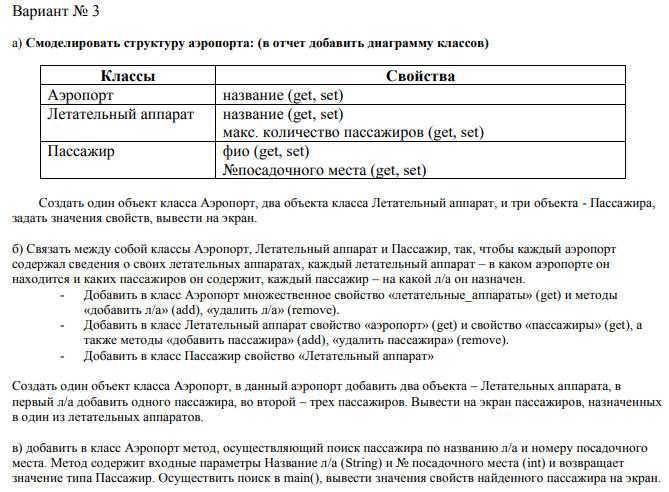
**ЛАБОРАТОРНАЯ РАБОТА №7**

**СОЗДАНИЕ КЛАССОВ И РАБОТА С ОБЪЕКТАМИ**

**Цель работы:** получить основные понятия и навыки по созданию классов, описанию свойств и работы с объектами.



**A)**

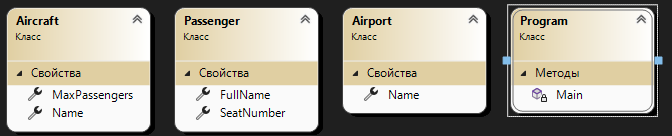


Рисунок 1 Диаграмма классов

**Листинг:**

using System;

using System.Collections.Generic;

class Airport

{

public string Name { get; set; }

}

class Aircraft

{

public string Name { get; set; }

public int MaxPassengers { get; set; }

}

class Passenger

{

public string FullName { get; set; }

public int SeatNumber { get; set; }

}

class Program

{

static void Main(string[] args)

{

// Создание объекта аэропорта

Airport airport = new Airport();

airport.Name = "Шереметьево";

// Создание двух объектов самолетов

Aircraft aircraft1 = new Aircraft();

aircraft1.Name = "Boeing 737";

aircraft1.MaxPassengers = 200;

Aircraft aircraft2 = new Aircraft();

aircraft2.Name = "Airbus A320";

aircraft2.MaxPassengers = 180;

// Создание трех объектов пассажиров

Passenger passenger1 = new Passenger();

passenger1.FullName = "Иванов Иван Иванович";

passenger1.SeatNumber = 1;

Passenger passenger2 = new Passenger();

passenger2.FullName = "Петров Петр Петрович";

passenger2.SeatNumber = 2;

Passenger passenger3 = new Passenger();

passenger3.FullName = "Сидоров Сидор Сидорович";

passenger3.SeatNumber = 3;

// Вывод информации на экран

Console.WriteLine("Информация об аэропорте:");

Console.WriteLine("Название: " + airport.Name);

Console.WriteLine();

Console.WriteLine("Информация о летательных аппаратах:");

Console.WriteLine("1. Название: " + aircraft1.Name);

Console.WriteLine(" Максимальное количество пассажиров: " + aircraft1.MaxPassengers);

Console.WriteLine("2. Название: " + aircraft2.Name);

Console.WriteLine(" Максимальное количество пассажиров: " + aircraft2.MaxPassengers);

Console.WriteLine();

Console.WriteLine("Информация о пассажирах:");

Console.WriteLine("1. ФИО: " + passenger1.FullName);

Console.WriteLine(" Номер посадочного места: " + passenger1.SeatNumber);

Console.WriteLine("2. ФИО: " + passenger2.FullName);

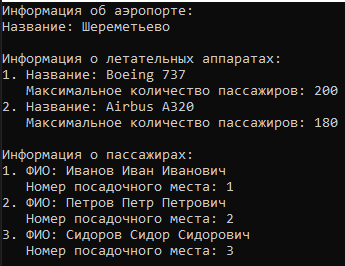
Console.WriteLine(" Номер посадочного места: " + passenger2.SeatNumber);

Console.WriteLine("3. ФИО: " + passenger3.FullName);

Console.WriteLine(" Номер посадочного места: " + passenger3.SeatNumber);

}

}



Б)

Листинг

using System;

using System.Collections.Generic;

class Airport

{

public string Name { get; set; }

private List<Aircraft> aircrafts = new List<Aircraft>();

public IReadOnlyList<Aircraft> Aircrafts => aircrafts;

public void AddAircraft(Aircraft aircraft)

{

aircraft.Airport = this;

aircrafts.Add(aircraft);

}

public void RemoveAircraft(Aircraft aircraft)

{

aircraft.Airport = null;

aircrafts.Remove(aircraft);

}

}

class Aircraft

{

public string Name { get; set; }

public int MaxPassengers { get; set; }

public Airport Airport { get; set; }

private List<Passenger> passengers = new List<Passenger>();

public IReadOnlyList<Passenger> Passengers => passengers;

public void AddPassenger(Passenger passenger)

{

passenger.Aircraft = this;

passengers.Add(passenger);

}

public void RemovePassenger(Passenger passenger)

{

passenger.Aircraft = null;

passengers.Remove(passenger);

}

}

class Passenger

{

public string FullName { get; set; }

public int SeatNumber { get; set; }

public Aircraft Aircraft { get; set; }

}

class Program

{

static void Main(string[] args)

{

// Создание объекта аэропорта

Airport airport = new Airport();

airport.Name = "Шереметьево";

// Создание двух объектов самолетов и добавление их в аэропорт

Aircraft aircraft1 = new Aircraft();

aircraft1.Name = "Boeing 737";

aircraft1.MaxPassengers = 200;

airport.AddAircraft(aircraft1);

Aircraft aircraft2 = new Aircraft();

aircraft2.Name = "Airbus A320";

aircraft2.MaxPassengers = 180;

airport.AddAircraft(aircraft2);

// Создание пассажиров и добавление их в летательные аппараты

Passenger passenger1 = new Passenger();

passenger1.FullName = "Иванов Иван Иванович";

passenger1.SeatNumber = 1;

aircraft1.AddPassenger(passenger1);

Passenger passenger2 = new Passenger();

passenger2.FullName = "Петров Петр Петрович";

passenger2.SeatNumber = 2;

aircraft2.AddPassenger(passenger2);

Passenger passenger3 = new Passenger();

passenger3.FullName = "Сидоров Сидор Сидорович";

passenger3.SeatNumber = 3;

aircraft2.AddPassenger(passenger3);

// Вывод информации о пассажирах, находящихся во втором летательном аппарате

Console.WriteLine("Пассажиры, назначенные во второй летательный аппарат:");

foreach (var passenger in aircraft2.Passengers)

{

Console.WriteLine("ФИО: " + passenger.FullName + ", Номер посадочного места: " + passenger.SeatNumber);

}

}

}



В)

Листинг

using System;

using System.Collections.Generic;

class Airport

{

public string Name { get; set; }

private List<Aircraft> aircrafts = new List<Aircraft>();

public IReadOnlyList<Aircraft> Aircrafts => aircrafts;

public void AddAircraft(Aircraft aircraft)

{

aircraft.Airport = this;

aircrafts.Add(aircraft);

}

public void RemoveAircraft(Aircraft aircraft)

{

aircraft.Airport = null;

aircrafts.Remove(aircraft);

}

public Passenger FindPassenger(string aircraftName, int seatNumber)

{

foreach (var aircraft in aircrafts)

{

if (aircraft.Name == aircraftName)

{

foreach (var passenger in aircraft.Passengers)

{

if (passenger.SeatNumber == seatNumber)

{

return passenger;

}

}

}

}

return null;

}

}

class Aircraft

{

public string Name { get; set; }

public int MaxPassengers { get; set; }

public Airport Airport { get; set; }

private List<Passenger> passengers = new List<Passenger>();

public IReadOnlyList<Passenger> Passengers => passengers;

public void AddPassenger(Passenger passenger)

{

passenger.Aircraft = this;

passengers.Add(passenger);

}

public void RemovePassenger(Passenger passenger)

{

passenger.Aircraft = null;

passengers.Remove(passenger);

}

}

class Passenger

{

public string FullName { get; set; }

public int SeatNumber { get; set; }

public Aircraft Aircraft { get; set; }

}

class Program

{

static void Main(string[] args)

{

// Создание объекта аэропорта

Airport airport = new Airport();

airport.Name = "Шереметьево";

// Создание двух объектов самолетов и добавление их в аэропорт

Aircraft aircraft1 = new Aircraft();

aircraft1.Name = "Boeing 737";

aircraft1.MaxPassengers = 200;

airport.AddAircraft(aircraft1);

Aircraft aircraft2 = new Aircraft();

aircraft2.Name = "Airbus A320";

aircraft2.MaxPassengers = 180;

airport.AddAircraft(aircraft2);

// Создание пассажиров и добавление их в летательные аппараты

Passenger passenger1 = new Passenger();

passenger1.FullName = "Иванов Иван Иванович";

passenger1.SeatNumber = 1;

aircraft1.AddPassenger(passenger1);

Passenger passenger2 = new Passenger();

passenger2.FullName = "Петров Петр Петрович";

passenger2.SeatNumber = 2;

aircraft2.AddPassenger(passenger2);

Passenger passenger3 = new Passenger();

passenger3.FullName = "Сидоров Сидор Сидорович";

passenger3.SeatNumber = 3;

aircraft2.AddPassenger(passenger3);

// Поиск пассажира по названию летательного аппарата и номеру посадочного места

string searchAircraftName = "Airbus A320";

int searchSeatNumber = 3;

Passenger foundPassenger = airport.FindPassenger(searchAircraftName, searchSeatNumber);

// Вывод информации о найденном пассажире

if (foundPassenger != null)

{

Console.WriteLine("Найден пассажир:");

Console.WriteLine("ФИО: " + foundPassenger.FullName);

Console.WriteLine("Номер посадочного места: " + foundPassenger.SeatNumber);

Console.WriteLine("Летательный аппарат: " + foundPassenger.Aircraft.Name);

}

else

{

Console.WriteLine("Пассажир не найден.");

}

}

}

