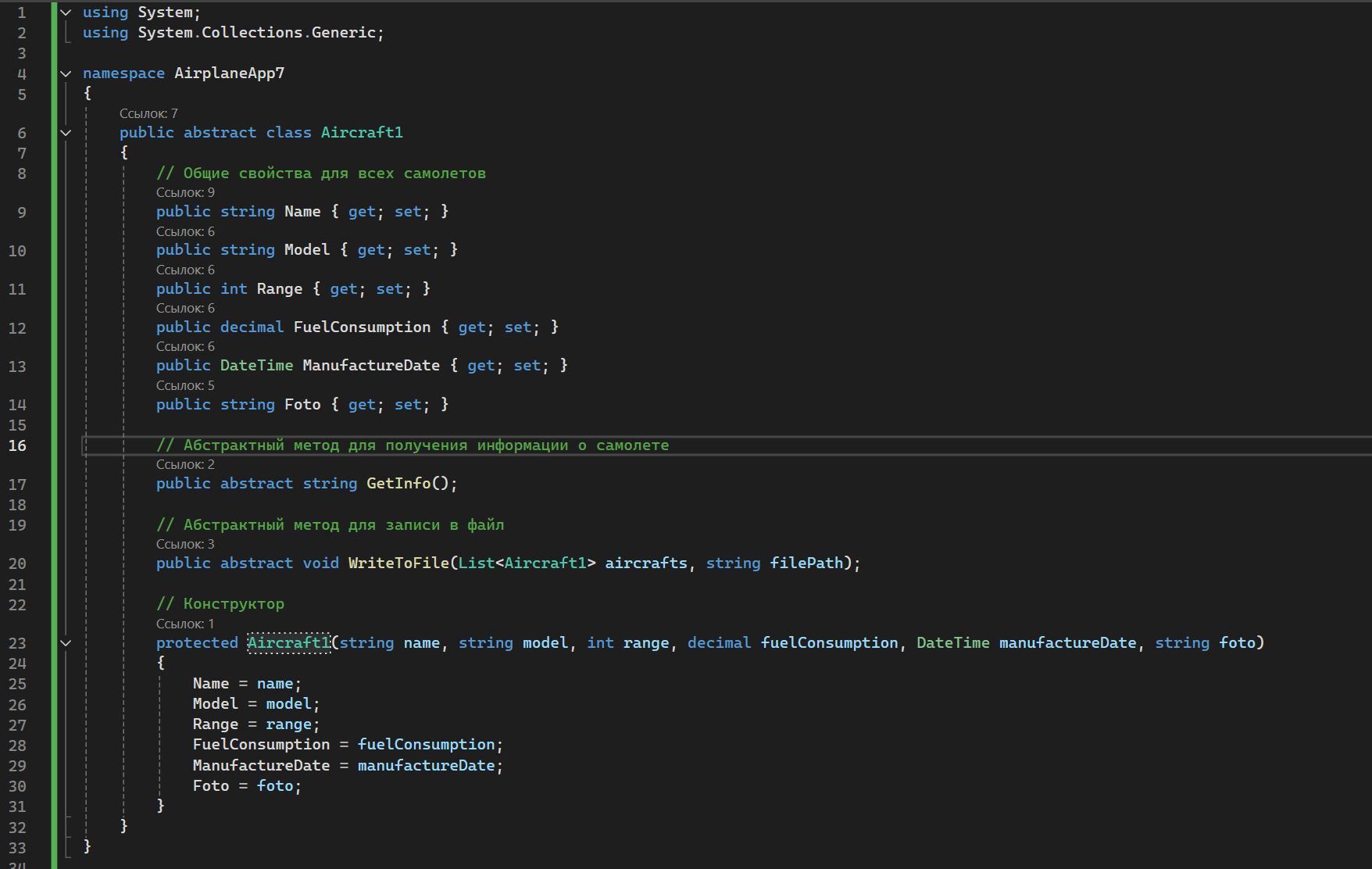


Рисунок 9 – Код абстрактного класса Aircraft1

**2.3 Тесты работы программы**

Программа была протестирована на следующих сценариях:

1. Смена шрифта и цвета текста для списка добавленных самолетов.

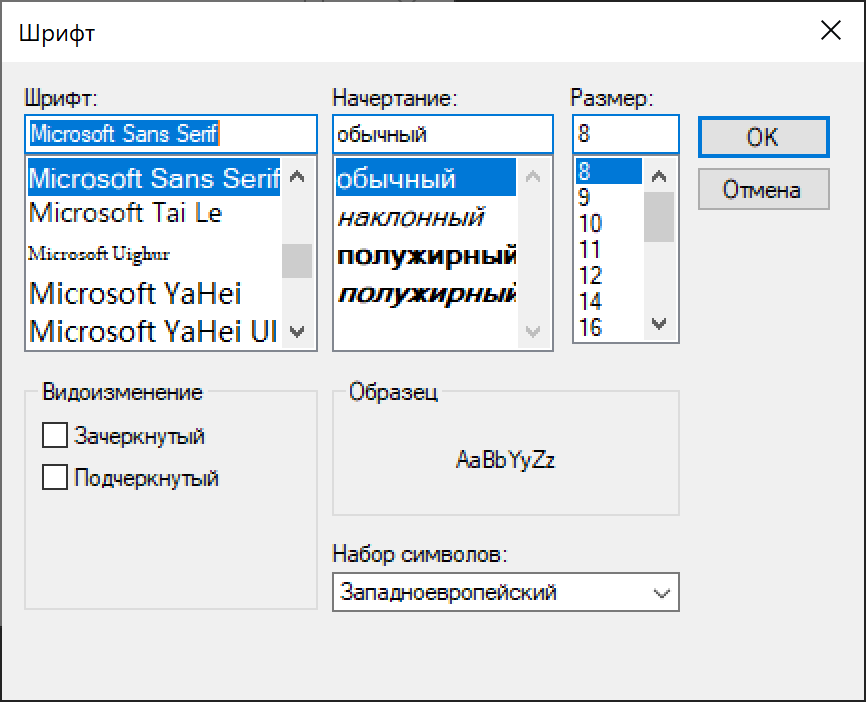


Рисунок 10 – Тест программы

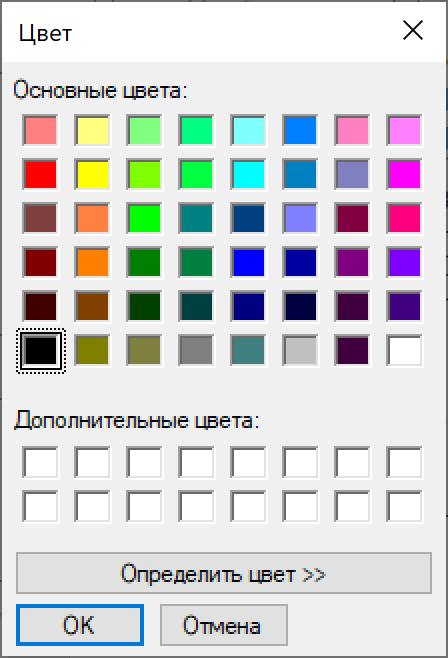


Рисунок 11 – Тест программы

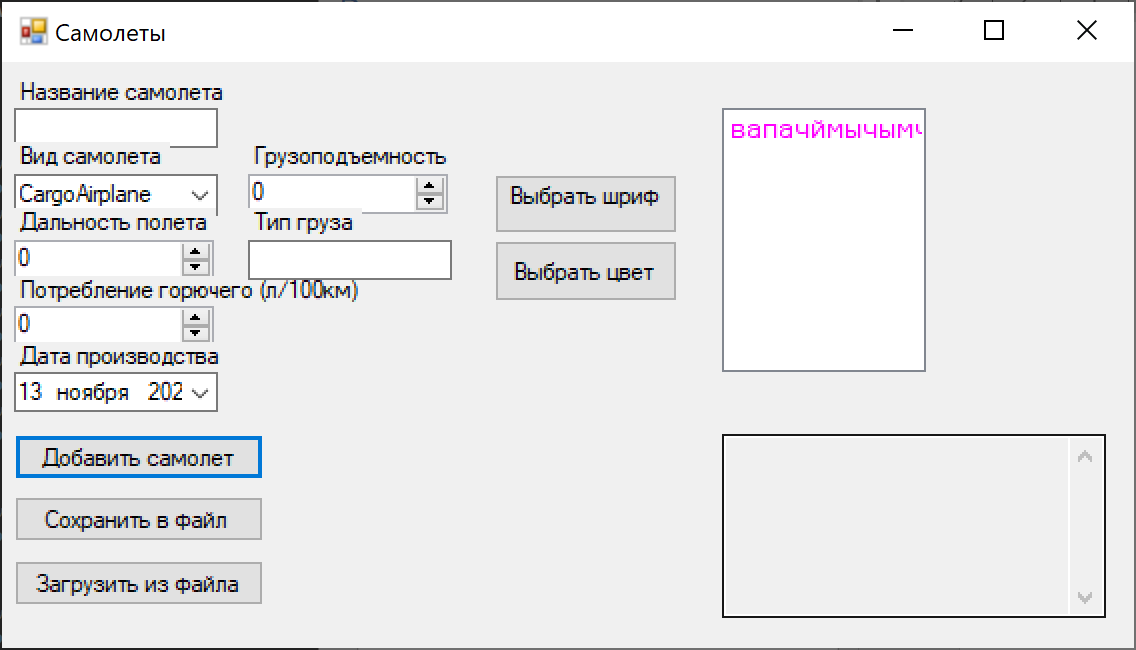


Рисунок 12 – Тест программы

* 1. **Интерфейс программы**

Интерфейс программы представляет собой форму для управления данными о самолетах, которая включает элементы для ввода и отображения информации, а также кнопки для выполнения различных действий с данными. В интерфейсе есть текстовое поле для ввода названия самолета, комбинированный список для выбора модели (пассажирский или грузовой), числовые поля для ввода дальности полета и расхода топлива, а также поле для выбора даты выпуска. Для каждого типа самолета (пассажирского или грузового) отображаются специфические поля: для грузового самолета — это грузоподъемность и тип груза, а для пассажирского — вместимость и наличие бизнес-класса. В интерфейсе также предусмотрены кнопки для добавления нового самолета в список, сохранения и загрузки данных о самолетах из файла, а также кнопка для отображения подробной информации о выбранном самолете. Для изменения внешнего вида списка самолетов, пользователю предоставляются кнопки для выбора шрифта и цвета текста через диалоги FontDialog и ColorDialog. Все добавленные самолеты отображаются в ListBox, а подробная информация о выбранном самолете выводится в текстовое поле.

**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.