Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение  
высшего образования   
«Алтайский государственный технический университет им. И. И. Ползунова»

Факультет информационных технологий

Кафедра прикладной информатики

Отчет защищен с оценкой \_\_\_\_\_

Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2023 г.

Отчет

По лабораторной работе №2

**«Реализация АТД на языке С и С++»**

по дисциплине «Программирование – 3 семестр»

Студент группы ПИ-23: Карев Иван Александрович

Преподаватель Троицкий В. С.

Барнаул 2023

**Задание:** Придумать и кратко описать некоторую прикладную задачу для решения которой обрабатываемые объекты разумно представлять в виде структур (не менее 5 шт. + они должны быть взаимосвязаны); Выделить набор значимых свойств этих объектов и разработать набор соответствующих структур языка C для их представления; Разработать набор функций на языке C для работы с этими структурами; В main продемонстрировать работу со статическими и динамическими переменными этих типов.

Модифицировать разработанную программу. Заменить struct на class; Функции становятся методами класса. Поля становятся private, доступ к ним при необходимости осуществляется через методы public; Дополнительно в main продемонстрировать работу с динамическим массивом объектов класса и с массивом динамических объектов класса; применение операторов new, delete, delete[]; использование (\*a) и ->.

**Прикладная задача:**

Необходима база данных которая хранит информацию о клиентах, заказах, билетах и операциях, и нуждается в удобном способе управления и взаимодействия с этой информацией.

**Предметная область:**

Создана статическая библиотека. Вот краткий обзор задач, которые могут быть решены с использованием этой библиотеки:

**Управление клиентами:**

Добавление новых клиентов в базу данных.

Просмотр списка клиентов.

**Управление заказами:**

Добавление информации о заказах (например, компания, дистрибьютор, цена) в базу данных.

Просмотр списка заказов.

**Управление билетами:**

Добавление информации о билетах (место отправления, место назначения, цена) в базу данных.

Просмотр списка билетов.

**Управление операциями:**

Добавление информации о операциях (например, дата операции, тип операции) в базу данных.

Просмотр списка операций.

**Ввод данных от пользователя:**

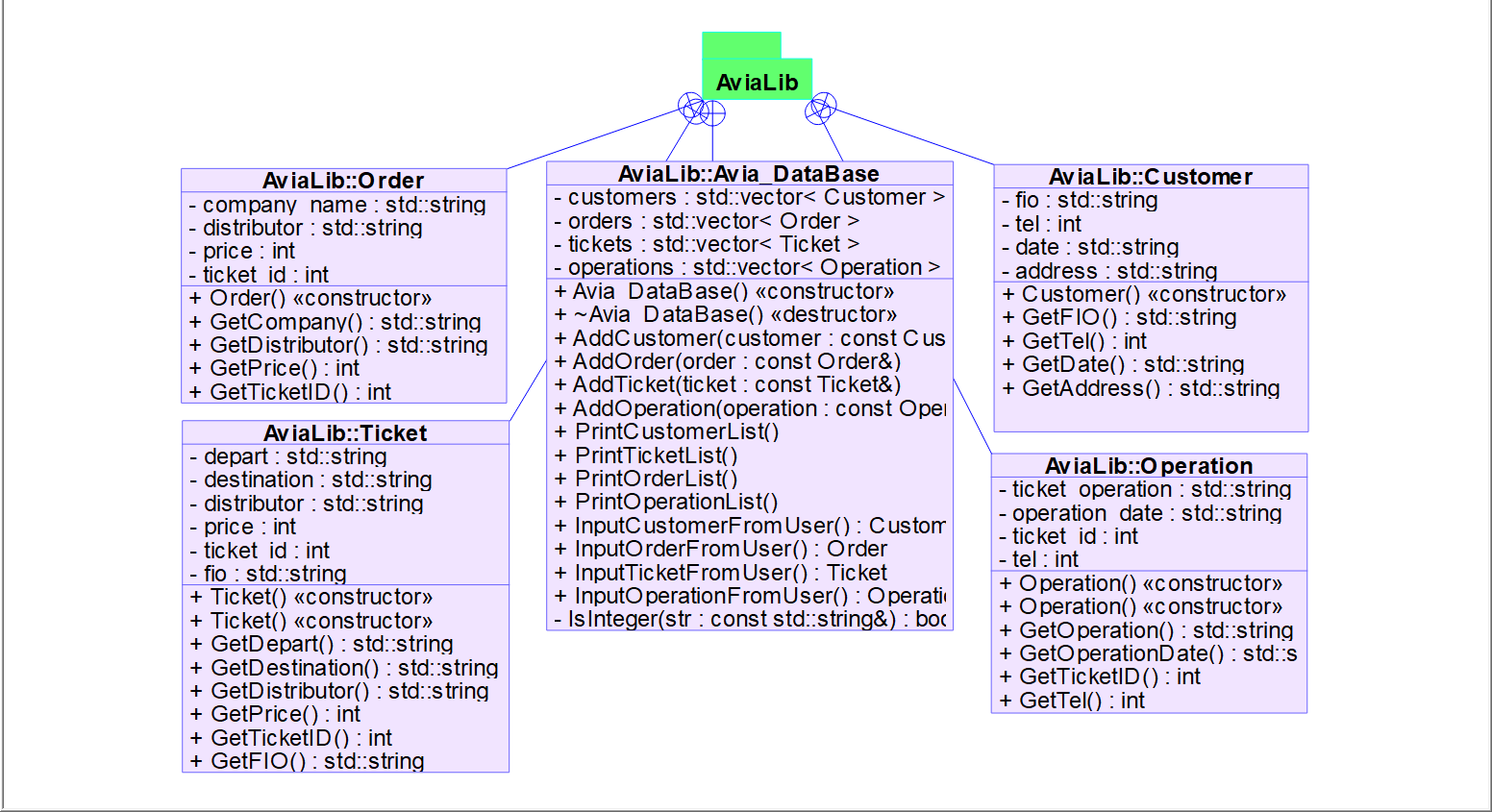
Классы также предоставляют функции для ввода данных от пользователя (например, ввод информации о клиенте, заказе, билете или операции).

Вот более подробное описание предметной области:

1. **Клиенты (Customers)**:
   * В этой предметной области клиенты представляют собой пассажиров, которые могут приобретать авиабилеты и размещать заказы.
   * Каждый клиент имеет уникальное имя (ФИО), контактный номер телефона, адрес и дату регистрации.
2. **Заказы (Orders)**:
   * Заказы представляют собой запросы от клиентов на приобретение авиабилетов.
   * Заказы могут содержать информацию о компании-перевозчике, дистрибьюторе, стоимости и идентификаторе билета.
3. **Билеты (Tickets)**:
   * Билеты представляют собой фактические авиабилеты, которые клиенты могут приобрести.
   * Каждый билет содержит информацию о месте отправления, месте назначения, дистрибьюторе, цене, идентификаторе билета и клиенте, который его приобрел.
4. **Операции (Operations)**:
   * Операции связаны с различными действиями и событиями, связанными с билетами и клиентами.
   * Примеры операций включают продажу билетов, возврат билетов, изменение данных клиентов и другие события.
   * Каждая операция содержит информацию о типе операции, дате операции, идентификаторе билета и контактном номере телефона.
5. **База данных (Database)**:
   * Библиотека предоставляет механизм для управления базой данных, включая добавление данных о клиентах, заказах, билетах и операциях, а также методы для просмотра списков этих данных.
   * Также предоставляются функции ввода данных от пользователя для удобной интерактивной работы с базой данных.

*Эта предметная область охватывает различные аспекты управления данными в авиационной компании, что может быть полезно для отслеживания информации о клиентах, заказах, билетах и операциях, а также для автоматизации и упрощения процессов, связанных с авиабилетами и клиентами.*

**UML-Диаграмма классов:**

****

**Пример работы с библиотекой:**

Код программы:

TestApplication.cpp

#include <iostream>

#include "../avialib/Avialib.h"

#include <Windows.h>

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

AviaLib::Avia\_DataBase aviaDB;

// Тестирование функций ввода данных

AviaLib::Customer customer = aviaDB.InputCustomerFromUser();

AviaLib::Order order = aviaDB.InputOrderFromUser();

AviaLib::Ticket ticket = aviaDB.InputTicketFromUser();

AviaLib::Operation operation = aviaDB.InputOperationFromUser();

// Добавление данных в базу данных

aviaDB.AddCustomer(customer);

aviaDB.AddOrder(order);

aviaDB.AddTicket(ticket);

aviaDB.AddOperation(operation);

// Тестирование функций вывода данных

std::cout << "Список покупателей:" << std::endl;

aviaDB.PrintCustomerList();

std::cout << "\nСписок заказов:" << std::endl;

aviaDB.PrintOrderList();

std::cout << "\nСписок билетов:" << std::endl;

aviaDB.PrintTicketList();

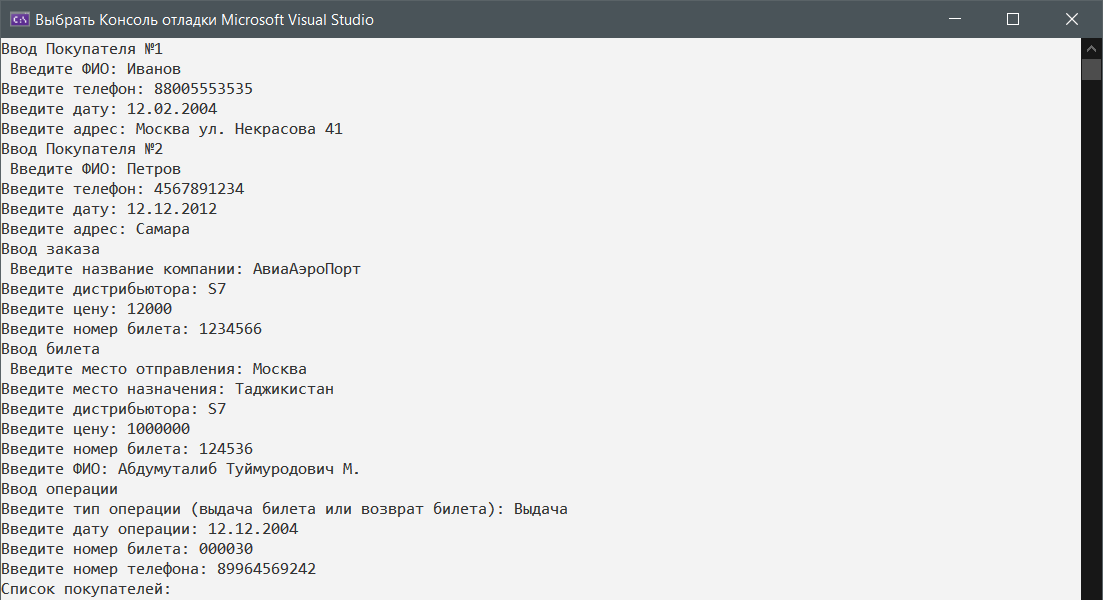
std::cout << "\nСписок операций:" << std::endl;

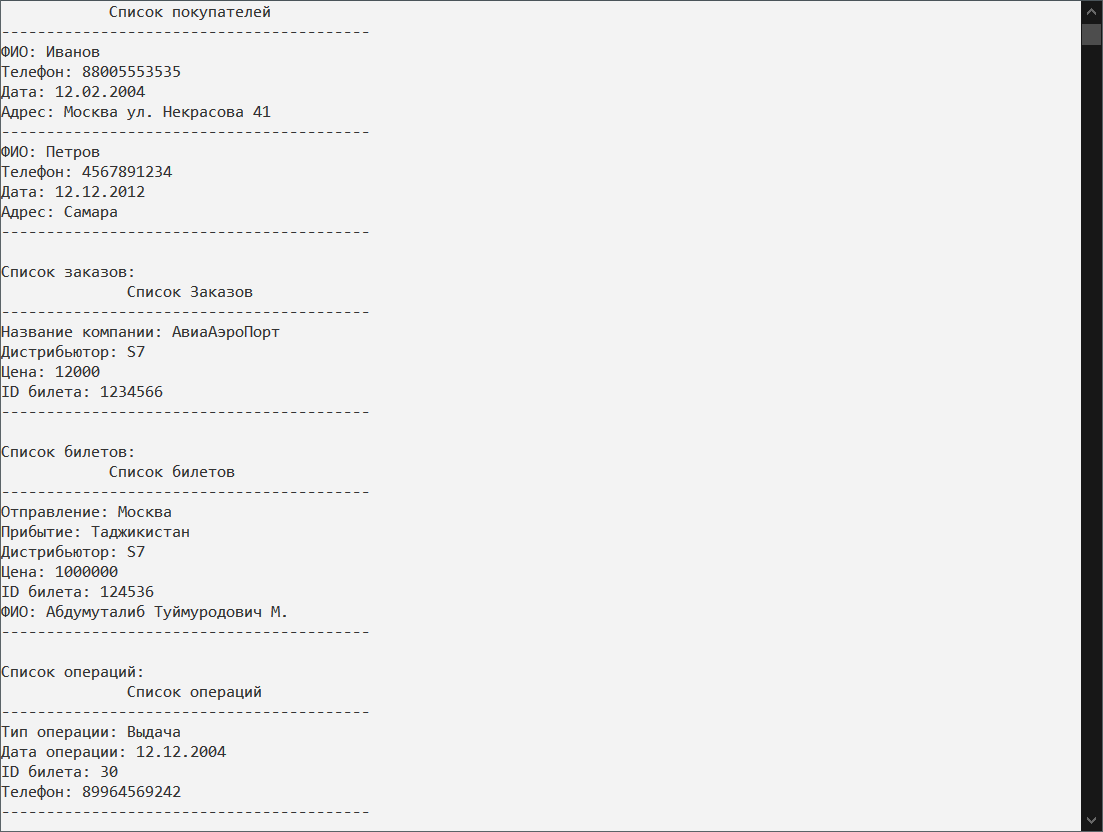
aviaDB.PrintOperationList();

return 0;

}

Тест программы:





Ссылки:

1. <https://github.com/VanoCry/Avialib> (Репозиторий)
2. <https://github.com/VanoCry/Avialib/blob/master/StaticLib/avialib/Avialib.h> (Заголовочный файл)
3. <https://github.com/VanoCry/Avialib/blob/master/StaticLib/avialib/avialib.cpp> (Основной файл статической библиотеки)
4. <https://github.com/VanoCry/Avialib/blob/master/StaticLib/TestApplication/TestApplication.cpp> (Тестовый файл)