Содержание

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | 4 |  | 3 |
| Введение | 3 | 5 |  |  |
| 1 Описание предметной области | 4 | 5 |  | 4 |
| 1.1 Постановка задачи | 4 | 5 |  | 5 |
| 1.2 Объектно-ориентированный анализ предметной области | 4 | 11 |  | 4 |
| 2 Проектирование программного средства | 6 | 11 |  |  |
| 2.1 Проектирование структур данных | 6 | 15 |  |  |
| 2.2 Проектирование взаимодействия с пользователем | 6 | 15 |  |  |
| 2.3 Проектирование интерфейса | 7 | 17 |  |  |
| 3 Реализация программного средства | 12 | 19 |  |  |
| 3.1 Реализация классов и структур данных | 12 | 19 |  |  |
| 3.2 Реализация требований | 13  14 | 20 |  |  |
| 3.3 Реализация концепций ООП | 16 | 21 |  |  |
| 3.4 Описание структуры проекта | 16 | 21 |  |  |
| 3.5 Реализация интерфейса взаимодействия с пользователем | 17 | 26 |  |  |
| 4 Тестирование программного модуля | 17 | 33 |  |  |
| Заключение Список использованных источников | 27 |  |  |  |
| Введение |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  | 57 |  |  |
|  |  | 58 |  |  |
|  |  | 60 |  |  |
|  |  | 61 |  |  |
|  |  | 62 |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Введение

Целью курсового проекта является создание программного средства для разработки базы данных пациентов ветеринарной клиники.

Задачи курсовой работы:

- изучить предметную область и установить требования к программному продукту;

- спроектировать программный продукт;

- разработать программный продукт;

- протестировать работу программного продукта.

В результате выполнения курсовой работы должно быть разработано полностью рабочее приложение, позволяющее выводить всю базу данных пациентов клиники, добавлять новые записи в базу данных, проводить поиск среди элементов базы данных, удалять записи из базы по введенному имени пациента, сохранять базу данных в файл, редактировать записи базы данных.

В пояснительной записке описан процесс создания программного продукта, описаны принципы его работы и результаты тестирования.

1 Описание предметной области

* 1. Постановка задачи

Целью данной курсовой работы является разработка программы на языке С++ для создания базы данных ветеринарной клиники для учета пациентов.

Данные программы должны храниться в файлах. Каждая логически завершенная задача программы должна быть реализована в виде метода. Разрабатываемый интерфейс должен быть понятным и защищенным от случайных ошибок, вылетов и ввода некорректных данных. Ввод исходных данных должен осуществляться либо с клавиатуры, либо из файла.

Разрабатываемая программа должна обеспечивать возможность выполнения перечисленных ниже функций:

-вывод данных, хранящихся в файле;

-запись данных в файл;

-добавление/удаление пациентов;

-добавление/удаление врачей;

-добавление/удаление приемов;

-добавление/удаление владельцев;

-сортировка приемов по фамилиям владельцев (по алфавиту);

-сортировка приемов по дате (по возрастанию и убыванию).

Требования к организации ввода входных данных:

-входные данные должны соответствовать меню программного обеспечения.

При выполнении программы необходимо предусмотреть обработку следующих ошибок:

- неправильный формат исходных данных;

-ошибка чтения/записи в файл;

-несоответствие номера пункта меню.

В программе будет возможность ввода нового пациента, хозяина, врача. Работа над ними: добавление, удаление.

Также будет возможность добавления приема пациентов, которая включает в себя: выбор врача, выбор животного, выбор хозяина, ввод года, месяца и дня приема.

* 1. Объектно-ориентированный анализ предметной области

Ветеринарная клиника оказывает помощь больным животным. Животные делятся на домашних и бездомных. Животные характеризуются именем, видом, породой, данными о хозяине, лечащем враче и диагнозе. Хозяин характеризуется именем и фамилией, домашним адресом, номером телефона. Лечащие врачи характеризуются именем и фамилией, специализацией, номером телефона и расписанием приема.

2 Проектирование программного средства

2.1 Проектирование структуры данных

Диаграмма классов

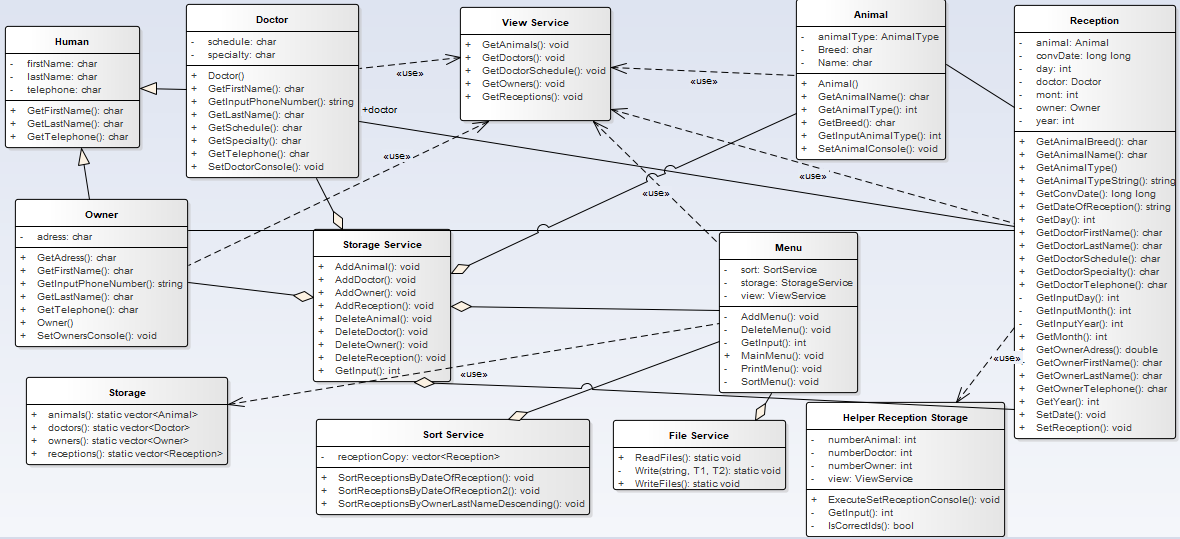
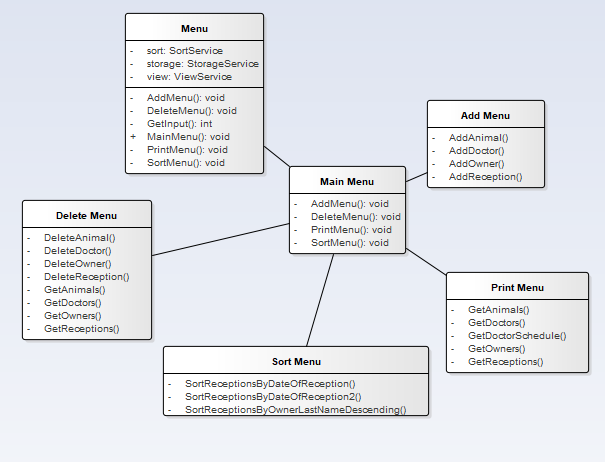
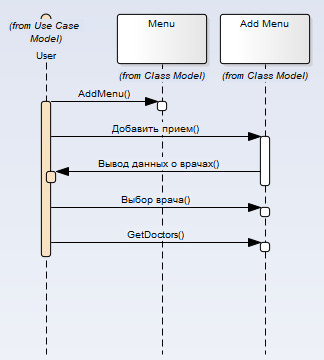
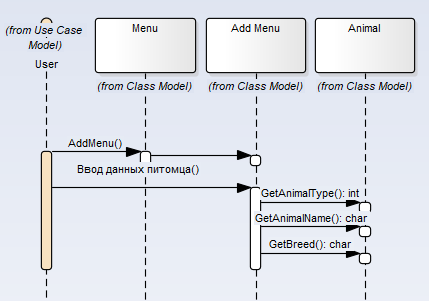


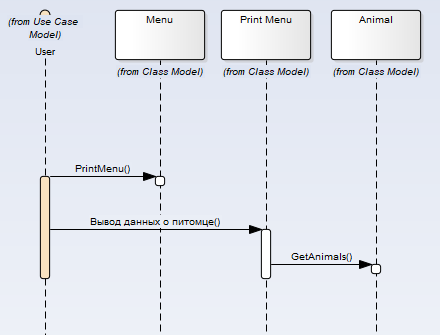
Диаграмма класса меню

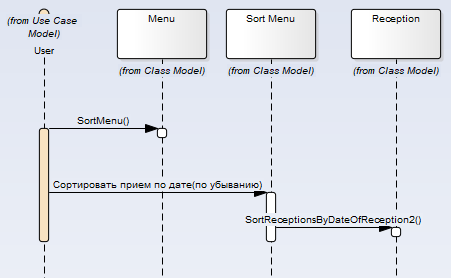


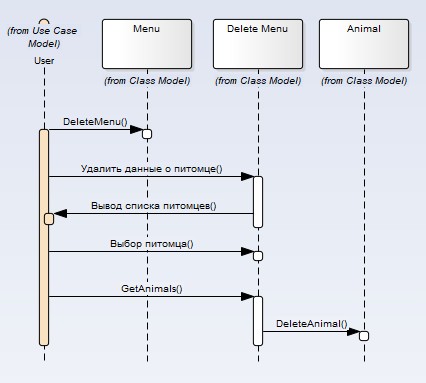
2.2 Проектирование взаимодействия с пользователем

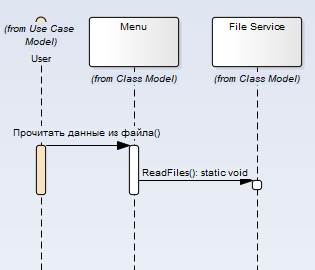


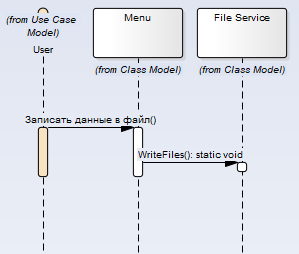




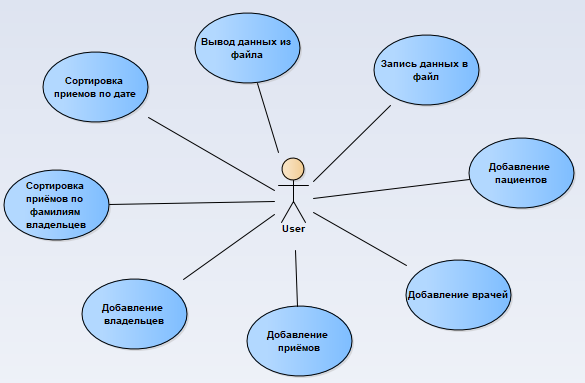








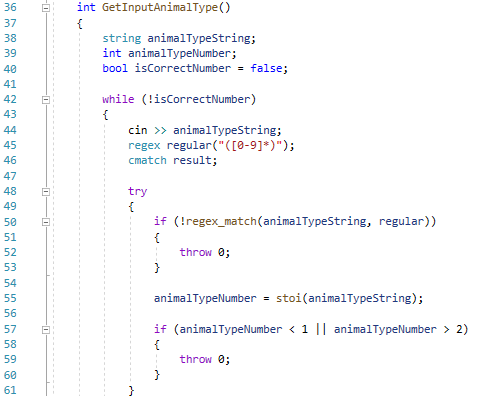
2.3 Проектирование интерфейса



3 Реализация программного средства

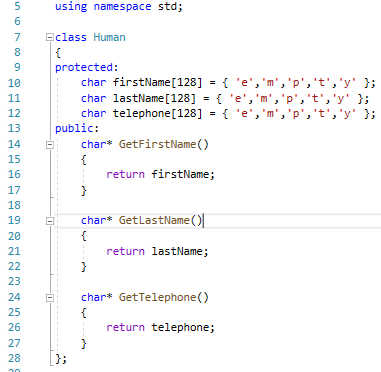
3.1 Реализация классов и структур данных

Реализация класса животные

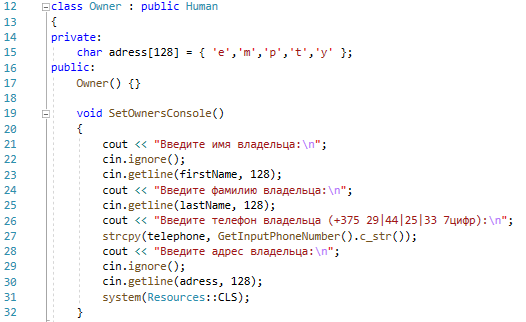




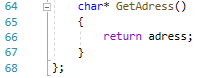
Реализация родительского класса человек



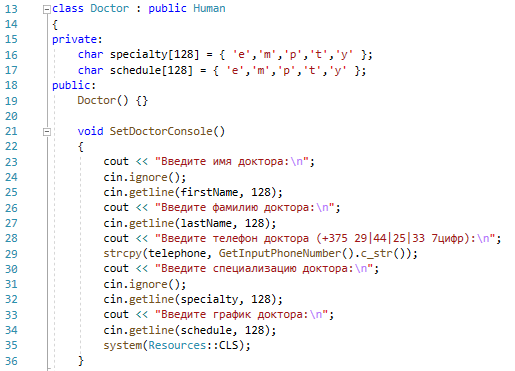
Реализация дочернего класса хозяин



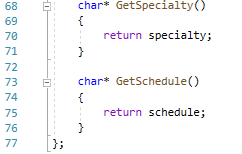




Реализация дочернего класса врача







3.2 Реализация требований

|  |  |  |
| --- | --- | --- |
| Функциональное требование | Реализация | Примечание |
| Вывод данных,  хранящихся в файле | Метод ReadFiles класса FileService |  |
| Запись данных в файл | Метод WriteFiles класса FileService |  |
| Сортировка приемов по фамилии владельцев (по алфавиту) | Метод SortReceptionsByOwnerLastNameDescending  класса SortService |  |
| Сортировка приемов по дате (по возрастанию) | Метод SortReceptionsByDateOfReception  Класса SortService |  |
| Сортировка приемов по дате (по убыванию) | Метод SortReceptionsByDateOfReception2  Класса SortService |  |
| Проверка корректности выбора id | Метод IsCorrectIds класса HelperReceptionStorage  Метод GetInput класса StorageService |  |
| Проверка ввода данных (только цифры) | Метод GetInput класса HelperReceptionStorage  Метод GetInput класса Menu  Метод GetInputAnimalType класса Animal | В методе GetInputAnimalType также происходит проверка ввода id(id<1, id>2) |
| Проверка правильности ввода номера телефона | Метод GetInputPhoneNumber класса Doctor  Метод GetInputPhoneNumber класса Owner |  |
| Проверка ввода даты | Метод GetInputYear класса Reception  Метод GetInputMonth класса Reception  Метод GetInputDay класса Reception |  |
| Удаление объектов классов | Метод DeleteOwner класса StorageService  Метод DeleteAnimal класса StorageService  Метод DeleteDoctor класса StorageService  Метод DeleteReception класса StorageService |  |
| Добавление объектов классов | Метод AddOwner класса StorageService  Метод AddAnimal класса StorageService  Метод AddDoctor класса StorageService  Метод AddReception класса StorageService |  |
| Вывод данных на консоль | Метод GetAnimals класса ViewService  Метод GetDoctors класса ViewService  Метод GetOwners класса ViewService  Метод GetDoctorSchedule класса ViewService  Метод GetReceptions класса ViewService |  |

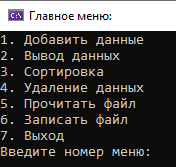
3.3 Реализация концепций ООП

Для реализации инкапсуляции ключевые поля классов имеют спецификатор доступа private.

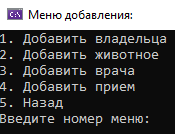
3.4 Описание структуры проекта

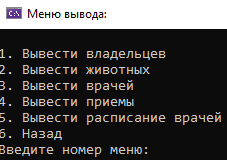
3.5 Описание интерфейса взаимодействия с пользователем

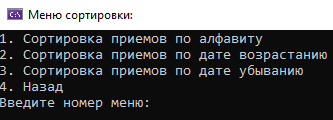
Изначально пользователю выводится на консоль главное меню:

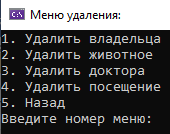


Далее пользователь выбирает пункт, который выводит на консоль одно из следующих меню:









Если же пользователь выбирает пункты 5 и 6 на консоль выводятся следующие сообщения:





Пользователь всегда имеет возможность вернуться назад, а также выйти из программы.

Если пользователь совершил ошибку во время ввода данных, ему выведется сообщение об ошибке и предоставится повторный ввод.



4 Тестирование программного модуля

При запуске программы пользователь увидит главное меню, в котором 7 пунктов (рис. 4.1).

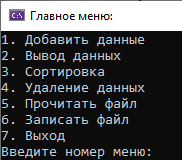


Рис. 4.1

Чтобы добавить данные следует выбрать пункт меню «Добавить данные». После выбора данного пункта выводится подменю «Меню добавления» (рис. 4.2).

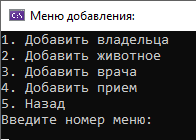


Рис. 4.2

При выборе пункта «Добавить владельца» пользователь попадает в окно внесения данных о владельце (рис. 4.3). В нем он вводит данные о владельце на английском или русском языке. После ввода последних данных пользователь попадает в подменю «Меню добавления». Из него он может вернуться в «Главное меню» или продолжить ввод данных.

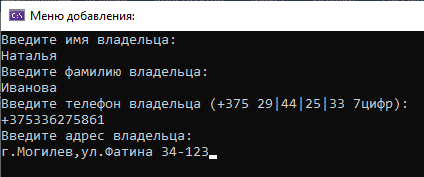


Рис. 4.3

При выборе пункта «Добавить животное» пользователь попадает в окно внесения данных о питомце (рис. 4.4). В нем он вводит данные о питомце на английском или русском языке. После ввода последних данных пользователь попадает в подменю «Меню добавления». Из него он может вернуться в «Главное меню» или продолжить ввод данных.

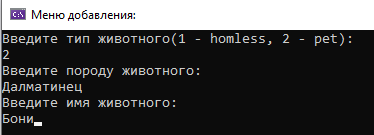


Рис. 4.4

При выборе пункта «Добавить врача» пользователь попадает в окно внесения данных о враче (рис. 4.5). В нем он вводит данные о враче на английском или русском языке. После ввода последних данных пользователь попадает в подменю «Меню добавления». Из него он может вернуться в «Главное меню» или продолжить ввод данных.

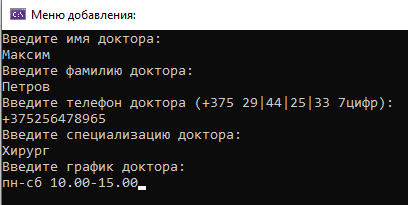


Рис. 4.5

При выборе пункта «Добавить прием» пользователь попадает в окно внесения данных о приеме (рис. 4.5). В нем пользователю выводятся данные о врачах, животных и владельцах. Пользователь выбирает id нужного ему врача, животного, владельца. Затем добавляет поочередно год, месяц и день приема (рис. 4.6). После ввода последних данных пользователь попадает в подменю «Меню добавления». Из него он может вернуться в «Главное меню» или продолжить ввод данных.

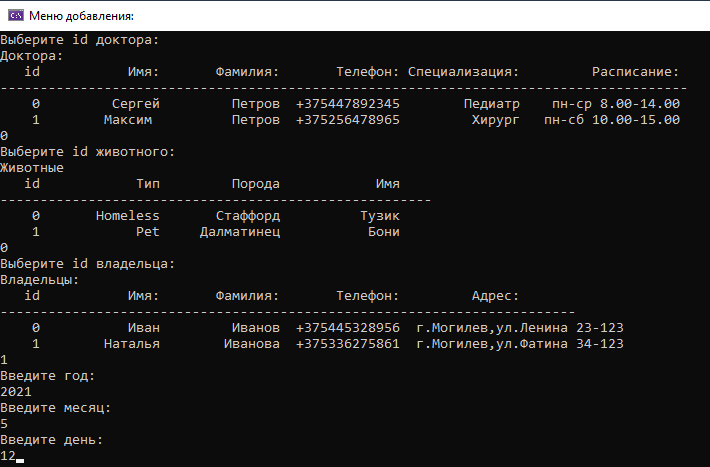


Рис. 4.6

При выборе пункта меню «Назад» пользователь попадает обратно в главное меню.

Для того чтобы вывести данные на консоль следует выбрать пункт «Вывод данных». После чего пользователь попадает в подменю «Меню вывода» (рис. 4.7).

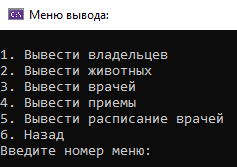


Рис. 4.7

При выборе пункта «Вывести владельцев» на консоль выводятся данные о владельцах (рис. 4.8). После вывода данных пользователю выводится подменю «Меню вывода». Из него он может вернуться в «Главное меню» или продолжить вывод данных.

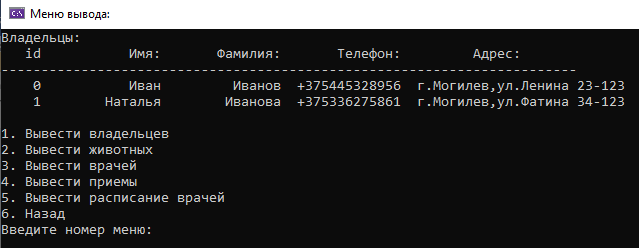


Рис. 4.8

При выборе пункта «Вывести животных» на консоль выводятся данные о животных (рис. 4.9). После вывода данных пользователю выводится подменю «Меню вывода». Из него он может вернуться в «Главное меню» или продолжить вывод данных.

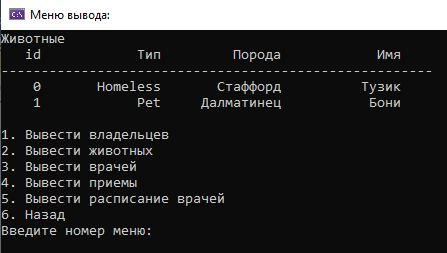


Рис. 4.9

При выборе пункта «Вывести врачей» на консоль выводятся данные о врачах (рис. 4.10). После вывода данных пользователю выводится подменю «Меню вывода». Из него он может вернуться в «Главное меню» или продолжить вывод данных.

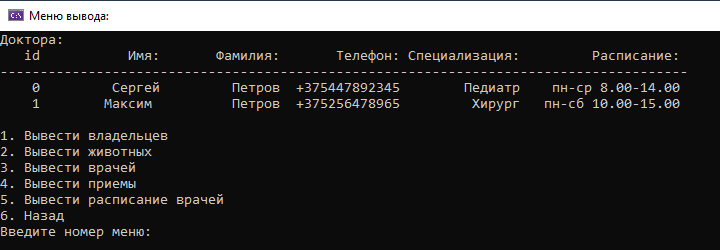


Рис. 4.10

При выборе пункта «Вывести приемы» на консоль выводятся данные о приемах (рис. 4.11-4.12). После вывода данных пользователю выводится подменю «Меню вывода». Из него он может вернуться в «Главное меню» или продолжить вывод данных.

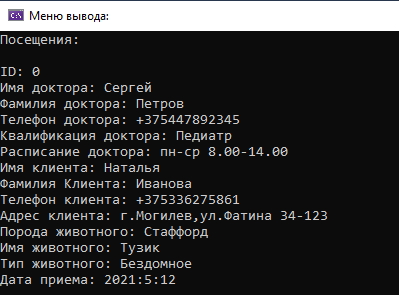


Рис. 4.11

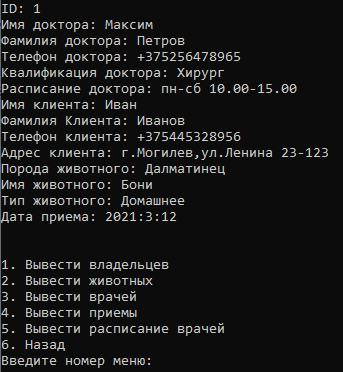


Рис. 4.12

При выборе пункта «Вывести расписание врачей» на консоль выводятся данные о расписании врачей (рис. 4.13). После вывода данных пользователю выводится подменю «Меню вывода». Из него он может вернуться в «Главное меню» или продолжить вывод данных.

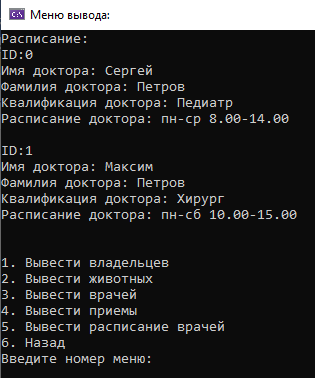


Рис. 4.13

При выборе пункта меню «Назад» пользователь попадает обратно в главное меню.

Для того чтобы произвести сортировку данных следует выбрать пункт «Сортировка». После чего пользователь попадает в подменю «Меню сортировки» (рис. 4.14).

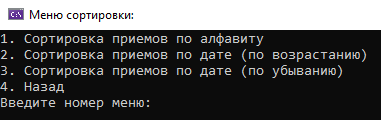


Рис. 4.14

При выборе пункта «Сортировка приемов по алфавиту» на консоль выводятся приемы, которые были отсортированы согласно алфавиту (сортировка производится по фамилиям владельцев) (рис. 4.15). После вывода данных пользователю выводится подменю «Меню сортировки». Из него он может вернуться в «Главное меню» или продолжить сортировку данных.

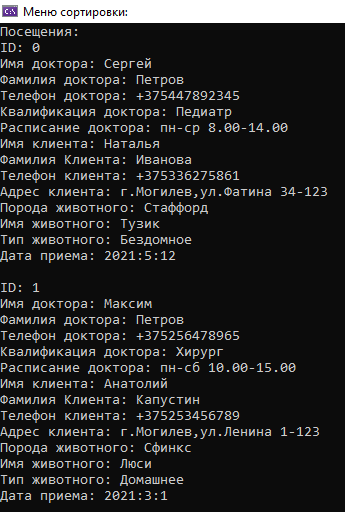
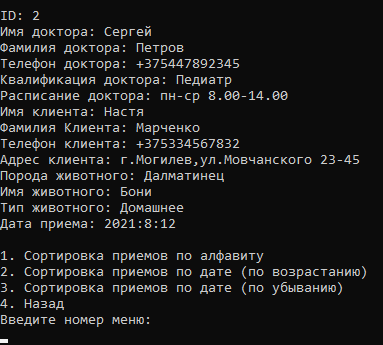


Рис. 4.15

Рис. 4.15

При выборе пункта «Сортировка приемов по дате (по возрастанию)» на консоль выводятся приемы, которые были отсортированы по дате в порядке возрастания (рис. 4.16). После вывода данных пользователю выводится подменю «Меню сортировки». Из него он может вернуться в «Главное меню» или продолжить сортировку данных.

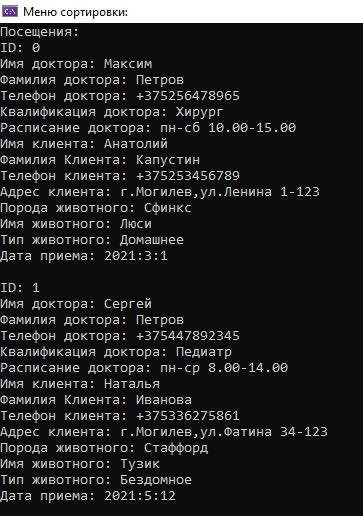
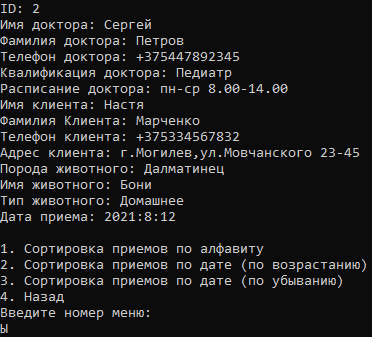


Рис. 4.16

При выборе пункта «Сортировка приемов по дате (по убыванию)» на консоль выводятся приемы, которые были отсортированы по дате в порядке убывания (рис. 4.17). После вывода данных пользователю выводится подменю «Меню сортировки». Из него он может вернуться в «Главное меню» или продолжить сортировку данных.

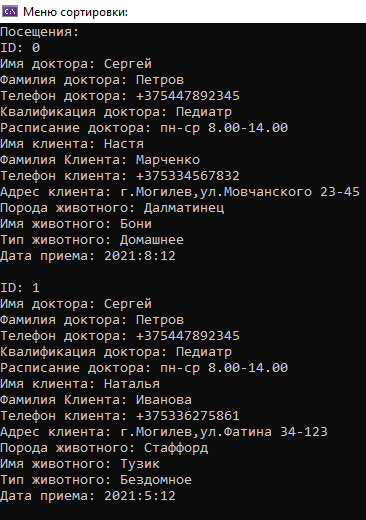
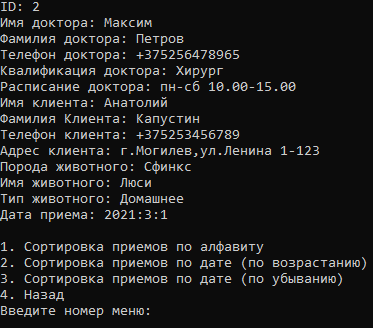


Рис. 4.17

При выборе пункта меню «Назад» пользователь попадает обратно в главное меню.

Для того чтобы произвести удаление данных следует выбрать пункт «Удаление данных». После чего пользователь попадает в подменю «Меню удаления» (рис. 4.18).

При выборе пункта «Удалить владельца» на консоль выводятся данные о владельцах. Пользователю следует выбрать id того владельца, данные о котором он хочет удалить (рис. 4.18). Затем пользователю выводятся данные после удаления владельца и подменю «Меню удаления». Из него он может вернуться в «Главное меню» или продолжить удаление данных.

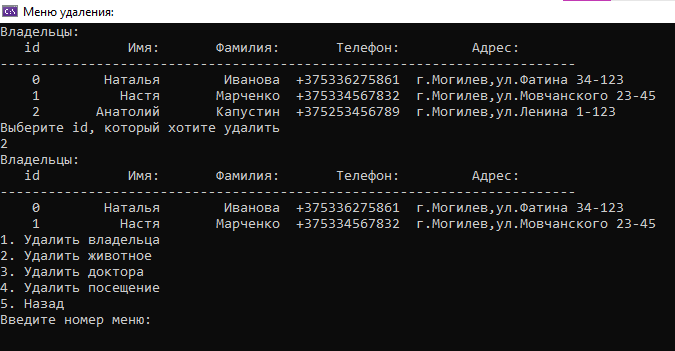


Рис. 4.18

При выборе пункта «Удалить животное» на консоль выводятся данные о животных. Пользователю следует выбрать id того животного, данные о котором он хочет удалить (рис. 4.19). Затем пользователю выводятся данные после удаления животного и подменю «Меню удаления». Из него он может вернуться в «Главное меню» или продолжить удаление данных.

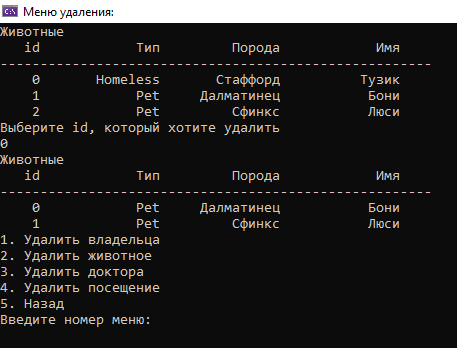


Рис. 4.19

При выборе пункта «Удалить доктора» на консоль выводятся данные о докторах. Пользователю следует выбрать id того доктора, данные о котором он хочет удалить (рис. 4.20). Затем пользователю выводятся данные после удаления доктора и подменю «Меню удаления». Из него он может вернуться в «Главное меню» или продолжить удаление данных.

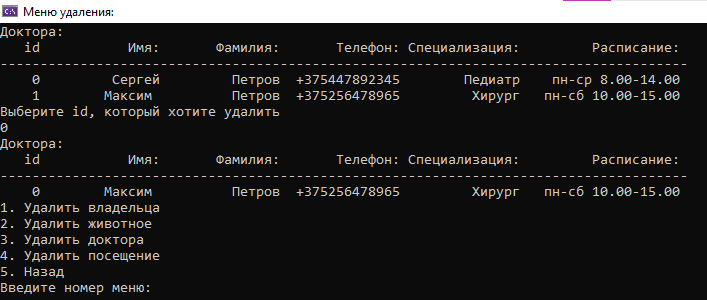


Рис. 4.20

При выборе пункта «Удалить посещение» на консоль выводятся данные о посещениях. Пользователю следует выбрать id того посещения, данные о котором он хочет удалить (рис. 4.21). Затем пользователю выводятся данные после удаления посещения и подменю «Меню удаления». Из него он может вернуться в «Главное меню» или продолжить удаление данных.

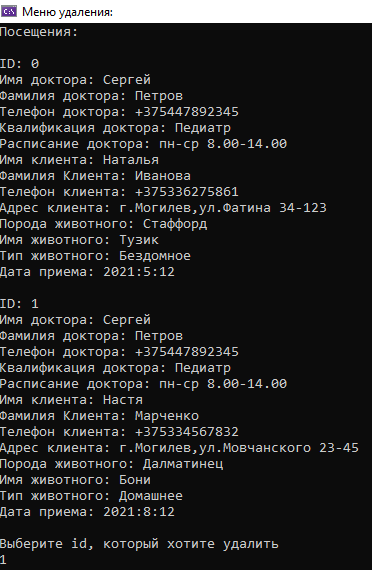
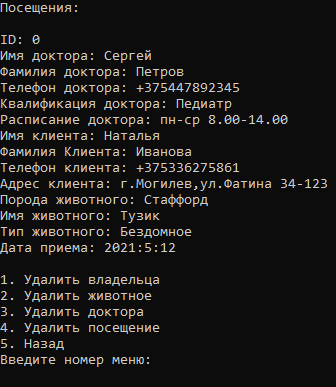


Рис. 4.21

При выборе пункта меню «Назад» пользователь попадает обратно в главное меню.

Для того чтобы прочитать данные из файла следует выбрать пункт «Прочитать файл». После чего данные будут прочитаны из файла и пользователю будет выведено соответствующее сообщение (рис. 4.22).

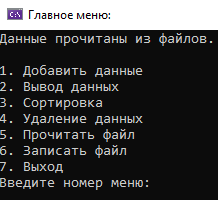
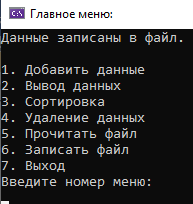


Рис. 4.22

Для того чтобы записать данные в файл следует выбрать пункт «Записать файл». После чего данные будут записаны в файл и пользователю будет выведено соответствующее сообщение (рис. 4.23).



Для того чтобы выйти из программы следует выбрать пункт «Выход».

# Заключение

В ходе проделанной работы, была разработана программа для создания базы данных ветеринарной клиники. Корректность работы программы подтверждена тестами.

В процессе выполнения работы были закреплены знания программирования на языке С++. Также получены практические навыки.

Достоинства данного программного обеспечения - это простой и понятный интерфейс для пользователя, множество проверок на корректность ввода, что исключает ошибки.

# Список литературы

1 <https://metanit.com/>

2. <https://www.cyberforum.ru/>

3. Васильев, А.Н. Программирование на C++ в примерах и задачах / А.Н. Васильев. - М.: ЭКСМО, 2017. - 416 c.

4. Методические указания к курсовому проектированию для студентов специальности 1-40 05 01 «Информационные системы и технологии (по направлениям)» дневной и заочной форм обучения